

Thèse

Présentée par

Laurent Pasquier

Pour obtenir le titre de

Docteur ès Sciences

de l'Université Pierre et Marie Curie

Spécialité Informatique

**Modélisation de raisonnements tenus en contexte.
Application à la gestion d'incidents sur une ligne de métro.**

Soutenue le 19 juillet 2002

Composition du jury :

Président du jury	Pr. Jean-Charles Pomerol
Rapporteurs	Pr. Roy Turner Pr. Charles Tijus
Directeur de thèse	Dr. Patrick Brézillon
Examineur	M. Jean-Maurice Bidault

A Virginie,

A mes Parents,

A ma Famille,

A mes Amis...

Mes premiers remerciements sont adressés à Patrick Brézillon, qui m'a guidé chaque jour pour mes premiers pas dans le monde de la recherche scientifique : Merci de m'avoir appris ce métier et encadré durant ces quatre années, merci de m'avoir enseigné des méthodes efficaces de recherches bibliographiques, merci de m'avoir donné les bases nécessaires à la publication des résultats et à la transmission orale de ceux-ci.

Je remercie également Jean-Charles Pomerol pour les nombreuses discussions fructueuses et ses relectures attentives de divers articles. Il me tient à cœur aussi de lui exprimer toute ma gratitude pour m'avoir fait l'honneur de présider le jury de soutenance.

Merci au LIP6, notamment au Poleia et à son personnel d'encadrement administratif, pour l'accueil qui m'a été fait. Je remercie particulièrement le thème SYSDEF et ses membres pour l'ambiance de travail qui règne au sein de cette entité et pour les activités annexes ayant accéléré mon intégration.

Je remercie la RATP pour l'encadrement et le financement de ces travaux de recherche et en particulier la Délégation Générale à l'Ingénierie d'Exploitation Ferroviaire pour m'avoir accueilli et facilité l'accès aux informations nécessaires. Je remercie le Département Développement pour la coordination des recherches au sein de l'entreprise et la réalisation du réseau des jeunes chercheurs.

J'exprime ma plus sincère gratitude à Corinne Clergerie pour son dévouement à la cause des CIFRE de la RATP, pour sa disponibilité, son écoute et ses conseils.

Je remercie également les chefs de régulation, leur encadrement et les équipes de formation pour leur aide et leur patience lors de la découverte de leurs métiers.

Une pensée toute particulière s'adresse à Catherine Zanarelli, pour m'avoir permis d'établir une correspondance entre mes travaux et les siens et pour m'avoir guidé dans mes premiers contacts avec l'ergonomie cognitive. Merci pour nos échanges aussi enrichissants scientifiquement qu'humainement agréables.

Je remercie également l'ANRT pour le financement de mes recherches appliquées et pour l'encadrement des relations entre le LIP6 et la RATP.

Merci à tous ceux qui ont relu ce document, soit pour en vérifier la validité scientifique, soit pour éliminer un maximum d'incorrections lexicales, grammaticales ou syntaxiques : Patrick Brézillon, Roy Turner, Charles Tijus, Jean-Charles Pomerol, Virginie P., Catherine, Virginie B. ...

Mes derniers mots sont pour Virginie, ma famille et mes amis qui ont eu la gentillesse de me supporter, dans tous les sens du terme, pendant ces quatre années et notamment pendant ces derniers mois.

Table des matières

<i>Tables des illustrations</i>	13
Figures	13
Tableaux	15
<i>Introduction</i>	16
Cadre de l'étude	16
Problématique	18
Approche suivie	21
Plan	22
<i>Premier chapitre : Des Systèmes Experts aux systèmes intelligents</i>	25
Section 1 : Systèmes experts et systèmes à base de connaissances	25
1.1 Limite des systèmes experts de première génération.....	26
1.2 Systèmes à base de connaissances	28
1.2.1 Acquisition de l'expertise	28
1.2.2 Différents types de systèmes experts	30
1.3 Limites des systèmes à base de connaissances	30

Section 2 : Systèmes d'aide	31
2.1 Systèmes Interactifs d'Aide à la Décision	31
2.2 Systèmes d'Assistance Intelligents	33
Section 3 : Importance du contexte	34
3.1 Définitions du contexte	35
3.2 Rôle du contexte dans un système d'aide	36
3.2.1 Rôle du contexte au niveau des connaissances	37
3.2.2 Rôle du contexte au niveau des raisonnements	37
3.2.3 Rôle du contexte lors de l'acquisition des connaissances	38
3.2.4 Rôle du contexte au niveau des explications	38
3.3 Dualité prescrit/réel et contexte	40
3.3.1 La notion de procédure	40
3.3.2 Stratégies et pratiques	41
3.3.3 Rôle du contexte dans la mise en pratique des procédures	41
3.3.4 Rôle du contexte dans la procéduralisation des pratiques réelles	43
Section 4 : Systèmes tenant compte du contexte	43
4.1 Le formalisme de McCarthy	43
4.2 Le système <i>ORCA</i> de Turner et les schémas contextuels	45
4.3 Context-Based Reasoning	47
4.4 « Context Aware Systems »	48
Section 5 : Système d'Aide Intelligent en Contexte	50
5.1 Modélisation du contexte	51
5.1.1 Connaissances contextuelles et connaissances externes	51
5.1.2 Contexte procéduralisé	53
5.1.3 Liens entre les différents états des éléments du contexte	53

5.2 Dynamiques du contexte.....	54
5.3 Définition d'un SAIC	55
Section 6 : Conclusion.....	56
<i>Deuxième chapitre : Gestion d'incidents sur une ligne de métro</i>	60
Section 1 : Le Métro Parisien.....	60
1.1 Un peu d'histoire	61
1.2 Structure du réseau.....	62
1.2.1 Qu'est-ce qu'une ligne de métro ?	62
1.2.2 Particularité du métro parisien	64
1.2.3 Eléments constitutifs d'une ligne.....	64
1.3 Exploitation du réseau	65
1.3.1 Organisation de l'exploitation.....	65
1.3.2 Incidents d'exploitation et gestion des incidents.....	69
Section 2 : Tâche et activité des chefs de régulation.....	70
2.1 Tâche des chefs de régulation.....	70
2.2 Activité des chefs de régulation.....	70
2.3 Gestion en exploitation normale	72
2.4 Gestion d'une situation incidentelle	72
2.4.1 Qu'est-ce qu'une situation incidentelle ?.....	72
2.4.2 Evénement incidentel.....	72
2.4.3 Gestion en deux phases.....	74
2.4.3.1 Phase de diagnostic et de gestion dans l'incertain.....	74
2.4.3.2 Phase de gestion d'un incident de type connu.....	75
2.4.4 Retour à la normale.....	75
Section 3 : Rôle du contexte dans la gestion d'incidents	76

3.1 Contexte de l'incident.....	77
3.2 Gestion de l'incident et évolution du contexte.....	78
3.3 Procédures et Pratiques réelles	79
Section 4 : Résultats de l'analyse de l'activité	80
4.1 Importance de l'expérience et du retour d'expérience.....	81
4.2 Concepts pragmatiques.....	82
4.3 Schèmes d'action.....	82
Section 5 : Systèmes d'aide à la gestion du trafic à la RATP.....	83
Section 6 : Conclusion.....	85
<i>Troisième chapitre : Le modèle de Graphes Contextuels.....</i>	<i>87</i>
Section 1 : Motivations et historique de la construction du modèle.....	87
Section 2 : Eléments constitutifs	94
2.1 Actions élémentaires.....	94
2.2 Activités.....	95
2.3 Nœuds contextuels.....	96
2.4 Nœuds de recombinaison.....	97
2.5 Branchements temporels.....	97
Section 3 : Graphes contextuels.....	98
3.1 Structure des Graphes Contextuels.....	98
3.2 Analyse de la structure.....	99
3.3 Algorithme d'acquisition incrémentale de connaissances.....	100
3.4 Autres algorithmes associés.....	104
3.5 Analyse de la complexité.....	105

3.6 Graphes Contextuels et dynamisme du contexte	105
Section 4 : Graphes contextuels et activités	107
4.1 Principes.....	107
4.2 Structuration de l'ensemble des activités.....	109
4.3 Conséquences sur l'acquisition incrémentale de connaissances.....	110
Section 5 : Conclusion.....	111
<i>Quatrième chapitre : Représentations de raisonnements et contexte.....</i>	<i>114</i>
Section 1 : Introduction	114
1.1 Motivations	114
1.2 Liens entre les représentations et l'objet représenté	115
Section 2 : Structuration des connaissances sur les raisonnements	116
2.1 Schéma.....	117
2.2 Frames.....	118
2.3 Scripts	119
2.4 MOPs	121
2.5 Schèmes	122
Section 3 : Raisonnement à Partir de Cas	125
3.1 Principes de fonctionnement.....	125
3.2 RàPC et contexte.....	127
3.3 Mesure de similarité dépendante du contexte.....	127
3.4 Limites du modèle	128
Section 4 : Formalismes de représentation par modèles informatiques	128
4.1 Modèles informatiques de structure similaire.....	129

4.1.1 Diagrammes d'influence.....	129
4.1.2 Réseaux de Petri.....	130
4.2 Diagramme d'activité dans UML	131
4.3 Décomposition des tâches et tâches génériques	133
4.4 Planification et planification dynamique	135
Section 5 : Conclusion.....	138
<i>Cinquième chapitre : Modélisation de raisonnements tenus en contexte</i>	<i>142</i>
Section 1 : Raisonnements tenus en contexte	142
1.1 Tâches et activités.....	143
1.2 Raisonnements stratégiques contextuels.....	145
1.3 Conséquences pour la conception d'un SAIC	146
Section 2 : Modélisation d'activités par les graphes contextuels.....	147
2.1 Organisation d'une activité par un graphe contextuel	147
2.2 Structuration des activités.....	149
2.3 Adaptation du formalisme des graphes contextuels	150
Section 3 : Acquisition de connaissances stratégiques.....	152
3.1 Algorithme	153
3.2 Rôle de l'acquisition incrémentale de connaissances stratégiques	154
3.3 Généralisations opérées	155
3.4 différences avec l'apprentissage automatique	157
Section 4 : Graphes Contextuels et schèmes d'action	158
4.1 Modélisation d'activités et schèmes d'action	158
4.2 Pouvoir représentatif du modèle	159

Section 5 : Graphes contextuels et développement de SAIC	160
5.1 Connaissances générales.....	160
5.2 Connaissances stratégiques.....	161
5.3 Génération d'explication	162
Section 6 : Conclusion.....	164
<i>Sixième chapitre : Système d'Aide à la Gestion d'Incidents dans le Métro</i>	<i>167</i>
Section 1 : Le projet SART	167
1.1 Genèse du projet	168
1.2 Fonctionnalités.....	168
1.3 Architecture logicielle.....	169
Section 2 : L'agent gestionnaire d'incidents.....	171
2.1 Rôle de SAGIM	171
2.2 Structure logicielle de SAGIM	172
2.3 Les trois modes d'utilisation de SAGIM.....	174
2.3.1 L'archiviste	174
2.3.2 Le conseiller.....	176
2.3.3 Le courtier.....	178
Section 3 : Spécification fonctionnelle de SAGIM.....	179
3.1 Raisonnement à base de Graphes Contextuels	179
3.1.1 Principes du raisonnement	179
3.1.2 Implémentation et choix d'implémentation.....	180
3.1.3 L'algorithme d'acquisition incrémentale de connaissances.....	181
3.2 Raisonnement à Partir de Cas	182
3.2.1 Principes.....	182

3.2.2	Spécificités de notre application	183
3.2.2.1	Mesure de similarité dépendante du contexte.....	183
3.2.2.2	Gestion de grandes bases de cas redondantes	184
3.2.3	Structure d'un cas	184
3.2.4	Mesures de similarité et contexte.....	185
3.2.5	Structure de la base de cas	185
3.3	Données communes	186
3.3.1	Evolution des connaissances contextuelles.....	186
3.3.2	Evolution des connaissances sur les incidents et les événements incidentels	186
Section 4 : Prototype de SAGIM		187
4.1	Spécifications.....	187
4.2	Implémentation	188
4.2.1	Graphes contextuels et modèle d'activités.....	188
4.2.2	Formats des fichiers de données	191
4.2.2.1	Listes de données	191
4.2.2.2	Connaissances contextuelles.....	192
4.2.2.3	Base d'actions et d'activités	192
4.2.2.4	Fichier rapport d'incident	194
4.2.3	Organisation des fichiers de données.....	195
4.2.4	Interface du prototype.....	196
4.3	Résultats obtenus	198
4.4	Capacités actuelles et potentielles.....	200
Section 5 : Conclusion.....		204

<i>Conclusion générale</i>	206
<i>Bibliographie</i>	213
Références	213
Publications	227
Bibliographie complémentaire.....	229
<i>Sigles et abréviations</i>	233
<i>Annexes</i>	238
Annexe 1 : Typologie des incidents et des causes.....	239
Annexe 2 : Exemple d'un fichier d'actions et d'activités	244
Annexe 3 : Exemples de fichiers d'incidents	246
Annexe 4 : Suivi de l'acquisition d'une pratique avec SAGIM.....	247
<i>Traductions de l'introduction et de la conclusion</i>	268
Introduction.....	268
Conclusion	276

Tables des illustrations

Figures

Nous remercions la RATP et son service d'édition de nous avoir gracieusement offert les droits de diffusion liés aux deux photos que nous présentons dans les figures 2-1 et 2-2.

Figure 1–1: Contexte, connaissances contextuelles et contexte procéduralisé	53
Figure 2–1: Photo des TCO et consoles des lignes 7-7bis (console de gauche, TCO du haut) et de la ligne 1 (console de droite, TCO du bas)	66
Figure 2–2: Photo d'un détail du TCO de la ligne 2	67
Figure 2–3: Phases de gestion d'un incident.....	76
Figure 3–1 : Arbre de gestion d'un incident	90
Figure 3–2: Graphe de gestion d'un incident	93
Figure 3–3: Graphe contextuel	98
Figure 3–4: Graphe contextuel avant acquisition de la nouvelle pratique	101
Figure 3–5: Sélection de la stratégie ayant dû être appliquée	102
Figure 3–6: Appariement des actions attendues et des actions réalisées.....	102
Figure 3–7: Graphe contextuel complet après acquisition de la nouvelle pratique.....	103
Figure 3–8: Mécanisme de procéduralisation/déprocéduralisation du contexte	106
Figure 3–9: Sous-graphes contextuels.....	107

Figure 3–10: Graphe contextuel avec activités.....	108
Figure 3–11: Structuration de l'ensemble des activités	109
Figure 4–1: Cycle du Raisonnement à Partir de Cas	126
Figure 4–2: Tir d'une transition dans un réseau de Petri	131
Figure 5–1: Décomposition des tâches liées à la gestion d'un incident traction suivant le formalisme de représentation des arbres ET/OU.....	148
Figure 5–2: Graphe contextuel non-planaire de l'activité <i>évacuation d'une rame de secours</i>	150
Figure 5–3: Graphe planaire obtenu par échange de branches des nœuds contextuels et copie de parties du graphe.....	151
Figure 5–4: Graphe planaire obtenu par la modification par un expert des connaissances contextuelles impliquées dans le diagnostic.....	151
Figure 5–5: Non-planarité lors de l'acquisition d'une pratique.....	153
Figure 5–6: Graphe rendu planaire par inversion de noeuds contextuels.....	154
Figure 5–7: Généralisation au niveau de l'élaboration du plan d'actions	156
Figure 5–8: Différents niveaux d'explications possibles.....	163
Figure 6–1 : Structure de communication de SART	170
Figure 6–2: Architecture logicielle de SAGIM	174
Figure 6–3 : Interaction utilisateur / Archiviste et flux de données	175
Figure 6–4 : Interaction utilisateur / Conseiller et flux d'informations.....	177
Figure 6–5 : Interaction utilisateur / Courtier et flux d'informations	178
Figure 6–6: Graphe contextuel de la gestion de l'incident traction, version série/parallèle ...	181
Figure 6–7: Hiérarchie d'héritage des classes du prototype de SAGIM.....	190
Figure 6–8: Hiérarchie d'inclusion des classes du prototype de SAGIM.....	191
Figure 6–9: Arborescence du répertoire du prototype de SAGIM	196

Figure 6–10: Exemple d'une page de saisie de SAGIM	198
Figure 6–11: Evolution de la base de stratégies de SAGIM	199
Figure 6–12: Evolution des bases d'incidents de SAGIM	200

Tableaux

Tableau 3-1: Actions	89
Tableau 3-2: Eléments contextuels.....	89
Tableau 3-3: Macro-actions et leur équivalent en terme de séquence d'actions.....	93
Tableau 5-1: Comparatif des schèmes d'action et de structures à base de graphes contextuels	159
Tableau 6-1: Format d'enregistrement des graphes contextuels.....	194
Tableau 6-2: Capacités actuelles de SAGIM.....	202
Tableau 6-3: Capacités potentielles de SAGIM	204

Introduction

Cadre de l'étude

Le but premier des ordinateurs était de décharger l'être humain des calculs longs et fastidieux qu'il était amené à réaliser. Le premier automate de calcul opérationnel, la Machine de Pascal, était un bon point de départ pour le développement des premiers ordinateurs. Les théories de Turing sur les machines universelles, la définition de l'architecture de Von Neumann et l'évolution de l'électronique ont été les points clés du développement des technologies informatiques telles que nous les connaissons. Ce dernier nous a permis d'assister, en parallèle, à l'essor des sciences informatiques. Dans les années 1980 le développement de la branche économique de l'informatique grand public accélère celui des ordinateurs tant en nombre qu'en puissance de calcul. Les ordinateurs dépassent largement les premiers objectifs de simples machines à calculer. De nouveaux objectifs plus ambitieux sont envisagés : simuler des systèmes plus complexes tels que les interactions entre de nombreux éléments, des phénomènes sociologiques, des systèmes biologiques, des processus de décision...

Dès la fin de la Deuxième Guerre Mondiale, l'Intelligence Artificielle (IA) se développe. Son but est d'imiter les comportements intelligents. Cette branche de recherche était restée purement théorique dans la première moitié du XXème siècle. L'avènement de calculateurs universels électroniques permet de mettre en pratique les théories obtenues. L'IA, dans sa diversité, cherche soit à rendre des systèmes autonomes, soit à aider l'Homme dans ses activités quotidiennes. Pour cela, elle s'inspire de la nature humaine : raisonnements, sociologie, psychologie, structure cérébrale ; soit de la nature qui nous entoure : éthologie, interactions entre espèces, évolution... De nombreuses voies sont explorées et divers langages spécifiques accélèrent le développement de projets de recherche. Dans les années 70, les premiers systèmes experts sont mis au point. Ceux-ci sont basés sur des règles et un moteur

d'inférence. Ils permettent de prendre en considération des raisonnements humains fortement structurés, basés sur la logique propositionnelle.

La conception de systèmes experts est fortement liée aux domaines de recherche sur la cognition qui étudient l'intelligence des êtres humains. Ces sciences humaines montrent que « le raisonnement » ne peut pas être limité aux seuls raisonnements basés sur des règles tels que ceux utilisés par les premiers systèmes experts. La logique propositionnelle n'est pas des plus naturelles pour l'Homme. Il est nécessaire, pour concevoir des systèmes performants, d'assouplir cette logique en intégrant des notions plus complexes inspirées de la logique humaine. C'est ainsi que des logiques floues, modales ou temporelles ont été proposées par l'IA.

La conception de systèmes experts modernes, dits systèmes à base de connaissances, nécessite le recueil et la structuration des connaissances sur le domaine considéré et sur les raisonnements suivis par les experts. Cette phase d'analyse peut être réalisée de deux manières différentes. Premièrement, le concepteur peut définir les différentes opérations réalisées par l'expert puis analyser leurs interactions et leur structuration. Cette approche de type ingénierie est ascendante : le concepteur part du plus petit élément (les opérations réalisées par l'expert) pour aboutir aux éléments plus complexes (les activités et les règles d'action). De nombreuses méthodologies ont été développées suivant cette optique, comme KADS [Wielinga, Schreiber et Breuker, 1992], KARL [Fensel, Angele et Landes, 1991] ou DE-KART [Baudin et Pell, 1994]. Deuxièmement, le concepteur peut analyser les activités de l'expert et en déduire les actions élémentaires réalisées, ainsi que les raisonnements sous-jacents. Cette approche de type cognitive est descendante : le concepteur part des éléments les plus complexes (les activités de l'expert) pour aboutir aux plus simples (les actions élémentaires). Ces deux approches sont complémentaires, mais sont rarement utilisées conjointement. La première est plus couramment usitée par des informaticiens, la seconde par les ergonomes.

Indépendamment de l'approche choisie lors de la conception de systèmes experts, on peut souligner le rôle capital que joue le contexte dans de nombreuses situations quotidiennes pour les prises de décisions ou pour la compréhension des situations, des objets ou des discours. D'abord un constat simple sur les discours, leur contexte et la compréhension de ceux-ci : toute personne arrivant au milieu d'une discussion met un temps plus ou moins long pour saisir le sens des échanges entre les intervenants. Ce temps d'adaptation correspond à la mise à niveau des éléments nécessaires à la compréhension du discours qui ont été échangés précédemment par les interlocuteurs. Ces éléments font partie du contexte de la discussion. En ce qui concerne les objets et les situations, le contexte joue un rôle particulier, surtout quand nous cherchons à catégoriser ceux-ci. Le résultat de la catégorisation dépend fortement

du point de vue adopté et donc du contexte dans lequel cette catégorisation est réalisée. Ceci a de grandes conséquences en ce qui concerne la prise de décision. La similarité des situations dépend du contexte, et les raisonnements de type analogique doivent en tenir compte. Quant aux raisonnements déductifs, ils doivent intégrer les contextes possibles dans leurs règles pour être assez précis. Le contexte joue un rôle capital sur différents niveaux intervenants dans le processus de décision. De nombreux travaux actuels cherchent à intégrer le contexte afin de rendre les systèmes experts plus souples vis à vis des situations rencontrées.

Parallèlement aux systèmes experts, des systèmes d'aide à la décision ont été développés. Le rôle de ces systèmes est d'aider un utilisateur en vue de la réalisation d'une tâche par celui-ci. Ce sont des outils évolués mis à disposition des utilisateurs plutôt que des assistants. Guy Boy [1991] propose la conception de systèmes d'assistance intelligents. Ces systèmes sont à mi-chemin entre les systèmes d'aide à la décision et les systèmes à base de connaissances : ils reprennent le rôle des systèmes d'aide et augmentent leurs capacités avec les fonctionnalités d'un système à base de connaissances. Ils ne sont pas autonomes mais cherchent à assister un opérateur dans ses activités quotidiennes. Ainsi, ils n'ont pas la contrainte de l'exhaustivité des systèmes autonomes et peuvent acquérir des connaissances grâce à une interaction avec l'utilisateur. Ils reprennent toutefois le fonctionnement d'un système à base de connaissances, ce qui leur donne une certaine capacité de raisonnement et de manipulation des connaissances en jeu.

Les nombreuses solutions apportées par les outils informatiques sont exploitées autour de nous dans de nombreux domaines d'applications, notamment le transport (aérien ou ferroviaire, de fret ou de personnes, urbain ou interurbain), comme le prouve le succès des programmes de recherche du PREDIT. Les ordinateurs, « super-machines à calculer » ont laissé la place à des outils plus évolués. Certaines tâches automatisables ne sont plus que supervisées par des opérateurs humains et sont contrôlées par des systèmes autonomes. Les tâches non automatisables peuvent être toutefois assistées par ordinateur. Il en est ainsi pour des tâches plus complexes de régulation d'exploitations ou de diagnostic de dysfonctionnements. L'état des connaissances et techniques actuelles permet désormais de proposer des outils d'assistance pour des tâches requérant un haut niveau d'expertise humaine et peu formalisées telles que la gestion d'incidents.

Problématique

La régulation du trafic d'une ligne de métro est une tâche difficile, où chaque décision est prise en fonction de la situation actuelle. A Paris, les opérateurs chargés de cette tâche sont des experts. Ils connaissent parfaitement le système à réguler : la ligne de métro, son

exploitation, les incidents possibles, les conséquences de leurs décisions et actions... La gestion des incidents d'exploitation survenant sur la ligne est sans conteste l'activité la plus complexe de ces opérateurs. Les trains étant très rapprochés (1 train toute les 90 secondes en heure de pointe), les opérateurs n'ont que quelques instants pour élaborer une stratégie afin d'éliminer la cause de l'incident en tenant compte des connaissances plus ou moins partielles qu'ils ont sur la situation (cause de l'incident, conditions du trafic, événements extérieurs...) tout en respectant les règles en usage (réglementation ferroviaire, règles de sécurité, rôle commercial de l'entreprise...). Les performances des opérateurs sont hautement liées à leur expertise. Cette expertise est acquise en deux temps : premièrement les opérateurs suivent une formation initiale au métier de chef de régulation, qui permet d'apprendre les bases théoriques ; deuxièmement, tout au long de leur carrière à ce poste, les opérateurs continuent d'apprendre par leurs expériences, celles de leurs collègues et les sessions de retour d'expérience organisées par leur encadrement.

Cette expérience de terrain n'est pas une règle générale : d'autres réseaux sont moins contraignants et plus techniques. Tel est le cas, notamment, à Rio de Janeiro. Le réseau est de conception récente et peu dense. L'exploitation met en jeu moins de trains et laisse plus de temps aux opérateurs pour réagir. L'interface du poste de régulation est plus complexe et l'exploitant a choisi d'employer de jeunes ingénieurs pour cette activité. Ceux-ci suivent une formation initiale conséquente, mais le manque de connaissance du terrain peut poser certains problèmes.

La RATP a été l'une des entreprises phare du développement des métros de nombreuses villes de part le monde. L'expertise acquise lors de la construction des différentes lignes parisiennes permet d'aider à la réalisation de plusieurs réseaux dans des situations très différentes. Les métros ainsi construits sont plus modernes. Les entreprises qui les exploitent deviennent compétitives sur la scène internationale, voire plus efficaces. Pour conserver sa place dans le marché international de conception et réalisation de réseaux métropolitains, la RATP doit faire valoir son expérience centenaire. En effet, après un siècle d'exploitation du réseau parisien, la RATP a acquis une grande expérience sur la régulation de trafic et la gestion des incidents. La quantité d'incidents déjà rencontrés et la qualité des solutions apportées sont d'une rare importance. Pour maintenir son excellente position face à d'autres compagnies, dont le savoir-faire commence à être compétitif, la RATP doit valoriser cette expérience.

En 1996, deux responsables du poste de contrôle de Rio de Janeiro, commencèrent des thèses de doctorat en France. Ils ont ainsi permis d'initier un projet franco-brésilien pour la conception d'un Système d'Aide à la Régulation du Trafic (SART). Le but de ce projet était de réaliser un système expert coopératif pour les opérateurs chargés de la régulation du trafic

et de la gestion des incidents, pour Rio comme pour Paris. Toutefois, même si les principes généraux sont communs (la RATP a participé à la conception du métro de Rio), de grandes différences apparaissent. Premièrement, comme souligné précédemment, le rythme n'est pas aussi soutenu à Rio qu'à Paris. Deuxièmement, les opérateurs n'ont pas les mêmes types d'expertises au sens de Prince [1996] : à Paris les opérateurs viennent du terrain, ils ont une expertise de type réflexe; à Rio les opérateurs sont de jeunes ingénieurs, ils ont une expertise livresque. Troisièmement, de nombreuses installations sont spécifiques (par exemple la climatisation des stations à Rio ou le dégivrage des rails à Paris...), ainsi que certaines actions, stratégies ou procédures.

Ces différences sont plus ou moins contraignantes. La différence de type d'expertise modifie l'interface et le type d'informations à fournir par le système d'aide. Ce sont des modifications majeures du fonctionnement d'une partie du système. Les autres différences sont plus simples à gérer en concevant un système où la plupart des connaissances sont déclaratives. Toutes ces différences d'exploitation entre les réseaux de Paris et de Rio augmentent l'importance du contexte et de sa gestion par SART.

Plus généralement, SART est un système d'aide : il doit être conçu comme un système expert, mais son rôle est de soutenir l'opérateur dans ses activités quotidiennes, sans prendre de décision à sa place. SART est interactif : il doit pouvoir renseigner l'opérateur sur des points précis en fournissant des explications ciblées et pertinentes, il doit également être capable d'apprendre en interrogeant l'opérateur. Ces interactions avec l'opérateur doivent être réalisées dans un langage compréhensible par celui-ci, SART doit donc être capable de tenir compte de la situation pour fournir des explications utiles à l'opérateur.

Les opérateurs à qui s'adresse SART basent leurs actions sur le contexte et sa richesse. Les opérateurs construisent des stratégies de gestion des incidents à partir de données techniques sur l'exploitation de la ligne et des procédures, mais en les adaptant à la situation réelle. Les procédures ne peuvent pas prendre en compte toute la richesse des situations rencontrées, mais le contexte des incidents à traiter influence grandement les stratégies de gestion. Ainsi deux incidents, par ailleurs identiques, ne sont pas traités de la même manière en heure de pointe qu'en heure creuse. La pression temporelle étant moindre en heure creuses, l'opérateur peut envisager des stratégies plus longues. Les éléments du contexte tels que le type d'heure (de pointe ou creuse) qui interviennent dans le choix des stratégies sont nombreux dans le cas de la gestion d'incidents sur une ligne de métro.

La notion de contexte est prépondérante dans le processus de décision des opérateurs. Il est donc nécessaire d'intégrer le contexte dans les connaissances et les raisonnements utilisés par SART. La conception de systèmes à base de connaissances nécessite la modélisation des connaissances du domaine d'application et des raisonnements suivis par les experts. Nous

cherchons à introduire la notion de contexte au plus profond de notre système. C'est pourquoi la modélisation des connaissances du domaine autour de la notion de contexte a été réalisée par Saker [1999] et que nous proposons, dans cette thèse, de modéliser les raisonnements suivis par l'opérateur en intégrant la notion de contexte.

Approche suivie

La tâche des opérateurs étant complexe, nous proposons d'utiliser une conception descendante. Nous commençons par l'observation et l'analyse de l'activité des opérateurs. Les résultats de cette analyse donnent une structure des raisonnements suivis et de l'organisation des sous-tâches et des actions réalisées par les opérateurs. Nous avons modélisé cette structure de l'activité par des graphes, représentant les différentes façons d'atteindre un but souhaité, associés à un système de décomposition de tâche en sous-tâches. Chaque graphe utilisé représente les différentes méthodes associées à une tâche donnée. Ces différentes méthodes se distinguent les une des autres par la situation dans laquelle elles s'appliquent. Les graphes qui organisent ces différentes méthodes sont donc fortement liés à la notion de contexte et montrent l'importance du contexte dans les choix stratégiques réalisés par les opérateurs. Ces graphes sont nommés ***graphes contextuels***.

L'association des graphes contextuels et de la décomposition tâche/sous-tâche donne un formalisme de représentation de l'activité très proche de ce qui peut être observé auprès des opérateurs. De plus, nous obtenons, dans le même formalisme, différents niveaux de détail des activités et de leur organisation. Ce formalisme est bien accepté par les opérateurs car il est basé sur une expression du raisonnement qu'ils suivent en pratique. Ceci a deux principaux avantages : premièrement l'aide et les explications fournies par le système sont directement assimilables par l'opérateur et données au niveau souhaité par celui-ci ; deuxièmement l'insertion, par l'opérateur, de nouvelles méthodes pour atteindre un but ou un sous-but est facilitée.

Cette approche peut ne pas paraître originale. Les tâches génériques définies par Chandrasekaran [1986], l'approche componentielle de Steels [1992] ou la classification heuristique de Clancey [1979 ; 1985] (entre autres) proposent des mécanismes de décomposition similaires. L'originalité de notre approche se situe dans la prise en compte de l'aspect dynamique des décisions prises par les opérateurs. Contrairement aux tâches génériques où les méthodes de décomposition sont fixées une fois pour toutes lors de l'implémentation du système, nous optons pour une sélection dynamique des décompositions en sous-tâches lors de l'élaboration de la stratégie, et ce afin d'accroître la sensibilité du système face au contexte. Ainsi notre but n'est pas de concevoir un système « clés en main »

qui prétendrait connaître la solution pour chaque incident à gérer dans les différents contextes possibles, mais de proposer un système flexible construisant des stratégies possibles pour chaque incident donné dans un contexte donné ; la construction des stratégies par le système repose alors sur le modèle de décomposition des tâches en sous-tâches en fonction du contexte. La flexibilité du système est renforcée par la possibilité d'intégration de nouvelles pratiques réalisées par les opérateurs lors de la phase d'acquisition incrémentale de connaissances sur la gestion d'incidents suivant immédiatement la gestion de l'incident par l'opérateur.

Concernant notre approche quant à la conception du système, partant de l'observation des activités des opérateurs nous obtenons à la fois les actions élémentaires possibles et les raisonnements suivis pour l'élaboration des stratégies de gestion des incidents. Partant d'une approche plutôt cognitive, nous aboutissons à un formalisme de représentation des raisonnements autorisant une phase ascendante de conception. Cette deuxième phase consiste à la mise en évidence des actions élémentaires (facilitée par la première phase de conception) puis en l'organisation de ces actions en des structures représentant les différents niveaux explicatifs des activités de gestion des incidents. Notre approche est donc intermédiaire par rapport aux deux principales approches de conception de systèmes à base de connaissances. Elle se situe ainsi à la frontière entre des sciences cognitives et l'ingénierie de connaissances nécessaire au développement d'un système opérationnel.

Plan

Dans le premier chapitre, nous faisons un point sur l'évolution des systèmes experts : des systèmes de première génération basés sur des règles aux systèmes de deuxième génération basés sur les connaissances, dont nous analysons les limites. Le contexte joue un rôle capital dans de nombreuses expertises humaines, il est donc nécessaire d'en tenir compte lors de la réalisation de systèmes experts, de systèmes d'aide à la décision ou de systèmes autonomes évoluant dans un environnement complexe. Dans le cadre d'un système d'aide à la décision, le contexte joue de plus un rôle particulier pour les explications fournies à l'utilisateur. Nous montrons le rôle du contexte tant pour les prises de décision que pour les explications. Nous proposons également un modèle de systèmes experts tournés vers l'utilisateur : les Systèmes d'Aide Intelligent en Contexte (SAIC) [Brézillon, 1999a], pour pallier les carences des systèmes experts actuels.

Nous proposons par la suite d'appliquer le modèle de SAIC à un domaine particulièrement sensible : la gestion d'incident sur une ligne de métro. Le métro de Paris est un système complexe où le maître mot est « sécurité ». En effet, le métro est un système ferroviaire

souterrain de transport de voyageurs, ce qui complique la gestion des incidents et oblige à une vigilance particulière concernant l'alimentation électrique des installations et la gestion des usagers. Nous présentons, dans le second chapitre, le métro parisien et la gestion des incidents sur une ligne de métro. Nous faisons d'abord un point sur l'histoire du métro et sa structure. Nous présentons ensuite la tâche de gestion des incidents, les agents qui en ont la charge, leur activité... Nous mettons particulièrement l'accent sur l'importance du contexte dans ces activités.

Le troisième chapitre est consacré à nos travaux sur un modèle de raisonnements tenus en contexte. Ce modèle est issu de travaux antérieurs sur l'organisation de larges bases de règles de production et sur les arbres de décision. L'introduction du contexte dans les raisonnements artificiels pour les rendre plus efficace implique un changement de point de vue. Nous proposons de laisser de côté le point de vue séparant l'action du diagnostic et de prendre celui de l'imbrication du diagnostic et des actions, rendant une place centrale à l'évolution de la situation au cours du traitement. Nous présentons ainsi le modèle de graphes contextuels : leurs éléments constitutifs et leur structure. Nous montrons également quelles sont les capacités d'évolution des graphes contextuels et comment de nouveaux chemins d'action sont pris en compte grâce à un système d'acquisition incrémentale de connaissances. Nous esquissons enfin la modélisation d'activités par les graphes contextuels.

Dans le quatrième chapitre, le modèle de graphes contextuels et la modélisation des activités par ces graphes sont comparés aux divers travaux déjà effectués ou en cours. Nous analysons particulièrement les travaux apparentés au niveau des connaissances et ceux apparentés au niveau de l'implémentation. Entre autres, nous discutons des Schémas, des Frames, des Scripts, des Schèmes d'Action, des diagrammes d'influence, des réseaux de Petri, de la décomposition des tâches et des Tâches Génériques, de la Planification Dynamique... Nous étudions pour chacun d'eux les points communs et différences avec les graphes contextuels, et ce qu'apporte notre modèle. Notre approche, à première vue peu originale, se révèle novatrice à bien des égards. Ce chapitre montre les principales avancées par rapport aux travaux connexes, quels que soient leurs domaines, et fait ressortir l'aspect fédérateur de notre modèle.

Le cinquième chapitre détaille, au niveau sémantique, l'utilisation de graphes contextuels pour la représentation d'activités humaines. Nous appliquons ainsi ce modèle aux raisonnements observés chez les opérateurs concernés par le système d'aide que nous concevons et établissons ainsi un pont entre deux disciplines importantes : *l'Intelligence Artificielle* et *l'Ergonomie Cognitive*. Ce chapitre fait un point sur les raisonnements tenus en contexte, où nous discutons essentiellement du lien entre les tâches et les activités puis des raisonnements stratégiques contextuels. Nous détaillons ensuite comment les graphes

contextuels peuvent être utilisés pour représenter des activités. Nous revenons également sur l'acquisition de connaissances et précisons les conséquences de la représentation d'activités sur l'algorithme décrit dans le troisième chapitre. Nous comparons ensuite plus précisément le modèle proposé (représentant des activités par le biais de graphes contextuels) aux schèmes d'action mis en évidence par les sciences cognitives. Nous finissons ce cinquième chapitre sur les conséquences d'une telle représentation pour la conception de SAIC.

Le dernier chapitre présente le prototype issu de ces recherches et les résultats obtenus pendant la phase de validation en situation réelle. Ce système est un SAIC construit autour de l'hybridation d'un raisonnement à base de graphes contextuels et d'un raisonnement à partir de cas. Nous précisons ici les conséquences de l'introduction du contexte au sein du système pour les raisonnements implémentés et pour les connaissances manipulées. Après une section situant notre système au sein du projet SART, ce chapitre présente les fonctionnalités et la structure de notre système. Nous développons par la suite les choix d'implémentation réalisés pour les deux modes de raisonnements hybridés. Nous revenons notamment sur la grande flexibilité obtenue grâce à l'usage massif de connaissances déclaratives. Nous finissons par l'analyse des résultats obtenus en situation réelle durant la phase de validation.

Dans la conclusion, nous discutons des évolutions possibles de SART. La conception prévoit ces évolutions notamment en ce qui concerne l'application à Rio de Janeiro. A Paris, un nouveau type de poste de contrôle du métro vient d'être mis en service pour la ligne 4. Le gestionnaire d'incidents de SART a été testé sur ce nouveau poste de contrôle, moyennant quelques modifications de certains fichiers de déclaration des connaissances. Actuellement la RATP travaille à la rénovation du poste de contrôle de la ligne 13 qui pourrait devenir un autre centre de test, d'autant plus intéressant que la ligne 13 se distingue des autres lignes sur plusieurs points : la ligne a un embranchement, les deux branches sont de longueur très différente, elle porte plusieurs terminus intermédiaires... Une autre extension du système est à l'étude concernant le Réseau Express Régional (RER). Le RER se distingue du métro par la présence de nombreuses bifurcations, des trains pas forcément omnibus, une fréquence plus faible des trains, deux exploitants du réseau (la RATP et la SNCF)...

Premier chapitre :

Des Systèmes Experts aux systèmes intelligents

Les premiers systèmes experts, développés dans les années 1970, se heurtent rapidement au problème de la gestion des connaissances nécessaires à leur fonctionnement. Les systèmes à base de connaissances, ou systèmes experts de deuxième génération, sont conçus autour de ces connaissances et intègrent dès leur conception les principes permettant leur maintenance. Toutefois les raisonnements des experts ne sont pas simples à cerner et sont souvent fortement liés aux détails des situations.

Le *contexte* joue un rôle capital dans de nombreuses expertises humaines. Actuellement, plusieurs voies sont explorées pour intégrer le contexte dans des systèmes d'aide à la décision ou dans des systèmes autonomes. Nous pensons également que le contexte joue un rôle important dans les *explications* qu'un système d'aide peut apporter à un utilisateur.

Dans ce chapitre, nous revenons sur les limites des systèmes experts de première et de deuxième génération. Nous montrons ensuite l'importance du contexte puis quelques exemples de systèmes tenant compte du contexte. Pour finir, nous présentons notre point de vue sur le contexte et le modèle des *Systèmes d'Aide Intelligents en Contexte* (SAIC).

Section 1 :

Systèmes experts et systèmes à base de connaissances

L'Intelligence Artificielle cherche, entre autres, à développer des outils intelligents d'aide pour la réalisation de diverses tâches. Pour cela elle s'inspire des activités humaines et, en particulier, cherche à imiter les raisonnements humains. Parmi ceux-ci les raisonnements fortement structurés basés sur les logiques mathématiques d'ordre 0 puis d'ordre 1 sont les plus faciles à mettre en œuvre dans des systèmes informatiques. Les premiers Systèmes Experts (SE) sont basés sur ces modes de raisonnements. Partant de faits et d'une base de règles d'inférence ils appliquent les règles pour prouver de nouveaux faits. Ils avancent ainsi dans leur raisonnement. Nous commençons cette section par l'étude des limites des SE de première génération. En réponse aux limites des SE de première génération, les Systèmes à Base de Connaissances (SBC), ou SE de deuxième génération, ont été développés. Ceux-ci

traitent des connaissances plutôt que des faits et séparent les connaissances du domaine et les connaissances sur les raisonnements. Nous étudions ces systèmes, leur fonctionnement et leur conception dans la seconde partie. Les SBC acquièrent ainsi une plus grande souplesse de fonctionnement et sont de maintenance plus aisée. Toutefois ils se heurtent également à des limites que nous analysons dans la dernière sous-section.

1.1 Limite des systèmes experts de première génération

Les premiers Systèmes Experts sont fondés sur les logiques d'ordre 0 et d'ordre 1. Ils ont à leur disposition une base de règles d'inférence de type :

SI prémisses ALORS conclusions

Ces règles permettent, partant d'une base de faits, d'inférer de nouveaux faits. A chaque étape le moteur d'inférence recherche les règles dont les prémisses sont vérifiées par les faits actuellement considérés comme prouvés par le système. Il sélectionne ensuite la règle qu'il va exécuter parmi celles-ci. Cette sélection peut être opérée suivant différentes méthodes : la première règle applicable, la règle qui apporte le plus de faits nouveaux... Les conclusions sont alors ajoutées à la base de faits et le cycle recommence, jusqu'à ce qu'aucune règle ne soit applicable ou que la conclusion attendue soit vérifiée ou contredite.

Certains systèmes experts fonctionnent sur des règles de production plutôt que des règles d'inférences. La différence se situe dans la seconde partie des règles. Ainsi plutôt que d'ajouter simplement des faits dans la mémoire de travail du système, les règles de production offrent une plus grande flexibilité. La seconde partie de la règle est en effet composée d'actions, telles que ajouter ou supprimer des faits, poser des questions à l'utilisateur... MYCIN [Shortliffe, 1976], l'un des premiers SE opérationnels, est fondé sur ce type de règles de production. Ce système a été développé pour faciliter le diagnostic de maladies infectieuses sanguines rares. Pour se faire, le système pose quelques questions à l'utilisateur pour initier sa base de faits. Le système fait ensuite opérer des règles de production. L'application de ces règles permet d'obtenir de nouveaux faits, soit par inférence, soit par réponse de l'utilisateur à une question. Le cycle continue jusqu'à obtention d'un diagnostic.

Qu'elles soient d'inférence ou de production, les règles sont traditionnellement placées « en vrac » dans une grande base de règles. Il est nécessaire, à chaque étape de parcourir l'ensemble des règles afin de choisir, parmi celles qui respectent les prémisses (ou les conditions), celle qui va être appliquée (à moins que le critère de choix soit *la première applicable* au quel cas seulement une partie de la base de règle est parcourue). Ces

connaissances mises en vrac augmentent le temps de recherche, ce qui, pour des bases de taille conséquente pèse lourd sur les performances du système.

L'une des principales obligations des bases de règles est la consistance. Une base de règles n'est pas consistante dès que deux règles (au moins) sont en conflit, redondantes ou que l'une subsume l'autre [Buchanan, 1984, pp. 161-162]. Les règles ne doivent pas se contredire, sans quoi les conclusions du système pourraient être différentes suivant les critères de sélection des règles appliquées à chaque étape. La redondance n'est pas un problème en soi, mais représente un risque important si l'une des règles en cause est modifiée et pas l'autre. Ces règles pourraient alors entrer dans une relation conflictuelle.

La consistance de la base de règle est un élément clé du succès d'un système expert. Qui plus est, dans les systèmes experts capables d'évolution, il est capital de veiller à ce que la base de règles reste consistante. Ainsi l'ajout de nouvelles règles dans une base préexistante n'est pas sans risques.

De plus, la plupart des raisonnements humains sont fondés sur des logiques non monotones. C'est à dire que l'ajout d'une règle peut modifier la conclusion. Par exemple l'ajout de la règle selon laquelle une autruche ne vole pas modifie la conclusion d'un système de règles connaissant le fait que les oiseaux volent et que l'autruche est un oiseau. La non-monotonie est déjà fort présente dans les raisonnements simples que nous rencontrons tous les jours, et ses conséquences sont encore plus importantes pour la modélisation de raisonnements plus complexes, notamment ceux concernant les prises de décision.

Ces deux derniers points rendent la maintenance de grandes bases de règles très délicate.

Plus une base de règles est complète, plus le système expert étend son champ d'action. Ainsi la recherche de complétude de la base de règles est un point important dans la conception d'un système expert. Une base de règles incomplète peut être corrigée par l'ajout d'une nouvelle règle permettant d'obtenir l'inférence manquante. Malheureusement, Les règles manquantes ne peuvent être détectées *a priori* que si il est possible d'énumérer toutes les circonstances dans lesquelles une décision donnée peut être prise ou une action donnée peut être réalisée [Buchanan, 1984, p. 162].

Par ailleurs, les raisonnements des experts ciblés ne peuvent pas toujours être directement modélisés dans une base de règles de production. Ceci est particulièrement vrai pour des raisonnements peu structurés. Les raisonnements les mieux adaptés concernent les diagnostics, les interprétation de données et les recommandations suite à des dysfonctionnement, comme le montre la répartition par catégorie des Systèmes Experts opérationnels [Durkin, 1993].

1.2 Systèmes à base de connaissances

Le principal problème soulevé par le développement de SE est l'acquisition de l'expertise et sa modélisation dans un formalisme utilisable par un ordinateur. Les bases non-organisées de règles sont difficiles à maintenir car les règles ne font pas explicitement référence aux connaissances auxquelles elles sont liées. Afin de mieux organiser (et donc gérer) les connaissances nécessaires aux raisonnements impliqués, la deuxième génération de SE est construite autour de la notion de *connaissance*, d'où leur nom de Systèmes à Base de Connaissances (SBC). Les règles ne sont plus mises en vrac dans une immense base de règles, mais elles sont organisées en éléments de connaissance sur le domaine. Ces éléments rassemblent des faits et des règles de fonctionnement des différents objets pouvant être maniés par le système.

Les SE de première génération ont proposé une approche séparant les connaissances et les principes d'utilisation de celles-ci. Cette idée est complétée, dans la deuxième génération de SE, par la séparation des connaissances utilisées par le système en fonction du rôle de leurs pendants utilisés par les experts. Entre autres nouveautés, les SBC séparent les connaissances sur le domaine et les connaissances sur le(s) raisonnement(s) et définissent ainsi différents *niveaux de connaissances*.

1.2.1 Acquisition de l'expertise

L'acquisition de l'expertise est le principal problème soulevé par les systèmes experts de première et deuxième génération. L'expertise est une forme compilée de connaissances de type savoir-faire qui nécessite une introspection importante pour être explicitée par les experts concernés.

Des méthodes telles que les entretiens ou les séances collectives de créativité permettent d'interroger un ou plusieurs experts afin de faire ressortir les connaissances sous-jacentes. Toutefois ces méthodes placent l'expert hors de la situation réelle d'application des connaissances et peut biaiser l'expertise explicitée. Des méthodes d'interrogation des experts en cours de travail fait ressortir des connaissances de meilleure qualité, mais cet interrogatoire peut perturber l'expert et n'est donc pas envisageable dans de nombreux cas. L'observation directe et l'entretien *à froid* autour d'enregistrement de situations réelles (vidéo, documents utilisés, données...) permet d'obtenir une base de connaissances de bonne qualité, mais est longue et fastidieuse.

De nombreuses autres méthodologies ont été développées pour organiser le recueil et la structuration des diverses connaissances utilisées par les experts aux différents niveaux souhaités, dont la plus populaire est KADS (KBS Analysis and Design Support) [Wielinga,

Schreiber et Breuker, 1992 ; Tansley et Hayball, 1993]. Cette méthode décompose la conception d'un SBC en plusieurs phases. La plus importante de celles-ci, et qui nous intéresse plus particulièrement ici, correspond à l'analyse de l'expertise. La construction du *modèle d'expertise* organise les connaissances de l'expert en quatre couches : le *domaine*, les *inférences*, les *tâches* et le *contrôle*. Les trois couches supérieures sont regroupées dans le modèle de *tâches génériques*. Le modèle d'expertise est conçu indépendamment de l'implémentation qui peut ensuite être basée sur des règles ou tout autre formalisme.

La couche de domaine contient les connaissances statiques de domaine. C'est à dire les connaissances, sous forme de faits, concernant le domaine d'application, sur lesquelles repose l'expertise en question. Ces connaissances sont généralement organisées dans une structure présentant les concepts et les relations entre concepts.

La couche d'inférence décrit les traitements basiques pouvant être réalisés sur les connaissances de domaine. Cette couche ne donne aucune information sur la manière d'organiser ces inférences afin d'atteindre un but précis.

L'organisation des inférences est obtenue grâce aux tâches contenues dans la couche suivante. Certaines tâches peuvent avoir des structures de contrôle figées, d'autres construites dynamiquement par la couche stratégique.

La plus haute couche du modèle d'expertise contient des connaissances permettant la sélection, le séquençage, la planification et la prise de mesures compensatoires en cas d'échec des tâches du niveau inférieur. Ce dernier niveau contient des connaissances sur la sélection des buts et leur ordonnancement, sur la sélection de tâches permettant d'atteindre le(s) but(s) sélectionné(s), sur la création de structures de contrôle de tâches pour atteindre des buts complexes, sur les modes d'utilisation du SBC et enfin sur le contrôle des tâches et des inférences.

La séparation des connaissances dans ces couches de niveaux organisationnels croissants facilite la mise en exergue des connaissances en jeu et leurs rôles au sein de l'expertise. Ainsi les connaissances sont plus facilement repérables une fois le SBC construit et leur maintenance est plus facile. De plus cette méthode d'analyse et de développement de SBC permet des gains de temps considérables et autorise la réutilisation partielle de couches pour le développement d'autres SBC. En effet, pour des expertises différentes sur le même domaine, la couche de domaine et une partie de la couche d'inférences peuvent être réutilisées ; Pour des expertises similaires dans des domaines différents, les couches stratégiques, des tâches et d'inférences peuvent être partiellement réutilisées.

1.2.2 Différents types de systèmes experts

Nous pouvons considérer deux grandes classes de systèmes experts (de première ou seconde génération) en fonction de leurs relations avec les experts humains : les systèmes *autonomes* et les systèmes *interactifs* [Pomerol et Brézillon, 1996]. Les systèmes autonomes ont pour objectif de prendre des décisions ou de résoudre des problèmes sans l'aide d'utilisateurs. Pour cela ils doivent être intégrés dans un système d'information afin d'obtenir les données du problème de manière indépendante. Etant conçus pour fonctionner sans aide extérieure, les systèmes experts autonomes ne peuvent qu'exploiter les connaissances qu'ils contiennent sans compléter leurs bases de connaissances. La conception d'un tel système expert est guidée par un souci d'exhaustivité de l'expertise contenue dans les bases de connaissances. Les méthodes citées ci-dessus sont donc bien adaptées au développement de tels systèmes. De plus les raisonnements et les connaissances exploitées peuvent s'écarter plus ou moins de ceux utilisés par les experts humains sans perte de performance.

Les systèmes interactifs ont besoin de plus ou moins de données de la part de l'utilisateur. Dans ce cas, la réussite de l'intégration du système dans le cadre d'activité des utilisateurs est fortement corrélée à l'équilibre entre la charge de travail apportée par la saisie des données et l'apport et la pertinence, pour l'utilisateur, de la conclusion du système. Ces systèmes peuvent être intégrés également dans un système d'information afin d'alléger la charge de saisie des données en obtenant une part importante de celle-ci de manière automatique et transparente pour l'utilisateur.

Parmi les systèmes interactifs, nous pouvons distinguer les systèmes « figés » et les systèmes évolutifs. Certains systèmes experts interactifs peuvent continuer à apprendre tout au long de leur cycle d'utilisation. Dans ce cas les méthodes d'acquisition de connaissances permettent de recueillir une partie de l'expertise et de définir un cadre guidant l'acquisition incrémentale de connaissances au cours de phases de raffinement du système par des *utilisateurs-experts* du système. Le système peut alors être utilisé selon deux modes : un mode de pure interrogation pour des utilisateurs néophytes qui interagissent avec le système comme ils le feraient avec un expert ; un mode utilisation-maintenance avec un utilisateur expert dans le domaine permettant, au cours de l'utilisation, d'ajouter, modifier ou supprimer des connaissances dans le système [Farreny, 1985].

1.3 Limites des systèmes à base de connaissances

Les systèmes experts (de première et de deuxième génération) rencontrent de nombreux problèmes d'intégration dans le monde réel. Brézillon [1998] résume les succès et échecs de ces systèmes. Les principaux problèmes sont dus à une incompréhension entre le système

expert et son utilisateur. Ces incompréhensions peuvent être causées par la non-prise en compte du rôle de l'utilisateur face au système ou par des explications inadaptées soit au niveau de l'utilisateur, soit au problème réellement rencontré.

La plupart des systèmes experts considèrent les utilisateurs comme novices. Même si ceci se justifie au début de l'utilisation, les utilisateurs novices assidus finissent par être tout aussi *experts* que les systèmes. Cette supposition est responsable de la plus grande partie des réassignations des buts des systèmes experts, ceux-ci passant du statut d'*oracle* à celui d'*entraîneur*. Tel est le cas de nombreux systèmes experts qui ne sont désormais utilisés qu'à des fins de formation et non plus comme outil pour les experts.

Le deuxième point d'achoppement des systèmes experts est lié aux explications fournies par le système à l'utilisateur. Le plus souvent ces explications pèchent par leur absence d'adaptation à la situation rencontrée par l'utilisateur. Les explications restant trop générales, elles ne sont que de peu d'utilité pour l'utilisateur. D'autre part, et ceci rejoint un peu le premier point, les explications ne tiennent pas compte du niveau d'expertise de l'utilisateur.

Les points soulevés ci-dessus sont communs à tous les systèmes experts, qu'ils soient de première ou de seconde génération. Les SBC apportent une plus grande structuration des connaissances, ce qui simplifie la maintenance des bases de connaissances. Toutefois quelle que soit la méthode utilisée pour l'élicitation des connaissances, celle-ci part du principe que le domaine d'application est figé. Hors ce n'est pas toujours le cas. Il est vrai que pour la plupart des applications le domaine varie assez lentement pour le considérer fixe (du moins, qu'il ne changera pas durant la vie du SBC construit). Pour les domaines en évolution rapide ou dans lesquels l'expertise est en perpétuelle progression, il est nécessaire de recourir à une acquisition incrémentale de connaissances tout au long de l'utilisation du SBC. Pour des domaines où les stratégies de contrôle des tâches évoluent, ceci se traduit par la nécessité de concevoir un modèle dynamique de ces stratégies, contrairement au modèle statique fourni par les méthodes traditionnelles.

Section 2 :

Systèmes d'aide

2.1 Systèmes Interactifs d'Aide à la Décision

Toute activité humaine repose sur des décisions plus ou moins complexes. Pour prendre celles-ci les décideurs estiment les conséquences et pèsent le pour et le contre des diverses

options qui s'offrent à eux. Pour être efficaces ils doivent recueillir l'information sur la situation nécessaire à leurs choix, traiter ces informations, construire des hypothèses, estimer les évolutions possibles et finalement prendre une décision en s'appuyant sur les résultats obtenus. Les systèmes informatiques « traditionnels » peuvent aider à la récolte d'informations, mais les décideurs font alors face à un excès d'informations et ont plus de mal à mettre en évidence les plus importantes. L'Intelligence Artificielle peut alors les aider en effectuant des analyses s'appuyant sur ce large panel d'informations, grâce à de nouveaux Systèmes Interactifs d'Aide à la Décision (SIAD).

Les SIAD sont des outils dont le rôle est d'aider un utilisateur à prendre une décision en facilitant les phases de recueil d'informations, d'analyse d'options possibles et en organisant la comparaison des points forts et faiblesses de chaque solution retenue [Lévine et Pomerol, 1989]. Le principe des SIAD est d'aider les utilisateurs à prendre des décisions, et non d'en prendre à leur place. Ainsi ils ne sont pas conçus pour un fonctionnement autonome, ni pour « remplacer » les utilisateurs. Ces derniers se déchargent de certaines tâches liées à la prise de décision tout en étant guidé par le SIAD. L'interactivité du système réside dans les échanges entre le SIAD et l'utilisateur et dans leur compréhension mutuelle. Cette compréhension mutuelle est facilitée par l'utilisation de modèles communs de représentation et de traitement de l'information nécessaire à la décision. Ces modèles communs permettent à l'utilisateur de s'approprier rapidement les résultats obtenus par le SIAD. Ils augmentent également la compréhensibilité des explications fournies par le SIAD pour l'utilisateur.

Les SIAD préexistent à l'émergence des outils informatiques, les feuilles comptables entrant dans le cadre théorique des SIAD. Les systèmes précédant l'avènement des outils informatiques sont fondés sur des modèles fortement structurés, en général issus des mathématiques. L'arrivée des outils informatiques ne fait qu'augmenter les capacités des SIAD en autorisant l'emploi de modèles plus complexes. Les modèles utilisés par les premiers SIAD informatiques sont toutefois inspirés de modèles mathématiques ou issus de la recherche opérationnelle. Ainsi MYCIN se veut être un SIAD basé sur un modèle constitué de faits et de règles de production.

Désormais les SIAD sont de structure plus complexe, composés de différents modules activés en fonction du type d'information et/ou du traitement des informations. Généralement de petits Systèmes Experts dédiés à une tâche précise peuvent être au cœur des modules d'un SIAD. Les principales fonctionnalités des SIAD modernes consistent en l'organisation du traitement des informations, de la présentation de celles-ci à l'utilisateur et des échanges avec ce dernier.

2.2 Systèmes d'Assistance Intelligents

La décision n'est pas le seul domaine dans lequel des outils d'aides informatisés peuvent être utiles. Guy Boy [1991] propose le développement de Systèmes d'Assistance Intelligents (SAI). Ceux-ci reprendraient le rôle joué par des assistants dans le contrôle de systèmes dynamiques complexes. Selon l'auteur, les assistants et les opérateurs entretiennent des relations telles que :

1. L'opérateur reste seul maître à bord : il peut consulter l'assistant quand il veut, mais prend les décisions ultimes ;
2. Si l'opérateur délègue une part de sa responsabilité sur l'assistant, l'assistant considère cette délégation comme une tâche à effectuer ;
3. L'opérateur peut à tout instant décider d'interrompre l'exécution d'une tâche par l'assistant, s'il le juge nécessaire ;
4. L'assistant peut toutefois prendre des initiatives : tester des paramètres, tenir au courant de l'évolution de la situation, prédire des erreurs déductibles...
5. Un assistant doit pouvoir mettre en œuvre les connaissances incluses dans le manuel, de sa propre initiative ou sur ordre de l'opérateur ;
6. Un assistant doit être capable d'expliquer, avec une quantité de détails adaptée à l'opérateur et à la tâche en cours, les résultats qu'il a obtenu.

Les relations citées ci-dessus induisent différentes conséquences pour le développement d'un SAI. Premièrement, l'utilisateur et le système d'assistance doivent établir une coopération étroite, basée sur une forte interactivité [Brézillon et Cases, 1995]. Le SAI n'est toutefois qu'un conseiller : il n'est pas autonome et ne cherche pas à le devenir. Le SAI doit être réactif et laisser toujours la main à l'utilisateur qui peut donner de nouvelles priorités à son assistant [Fischer, 1990]. Sur ces points, l'optique SAI rejoint les SIAD. Le SAI doit, deuxièmement, être capable de réaliser des tâches de surveillance continue du processus à contrôler et informer l'utilisateur de toute déviance. Il peut alors proposer, de son propre chef, un diagnostic plus ou moins avancé, une critique des actions menées par l'utilisateur en mettant en œuvre les connaissances qu'il a sur le domaine d'application [Aamodt et Nygard, 1995]... Ces connaissances doivent également être utilisées pour la réalisation de tâches spécifiques demandées par l'utilisateur (comme cela est le cas des SBC en général). Enfin, le SAI doit être capable de fournir des explications ciblées sur la tâche et de qualité adaptée précisément au niveau d'expertise de l'utilisateur [Aamodt et Nygard, 1995].

Par ailleurs, la logique de l'utilisateur est très différente de la logique fonctionnelle liée au processus à contrôler. La logique de l'utilisateur est en perpétuelle évolution, grâce à la

diversité des situations rencontrées en pratique, surtout dans des environnements changeants. Un SAI doit être capable de prendre en compte ces évolutions des connaissances impliquées dans les raisonnements de l'utilisateur [Fischer, 1990 ; Aamodt et Nygard, 1995 ; Ekdahl, Astor et Davidsson, 1995 ; Reddy, 1996 ; Bainbridge, 1997]. Ceci est une grande différence par rapport aux SIAD et aux SBC classiques : les connaissances manipulées par un SAI sont principalement axées vers l'acquisition incrémentale des connaissances nécessaires.

Les SBC offrent toutefois un support intéressant : ils utilisent divers types de représentations et de traitements des connaissances. Ils permettent d'éditer, de modifier et d'administrer les connaissances avec une grande flexibilité (comparativement à la programmation procédurale classique). Ils peuvent remplir les conditions d'utilisation requise pour un SAI dans le sens système-utilisateur (réalisation d'une tâche, surveillance d'un processus, explications). Toutefois ils ne sont pas très doués dans le sens utilisateur-système (acquisition incrémentale de connaissances).

Selon Boy [1991], un SBC acceptant l'acquisition incrémentale de connaissances est un bon point de départ pour le développement d'un SAI. Ce point est discutable, car les SBC ne sont pas conçus pour un niveau d'interactivité aussi élevé. Ils sont incapables d'adapter automatiquement leur activité à celles de l'utilisateur. Pourtant un véritable assistant est capable de prendre ces initiatives. Un système coopératif peut, selon Frontin, Kacem et Soubie [1993], adapter son comportement en observant les actions de l'utilisateur. Le deuxième point capital est l'acquisition incrémentale des connaissances. Ceci n'est pas une simple fonctionnalité que l'on pourrait rajouter à un SBC. Il faut prendre en compte ce mode d'évolution des connaissances pour développer un système complet en l'intégrant au cœur du système.

Section 3 :

Importance du contexte

Quel que soit le domaine, on entend de plus en plus parler du *contexte*. Mais que recouvre cette notion ? Etymologiquement, le contexte est ce qui accompagne le texte, c'est à dire les non-dits qui sont toutefois nécessaires à la pleine compréhension d'un message. Le sujet n'est abordé que depuis peu en intelligence artificielle et le consensus est loin d'être totalement établi autour de cette notion. Toutefois les nombreuses définitions proposées se rejoignent sur de nombreux points [Brézillon, 1999b], notamment en ce qui concerne son rôle.

3.1 Définitions du contexte

Le contexte fait toujours plus ou moins référence à ce qui est autour du centre d'intérêt mais qui permet d'accroître sensiblement la compréhension que l'on peut avoir du problème considéré [Anderson, 1993] ou de modifier la réponse à un stimulus [Tiberghien, 1986]. Le contexte est vu comme un ensemble plus ou moins structuré d'éléments plus ou moins dynamiques, l'ensemble évoluant également dans le temps. Les modèles proposés forment un ensemble très varié allant d'une simple collection d'éléments prédéfinis à des structures de taille dynamique d'éléments définis au coup par coup.

En communication [Rastier, 1996], le contexte fait souvent référence à ce qui a déjà été communiqué. C'est une collection de connaissances acquises lors d'échanges précédents ou construites au cours de la discussion sur lesquelles les interlocuteurs peuvent s'appuyer, sachant que ces connaissances sont partagées [Maskery et Meads, 1992], afin de donner un sens à un message [Karsenty, 1994]. Chaque participant a sa propre vision du contexte de l'échange. Ceci peut aboutir à des incompréhensions quand les interlocuteurs pensent, à tort, qu'ils partagent une part de leurs contextes respectifs [Cahour et Karsenty, 1993].

Pour la compréhension du langage naturel, le contexte permet de limiter les termes possibles en supposant que les mots employés ont un rapport sémantique avec les mots précédents. Il permet aussi d'affiner les relations entre les termes lors de la recherche de la structure de la phrase et du discours. La difficulté principale dans ce domaine est que certaines propositions changent totalement de sens quand elles sont placées dans un autre contexte (ceci est expérimenté tous les jours par chacun d'entre nous). La séparation des propositions contexte-dépendantes de celles qui ne le sont pas donne d'excellents résultats [Pereira et Pollack, 1991]. Dans le domaine fortement lié qu'est la traduction automatique, le contexte joue un rôle capital pour sélectionner la traduction la plus probable parmi l'ensemble des traductions possibles [Berthouzoz, 1999]. Dans ce cas le contexte est constitué des phrases précédentes et permet d'éliminer les traductions qui casseraient le fil conducteur du discours.

Le contexte joue également un rôle important pour la vision et a donc été étudié dans ce domaine. Desvignes, Porquet et Spagnou [1989] précisent que, dans le domaine de la reconnaissance de séquences vidéo, le contexte permet de guider la recherche, de résoudre les ambiguïtés, de combler les « trous », de corriger des erreurs et d'apprendre. Le contexte est alors défini comme l'ensemble des propriétés associées à une entité en fonction de son environnement. Brémond et Thonnat [1997] définissent le contexte par la description des différents types d'informations manipulés. Ils proposent également de représenter une scène sous différents points de vue.

D'une manière plus générale, le contexte regroupe des éléments de natures diverses qui délimitent la compréhension, le champ d'application ou les choix possibles [Hasher et Zacks, 1984]. Les éléments les plus couramment invoqués concernent des données spatio-temporelles (lieu, heure, jour...) ou des connaissances spécifiques en relation avec le domaine étudié. Plus rarement nous observons l'utilisation d'éléments concernant des émotions, des états d'esprit, des données culturelles, anthropomorphiques... Ainsi certains éléments du contexte peuvent être difficiles à cerner car nous les utilisons inconsciemment, d'autres se trouvent hors d'atteinte des périphériques d'entrée des machines et donc difficiles à mettre en œuvre dans des systèmes automatiques [Degler et Battle, 2000].

Edmonds [1999] explique les bases du contexte et de ses différentes formes : la répartition causale [Wheeler et Clark, 1999] oblige à limiter le nombre de facteurs pouvant être pris en compte par les modèles (la modélisation complète d'un événement est impossible en pratique). L'omission des facteurs non-retenu est sans conséquence dans deux cas : soit les facteurs écartés sont assez fiables et leur omission n'a pas d'importance lors d'applications pratiques ; soit nous sommes capables de reconnaître quand notre modèle restreint peut être appliqué et nous ne l'utilisons pas quand il n'est pas applicable. Les facteurs non-explicitement inclus dans un modèle forment ainsi le contexte de ce modèle. Ce contexte permet de savoir quand ce modèle est applicable, c'est à dire son domaine de validité. Cette définition du contexte a le mérite d'expliquer la variabilité des définitions proposées dans les différents domaines d'application : les modèles obtenus étant différents, il est logique que le contexte soit perçu de diverses manières.

L'idée d'Edmonds sur le rôle que joue le contexte sur la limitation du domaine de validité d'un modèle rejoint celle de McCarthy [1979] quand il considère que les contextes sont des objets riches dans le sens où ils ne peuvent pas être entièrement décrits. Par contre ce dernier, cherchant à introduire des éléments du contexte dans des modèles opératoires¹, rend explicite les facteurs précédemment implicites liés aux propositions contenues dans le modèle considéré.

3.2 Rôle du contexte dans un système d'aide

Dans un système d'aide, le contexte intervient à plusieurs niveaux [Brézillon et Pomerol, 1999]. L'interactivité de tels systèmes est une cause importante d'intervention du contexte car le système doit être capable de comprendre l'utilisateur, ses questions et ses attentes. Ainsi le contexte joue un rôle important au niveau des explications et des interactions

¹ Nous n'exposons pas ici le modèle de McCarthy, sur lequel nous revenons dans la section 4.

système/utilisateur visant à l'élaboration de ces explications. Le contexte est également important, pour certains domaines soumis à une grande variabilité, pour limiter le champ de validité des connaissances et des raisonnements impliqués [Wielinga et Schreiber, 1997].

3.2.1 Rôle du contexte au niveau des connaissances

De nombreux faits et connaissances ne sont vrais que dans certains contextes ou changent de sens en changeant de contexte. Par exemple « je mange » est vrai dans certains contextes (je ne mange pas tout le temps). L'affirmation « je pense » a une relation plus subtile avec le contexte. Hors de tout contexte ou dans un contexte général, cette affirmation a pour sens : « je suis un être capable d'une certaine réflexion ». Dans des contextes plus précis tels que ceux faisant référence aux activités, cette affirmation se rapporte plutôt à l'action de penser et est traduite par « mon activité principale est actuellement de penser ». Dans un contexte d'échanges d'opinions, cette affirmation est généralement synonyme de « mon avis est que... ».

Le contexte permet ainsi de définir le sens à donner à certains faits ou connaissances à chaque instant, en mettant de côté les connaissances qui ne sont pas vérifiées dans ce contexte ou les sens inadaptés. Le contexte permet, à chaque instant, de définir quelles connaissances peuvent être considérées, quelles sont leurs conditions d'activation, quelles sont leurs limites de validité et quand les utiliser [Bastien, 1992].

Dans PROTEGE [Walter, Eriksson et Musen, 1992], les connaissances sont organisées par composants, chacun d'eux étant définis par rapport à un contexte intervenant sur 4 niveaux. Le premier niveau d'intervention du contexte consiste en la définition du rôle du composant pendant l'assemblage, définissant ainsi ce que le composant peut réaliser. Le contexte décrit également les connaissances requises par le composant, définissant ainsi les ressources qui doivent être disponibles. Le troisième niveau spécifie les besoins en entrée et sortie du composant. Enfin un dernier niveau de contexte ajoute des éléments permettant de réutiliser ou de partager le composant dans un autre système. Ce système a l'avantage de bien délimiter le champ de validité des composants ainsi construits autour des connaissances et permet de mieux gérer celles-ci en fonction de la situation.

3.2.2 Rôle du contexte au niveau des raisonnements

Le contexte est également un point clé des raisonnements impliqués. Les raisonnements par analogie sont, *a priori*, bien équipés pour faire face au contexte. Ceux-ci consistent en la comparaison d'une situation donnée à des situations déjà traitées, afin de déterminer leurs points communs et différences. Partant de ces résultats, ces raisonnements proposent des solutions par adaptation des solutions passées les plus proches. Le Raisonnement à Partir de

Cas (que nous détaillons dans le quatrième chapitre, section 3, page 125) est l'exemple type de cette catégorie de raisonnements. Le principe même de ces raisonnements fait référence aux situations et à leurs similarités. Il est intrinsèquement lié au contexte. Toutefois Le contexte n'intervient pas seulement au niveau de la description des situations, mais également dans la recherche de similarité entre deux situations.

Illustrons ceci par un exemple simple. Etant donnés trois objets : une poire, une orange et une balle de tennis, nous cherchons la similarité entre ces objets. Premièrement, dans un contexte faisant référence à la comestibilité, il est évident que la poire et l'orange seront très proches et la balle de tennis loin des deux fruits. Au contraire, dans un contexte faisant référence à la forme, l'orange et la balle de tennis seront très proches et la poire assez éloignée des deux sphères. Cet exemple simple met en évidence l'intervention du contexte au niveau du raisonnement même. Une intervention similaire du contexte est également observable pour les problèmes de classifications, qui sont très proches de cette phase du raisonnement par analogie, comme Tijus [2001b] le montre au sujet de la catégorisation d'objets en fonction des leurs propriétés.

3.2.3 Rôle du contexte lors de l'acquisition des connaissances

Nous avons vu que les connaissances (faits et règles) ont, dans certains cas, des domaines de validité limités par la situation. Il est donc important d'attacher à chacune d'elles les contextes dans lesquels elles sont pertinentes. Le contexte joue, en outre, un rôle particulier lors de l'acquisition des connaissances. En effet, les experts ont en général des difficultés pour expliquer le cheminement exact de leurs raisonnements, car de nombreux points de celui-ci sont des automatismes. Par contre, les experts ont plutôt tendance à expliquer une décision en la justifiant par rapport à son contexte [Jansen, 1993]. La difficulté de l'acquisition de connaissances en vue du développement d'un SBC vient du décalage entre l'explication, par les experts, de leurs propres décisions faisant référence aux contextes de ces décisions et l'extraction du raisonnement hors-contexte en vue de l'intégration dans le système.

Le contexte complique alors l'extraction des connaissances, mais il est important de faire la part des choses et de repérer les connaissances vraies quelque soit la situation de celles qui sont sensibles au contexte.

3.2.4 Rôle du contexte au niveau des explications

Selon Leake [1992], les explications sont nécessaires quand il y a un conflit entre un événement et les événements attendus par l'utilisateur (le modèle que l'utilisateur a des événements). Les conflits de ce genre sont une propriété intrinsèque des interactions entre les événements et le contexte. Un même événement (ou absence d'événement) peut être normal

ou inattendu en fonction de la situation et des processus en cours. Quand un tel conflit de croyance survient, les explications doivent le résoudre en identifiant les éléments ayant poussé l'utilisateur à cette attente.

Dans les paragraphes précédents, nous avons souligné que les connaissances sont plus ou moins sensibles au contexte. Les connaissances sensibles au contexte peuvent ainsi être attachées à diverses explications, le contexte d'utilisation de ces connaissances opérant une sélection de l'explication adéquate [Suthers, 1993]. Un tel système ne prend en compte que le contexte du raisonnement interne et reste toutefois peu sensible au contexte de l'utilisateur.

Schmidt [1995] souligne que le contexte de l'utilisateur et surtout son évolution doivent être pris en compte pour la fourniture d'explications. Il prend l'exemple d'un distributeur automatique (interface homme-machine réduite à sa plus simple expression). Le concepteur a le choix de la quantité d'informations mise à disposition de l'utilisateur : plus le système fournit d'explications, moins l'utilisateur a besoin de savoir-faire. Pourtant, à force d'utiliser le système, un utilisateur acquiert le savoir-faire nécessaire et n'a donc plus besoin des explications. Cet exemple montre que les systèmes doivent adapter leurs explications aux connaissances et savoirs-faire de l'utilisateur (son contexte) et à l'évolution de ce contexte de l'utilisateur au fur et à mesure de l'interaction avec le système.

Même dans des systèmes plus évolués capables d'adapter les explications en fonction des connaissances employées, une incompréhension entre l'utilisateur et le système peut survenir. Le raisonnement du système diffère souvent de celui des utilisateurs. Les deux acteurs peuvent avoir des interprétations différentes des faits et des connaissances impliquées dans l'étape courante du processus. L'incompréhension qui s'en suit provient du décalage entre les connaissances de l'utilisateur et celles du système et persiste après une tentative d'explication. Mark [1988] précise que, lors de coopérations entre êtres humains, dans un tel cas d'incompréhension des explications, les personnes concernées ont tendance à fournir leur compréhension courante en attendant une correction éventuelle de celle-ci par leur interlocuteur. Cette idée, appliquée à une coopération homme/machine se traduit par le fait de préciser au système ce que l'utilisateur a compris. Fort de ces informations, le système peut ainsi être guidé pour l'élaboration d'explications ciblées en tenant compte du contexte de l'utilisateur et de son évolution. L'utilisateur et le système sont alors dans une logique de co-élaboration des explications afin de réduire l'écart entre leurs compréhensions respectives de la même situation [Karsenty et Brézillon, 1995]. Cette co-construction des explications se base sur un contexte de coopération utilisateur/système pour rapprocher les contextes internes du système et de l'utilisateur. Dans une telle coopération, l'utilisateur doit également être informé de l'état du système concernant l'élaboration des explications. Le système doit donc être capable de fournir des explications sur ses propres explications, notamment en utilisant le

contexte d'explication [Lester et Porter, 1991]. Ce contexte fait référence aux connaissances impliquées dans les explications et aux raisonnements appliqués pour la génération de celles-ci [Garfinkel, 1981].

3.3 Dualité prescrit/réel et contexte

La dualité entre le prescrit et le réel n'est pas un phénomène isolé. Leplat [1985] étudie cette question et montre comment les employés adaptent ce qui a été développé par les bureaux des méthodes. Nous détaillons maintenant ce que nous entendons par procédure, stratégie et pratique et comment ces notions interagissent.

3.3.1 La notion de procédure

La gestion de systèmes dynamique est complexe : il faut connaître les tenants et aboutissants de chaque action susceptible d'être réalisée, les règles sécuritaires, les contraintes techniques des systèmes mis en jeu, les contraintes temporelles... Pour faciliter le travail des opérateurs chargés de la gestion de systèmes dynamiques, les responsables de l'exploitation de ces systèmes, parfois aidés par les concepteurs, réglementent ces activités. Les règlements prennent, le plus souvent, la forme de *procédures*.

Les procédures sont des fiches techniques présentant les actions à réaliser pour atteindre un état souhaité du système à partir d'un état initial. Elles encadrent les activités des opérateurs en dirigeant l'attention de ceux-ci sur les éléments critiques du système à surveiller, en indiquant quelles actions réaliser en fonction de différentes conditions sur l'état actuel du système, en dirigeant éventuellement l'opérateur vers une autre procédure... Les actions prescrites par les procédures respectent au pied de la lettre les règles de sécurité relatives au système et les contraintes techniques et temporelles. Quand une procédure relative à un état donné existe, celle-ci est conçue pour être applicable dès que cet état est rencontré, quelle que soit la particularité de la situation.

Les systèmes actuellement rencontrés dans les diverses activités industrielles sont de plus en plus complexes et on observe une multiplication et une spécialisation des procédures. Les activités liées à des risques sécuritaires importants sont également les plus réglementées. Il en est ainsi pour le contrôle de centrales nucléaires, le pilotage d'avions, les systèmes de transport de voyageurs... La multiplication des procédures a toutefois ces limites : l'opérateur peut être submergé par toutes ces réglementations. De Brito et Boy [1999], travaillant sur la gestion d'incidents aéronautiques, précisent que les pilotes préfèrent reconstruire une stratégie à partir d'une procédure plutôt que de l'appliquer à la lettre, ceci pour rester bien attentif aux particularités de la situation. De plus, cette attention permet de rester actif face à l'évolution du système et de ne pas se laisser entraîner par la procédure vers une situation non-souhaitée.

3.3.2 Stratégies et pratiques

Les alternatives s'offrant aux opérateurs sont multiples. De nombreuses solutions sont possibles pour les différents problèmes qu'ils rencontrent. Les opérateurs construisent, consciemment ou inconsciemment, une *stratégie* pour gérer le problème auquel ils font face. Ici la stratégie correspond au choix et à l'ordonnancement des diverses actions possibles permettant d'atteindre le but souhaité [Saint-Sernin, 1968]. L'établissement d'une stratégie réalise deux types de choix en même temps. Premièrement, connaissant le but souhaité, elle établit des buts intermédiaires et sélectionne ainsi des groupes d'actions possibles permettant de passer d'une étape à la suivante. C'est la partie « ordonnancement des actions » réalisée par la stratégie. Deuxièmement, pour chaque groupe d'actions sélectionnées, la stratégie choisit une action particulière, la mieux adaptée possible au cas considéré, pour atteindre le but intermédiaire souhaité. Le résultat obtenu est une structure d'actions à réaliser pour atteindre le but final, définissant plusieurs *chemins d'actions* en fonction de la situation.

La structure d'actions ainsi obtenue doit ensuite être instanciée en fonction de la situation réelle. Par exemple, quand la stratégie dit « prendre contact avec le conducteur de la rame endommagée », l'instanciation correspondra à « appeler le conducteur de la rame 401 à Châtelet par la téléphonie haute fréquence ». Cette opération rend opérationnelle la stratégie proposée et construit une *tactique*. L'instanciation de la stratégie est immédiate chez les opérateurs. Aussi ne perçoit-on pas la tactique. La tactique est ensuite appliquée au problème en cours en choisissant le chemin d'actions instanciées.

Toutefois, après coup, nous pouvons avoir accès aux actions réellement entreprises. Nous appelons *pratique* la séquence d'actions effectivement réalisée par l'opérateur pour résoudre son problème. La pratique est de la même nature que l'application de la tactique : c'est une séquence d'actions instanciées en fonction d'une situation particulière. La pratique est souvent différente de l'application de la tactique initialement construite, car certaines actions peuvent modifier des éléments de la situation ou en dévoiler d'autres, remettant en cause la tactique prévue voire la stratégie elle-même. Nous sommes en fait confrontés à une planification dynamique des opérations en fonction de la situation et de son évolution pendant le traitement de celle-ci.

3.3.3 Rôle du contexte dans la mise en pratique des procédures

L'adaptation des procédures à la situation réelle n'est pas un cas isolé. En réalité, les opérateurs ne sont pas contraints par les procédures. Celles-ci ne sont données qu'à titre indicatif et sont générales. Elles ne considèrent pas toutes les finesses possibles des situations réellement rencontrées. Les opérateurs sont en réalité guidés par les procédures (qu'ils connaissent), leur expérience et leurs collègues présents à leurs côtés (en général, un opérateur

n'est pas seul face à un système complexe). Leur expertise est mise à contribution pour adapter le prescrit au réel [Hatchuel et Weil, 1992], afin de gérer au mieux la situation en respectant les règles inflexibles de sécurité élémentaire, les contraintes techniques ou temporelles... Les procédures étant générales, cette adaptation permet d'améliorer sensiblement les résultats obtenus : il est souvent plus efficace d'adapter la procédure à la situation réelle, car certaines étapes peuvent être simplifiées.

Le contexte influence alors l'opérateur de plusieurs manières. Premièrement, le contexte dirige la perception de l'opérateur : certains éléments seront mis en avant dans certaines conditions et passeront inaperçus dans d'autres. Deuxièmement, les éléments du contexte identifiés sont le support, consciemment ou inconsciemment, des décisions qui sont prises. Enfin, les résultats possibles des actions dépendent du contexte.

Les opérateurs préfèrent ainsi gérer une situation en adaptant les procédures à la réalité, afin de rester dynamique face au système. Cette adaptation est d'autant plus nécessaire que la gestion d'un système est régie par une logique d'utilisation, alors que les procédures sont établies sur une logique de fonctionnement [Richard, 1983 ; De Terssac, 1992]. Pour chaque état nécessitant un traitement de leur part, les opérateurs reconstruisent, plus ou moins consciemment, une stratégie spécifique à la situation, puis réalisent une suite d'actions que nous percevons comme une pratique. Les pratiques sont construites par l'opérateur, grâce à sa connaissance du système contrôlé, à son expérience et à son expertise, en adaptant les procédures à la situation réelle, en tenant compte de la richesse de celle-ci.

Le passage des procédures aux stratégies puis aux tactiques, que nous ne percevons qu'au travers des pratiques, est un processus continu de contextualisation des procédures. Les procédures sont très générales. L'opérateur établit alors un plan d'action en fonction des éléments de la situation qu'il connaît. Ces éléments (soit directement liés au problème, soit contextuels) permettent d'établir les grandes lignes de la gestion de la situation puis de les raffiner en un plan d'actions. Ce plan est ensuite instancié en fonction de la situation précise (instanciation des paramètres des actions, notamment) pour fournir une tactique prévue. La réalisation de cette tactique peut faire ressortir des éléments de la situation précédemment inconnus ou ignorés et les mettre en avant, ce qui implique une remise en cause de la tactique voire de la stratégie. Une fois la situation redevenue normale, l'opérateur remonte difficilement à la tactique et à la stratégie, qui plus est quand celles-ci ont évolué en cours de traitement. La seule trace vraiment fiable de la contextualisation des procédures est la pratique mise en œuvre. Celle-ci est hautement contextuelle : elle contient une séquence d'actions instanciées associées aux choix réalisés et aux éventuelles reconsidérations stratégiques et tactiques.

3.3.4 Rôle du contexte dans la procéduralisation des pratiques réelles

Pour des systèmes dynamiques utilisés depuis assez longtemps pour avoir un retour d'expérience considérable, nous pouvons assister à une deuxième phase d'établissement de procédures. Cette nouvelle phase consiste en un enrichissement des procédures par la considération d'éléments des situations couramment rencontrées en pratique.

Cette opération s'appuie sur les pratiques observées, sur leur régularité et leur adéquation avec le système contrôlé et les règles de sécurité. Elle consiste à rechercher les éléments communs à différentes pratiques réalisées pour des situations concernant un même état du système, puis à analyser ces points communs du point de vue des règlements. La généralisation de pratiques passe par une décontextualisation de celles-ci. Les actions instanciées sont remplacées par leurs versions paramétrées, les éventuelles reconsidérations en cours d'exécution sont, pour la plupart, remises en question (car elles correspondent en général à des connaissances incomplètes de la situation), les choix sont uniformisés... Le résultat obtenu est une procédure plus fine que la procédure initiale car elle comporte plus de plans possibles et prend en compte un plus grand nombre d'éléments de la situation.

Section 4 :

Systèmes tenant compte du contexte

Plusieurs travaux de recherche sont ou ont été réalisés pour replacer le contexte au centre des raisonnements artificiels. Nous présentons ici quatre exemples de nature et de conception très différentes. Nous étudions ainsi le modèle de McCarthy permettant d'introduire le contexte dans les systèmes de logique du premier ordre en activant ou désactivant les règles de production en fonction d'éléments du contexte. Nous détaillons ensuite le système *ORCA* de Turner implanté au cœur d'un système de contrôle de submersibles automatiques. Nous étudions ensuite le point de vue de Gonzalez sur le « Context-Based Reasoning ». Enfin nous présentons brièvement un domaine de recherche émergeant sur les « Context Aware Systems ». Le choix de ces quatre exemples est guidé par leur complémentarité tant au niveau de leur prise en compte du contexte que de la modélisation et de l'utilisation qui en est faite.

4.1 Le formalisme de McCarthy

Le premier exemple que nous exposons ici correspond à l'introduction du contexte dans des systèmes de logique du premier ordre par McCarthy [1979]. Le principe est basé sur l'introduction de relations basiques $ist(c, p)$, signifiant qu'une proposition p est vraie dans

le contexte c , et de fonctions $value(c, term)$, calculant la valeur de $term$ dans la contexte c . Les contextes ainsi définis sont des objets abstraits. La relation ist est elle-même définie dans un contexte c' , appelé *contexte externe* à la relation. Ainsi les contextes sont définis par rapport à d'autres contextes, et ce jusqu'au niveau de détail nécessaire à la définition du modèle. Suivant les modèles construits, ces contextes sont soit riches, s'ils ne peuvent pas être décrits complètement, soit pauvres s'ils peuvent l'être.

Les contextes interviennent à deux niveaux dans la définition du modèle. Premièrement, les contextes permettent de délimiter le domaine de validité des propositions. Un exemple de McCarthy [1993] montre ce rôle du contexte : l'auteur souligne que, dans un contexte externe de la vie courante, chacun peut délimiter les domaines de validité des propositions suivantes. Dans le contexte des « histoires de Sherlock Holmes », la proposition « Holmes est un détective » est un axiome reconnu ; alors que dans le contexte « Histoire du droit Américain », « Holmes est un juge de la Court Suprême ». Sans l'explicitation du contexte, il serait impossible de définir qui est Holmes. Le Deuxième niveau d'intervention du contexte est en rapport avec le sens à donner à des proposition (et par la même, peut influencer sur la véracité de ces propositions). McCarthy [1993] donne l'exemple de la proposition $at(jmc, Stanford)$ qui peut avoir deux sens suivant le contexte : soit elle signifie que John McCarthy est en poste à l'Université de Stanford, soit qu'il y est physiquement.

L'introduction du contexte dans le calcul des valeurs des termes non seulement modifie le résultat (comme dans le cas de $at(jmc, Stanford)$), mais aussi l'espace des valeurs admissibles. Par exemple $setof-wives(Holmes)$ est un terme dont les valeurs possibles sont soit des femmes réelles dans le contexte de « l'Histoire du droit Américain », soit des femmes fictives dans le contexte des « histoires de Sherlock Holmes ».

McCarthy [1993] définit plusieurs relations entre différents contextes permettant ainsi de transférer des propositions définies par rapport à un contexte en des propositions définies dans un contexte relatif. Cette opération est appelée *lifting*. Les relations entre contextes sont la spécialisation par rapport à un élément du contexte (le temps, le lieu, les orateurs et auditeurs...), la supposition de propositions, la spécialisation générale, la décomposition de systèmes complexes en ses parties... Ces relations permettent de définir des règles de transfert de propositions d'un contexte à un autre et donc de raisonner sur les contextes mêmes. La plupart des règles obtenues sont non-monotones, ce qui introduit des conséquences intéressantes dans un système de logique du premier ordre. En effet, les raisonnements humains sont rarement monotones, alors que les systèmes de logique le sont. Les règles de transfert de propositions d'un contexte à un autre introduisent des non-monotonies telles que celles rencontrées dans les raisonnements humains.

Les premiers contextes et propositions définis dans le modèle construit le sont par rapport à un contexte externe commun implicite qui correspond au contexte externe du modèle et délimite le champ de validité de celui-ci. Quand le modèle est appliqué dans une autre situation, il peut être invalidé. Il devient alors nécessaire de rendre explicite le contexte externe précédemment implicite. Cette opération est appelée *transcendance*. Elle est utile pour relâcher des suppositions implicites en vue d'obtenir un modèle plus général. Les propositions ainsi obtenues sont alors définies par rapport à un nouveau contexte externe implicite. Cette opération peut être répétée autant que nécessaire. Nous retrouvons là le problème de la dimension infinie du contexte.

Le formalisme de McCarthy est une première étape dans l'intégration du contexte dans des systèmes de raisonnement formels. Les raisonnements fondés sur le transfert de propositions d'un contexte à un autre permettent d'obtenir des raisonnements non-monotones plus proche de la réalité de raisonnements humains. Cependant, le passage d'un contexte à un autre reste difficile à mettre en œuvre en pratique.

4.2 Le système *ORCA* de Turner et les schémas contextuels

Le deuxième exemple que nous proposons est quant à lui orienté vers une application directe. Il s'agit du système *ORCA* de Turner [1995] de contrôle de véhicules sous-marins autonomes. Partant du constat que la plupart des applications de l'IA sont lentes et peu performante face à la diversité des situations réelles, l'auteur place le contexte au centre du système de contrôle. Le but n'est pas d'intégrer le contexte dans un système de contrôle, mais de construire le système de contrôle autour du contexte. Pour cela il explicite les caractéristiques environnementales, celles liées aux missions ou internes à l'agent qui ont une capacité d'influence sur son comportement.

Le comportement du système est fondé sur le principe qu'un agent doit avoir des connaissances explicites sur les situations qu'il peut rencontrer, afin d'adapter son comportement à celles-ci (*Context-Mediated behavior* [Turner, 1999]). Pour atteindre ce but, Turner [1989] propose l'utilisation de schémas contextuels (c-schemas) qui contiennent chacun des *connaissances descriptives* sur un contexte précis et des *connaissances prescriptives* sur le comportement approprié de l'agent dans ce contexte. Les connaissances descriptives permettent de savoir dans quelles conditions le c-schema correspondant peut être appliqué et quelles sont les connaissances qui sont vraies dans ce contexte. Par exemple, si le système remarque que la profondeur est faible et que l'eau est très peu agitée, il peut conclure qu'il se trouve probablement dans un port et donc que le risque de croiser des bateaux est plus grande (et donc que le système doit anticiper d'avantage cette éventualité). Ces connaissances peuvent également lever l'ambiguïté sur certains éléments perçus. Les connaissances

prescriptives modifient le comportement. Des actions sont autorisées ou interdites, d'autres sont réalisées différemment en fonction des contextes. La planification des actions peut également être dépendante du contexte. Ces connaissances orientent aussi la perception vers des éléments non-identifiés précédemment (car jugé peu important dans les contextes précédent). Enfin la gestion des événements est également gérée, entre autres, par les connaissances prescriptives des c-schemas.

Les c-schemas sont organisés dans une hiérarchie de généralisation/spécialisation. Cette hiérarchie facilite grandement la recherche des schémas correspondants à la situation rencontrée. A chaque modification importante de l'environnement, le système de contrôle compare les caractéristiques de la situation et les descriptions des contextes des c-schemas, puis combine les c-schemas retenus pour obtenir une ligne de conduite la mieux adaptée possible à la situation. Ce mécanisme de combinaison de schémas est intéressant car il permet d'obtenir une ligne de comportement spécifique à une situation relevant de plusieurs situations connues. Ainsi quand le système se trouve dans un port et que ses batteries sont faibles, il peut adopter un comportement ad-hoc bien que le c-schema adéquat n'existe pas. La combinaison des c-schemas « dans un port » et « batterie faible » suffisent pour permettre à l'automate de conclure que la profondeur est faible et qu'il peut se poser sur le fond en attendant d'être récupéré. La combinaison de c-schemas peut faire apparaître des conflits quand des informations sont contradictoires. Plusieurs approches sont alors possibles : soit on ne fait rien et on laisse le raisonnement régler le conflit ; soit on conserve le conflit en gardant en mémoire un ensemble de valeurs, laissant ainsi le soin à l'agent de choisir parmi celles-ci si besoin est ; soit on conserve la meilleure information (celle avec le plus grand coefficient de certitude) ; soit on combine les valeurs. La sélection de l'attitude à avoir lors de conflit dépend de la nature des informations en conflit et s'opère donc dynamiquement chaque fois que ceci est nécessaire.

Une fois qu'une ligne de conduite est adoptée, le système de contrôle ne s'inquiète plus du contexte tant que la situation ne change pas notablement. Les éléments contenus dans le c-schema suffisent à adapter convenablement le comportement de l'automate. Les principaux avantages de cette approche sont rappelés par Turner [1998] :

1. Le système est efficace : la recherche du contexte ne se fait que lors de changements importants de la situation. Dans les autres systèmes, le contexte intervient sans cesse dans les processus de décision, ce qui ralentit le système.
2. La ligne de conduite est adaptée au contexte et peut même contenir des « réflexes » appropriés, ce qui rend le système particulièrement réactif en minimisant les processus de décisions souvent longs et fastidieux.

3. Les c-schemas peuvent contenir des informations permettant de mieux comprendre des éléments de l'environnement, de focaliser l'attention sur des éléments importants dans ce contexte, d'aider à la prédiction d'événements inattendus, de préciser le sens de concepts sensibles au contexte...
4. La représentation explicite du contexte permet aux agents d'ajuster leurs connaissances contextuelles en fonction des expériences passées. Elle autorise aussi la manipulation de contextes originaux obtenus par combinaison de contextes proches.

Le modèle des c-schemas est très général et peut être appliqué dans de nombreuses applications concernant des agents autonomes. Le système ORCA est une application particulière des c-schemas pour des véhicules sous-marins autonomes. Il représente une implémentation effective des idées soutenues et est parfaitement opérationnel.

4.3 Context-Based Reasoning²

Le *Context-Based Reasoning* (CxBR) est un modèle de raisonnement basé sur le contexte introduit par Gonzalez et Ahlers [1994]. Les auteurs travaillent sur les structures de contrôle de plates-formes autonomes intelligentes (PAI) et sur l'adaptation de leur comportement au contexte. Sur ce point leurs travaux sont proches de ceux de Turner présentés ci-dessus. Nous retrouvons ainsi une approche du contexte comme cadre dirigeant le comportement et l'interprétation de la situation.

Les structures de contrôle des PAI gèrent une séquence continue de contexte qui changent quand la situation change. Le comportement d'un PAI est contrôlé par le contexte actif à cet instant, qui peut être différent d'un PAI à un autre dans une même situation (à cause des spécificités des missions, des capteurs et des capacités de chacun d'entre eux). A chaque instant au moins un contexte est actif. Quand plusieurs le sont, ils doivent être compatibles et l'un d'eux correspond plus précisément au centre d'intérêt courant. Les contextes sont définis sur des intervalles de temps et peuvent définir des transitions vers des buts ou être des buts en soi. Un nombre limité d'événements est autorisé pour chaque contexte. Ceci permet de limiter l'espace de résolution d'un problème. L'avènement d'un nouveau contexte modifie plus ou moins le cours des actions ou les attentes de résultat de celles-ci. Le nouveau et l'ancien contextes peuvent soit être mutuellement exclusif (dans quel cas le nouveau contexte élimine le contexte précédent), soit non-mutuellement exclusifs (dans quel cas le nouveau contexte recouvre le contexte précédent puis rends la place à celui-ci quand il n'est plus actif).

² Nous conservons la terminologie anglo-saxonne car celle-ci fait référence à une approche spécifique.

Les auteurs utilisent des Scripts [Schank et Abelson, 1977] (que nous détaillons dans le quatrième chapitre, section 2.3, page 119) pour représenter les contextes. Ainsi le modèle de chaque contexte contient des champs fixes, des champs variables, des procédures et des règles. Les champs fixes décrivent les connaissances statiques dans ce contexte (capacités, types et champs d'action des capteurs...). Les champs variables contiennent des variables du PAI (position, vitesse...). Les procédures décrivent des commandes de base du PAI. Les règles sont de trois types. Premièrement, les règles sentinelles : elles surveillent continuellement les données à l'affût de facteurs pouvant modifier la situation. Elles sont à la base de la *situation-awareness*. Deuxièmement, les règles de transition : elles réagissent à une situation identifiée par les règles sentinelles et déterminent quels contextes peuvent être activés. Troisièmement, les règles internes : elles servent à prendre des décisions dans le cadre d'un contexte.

Le CxBR [Gonzalez & Ahlers, 1995] est fondé sur trois types de contextes : le contexte de mission, les contextes majeurs et les sous-contextes. Le contexte de mission définit les objectifs et les contraintes de la mission. Ainsi ce contexte est fixé une fois pour toute en début de mission et restera actif jusqu'à la fin de celle-ci. Les contextes majeurs encadrent le comportement stratégique des PAI. Ils contiennent des informations sur le contrôle du PAI et sur les règles de désactivation de ce contexte et sur les contextes pouvant les suivre. A chaque instant un et un seul contexte majeur est actif. Les sous-contextes encadrent le comportement tactique du PAI. Ces contextes ne sont pas mutuellement exclusif et ne sont pas nécessaire à chaque instant. Ils délimitent plus précisément le comportement en fonction de détails de la situation. Cette approche distinguant trois niveaux de contexte est originale. Elle permet de décrire le contexte de manière opérationnelle : le contexte de la mission définit les objectifs, le contexte majeur dirige les raisonnements stratégiques et définit les buts à atteindre en fonction de la situation, et enfin les sous-contexte restreignent, si nécessaire, les actions du PAI pour les adapter à des situations particulières.

4.4 « Context Aware Systems »³

Le dernier exemple choisi correspond à un domaine de recherche émergent : les *context-aware systems*. Ceci regroupe divers types de systèmes et d'objets qui adaptent leurs

³ Nous gardons ici la terminologie anglo-saxonne car la traduction française, qui se situe entre *systèmes sensibles au contexte* et *systèmes conscients du contexte*, ne reflètent pas le sens réel de cette expression : la première traduction efface certaines caractéristiques importantes (ces systèmes ne se contentent pas d'une simple prise en compte du contexte) ; la seconde est, quant à elle, trop optimiste.

fonctionnalités et/ou leurs comportements à leur environnement ou leur état interne. Ceci concerne principalement la programmation mobile, dont le contexte et les besoins de l'utilisateur changent fréquemment.

Les systèmes actuellement étudiés traitent des contextes pauvres pour des raisons de capacité limitée des objets mobiles support des applications, mais comptent beaucoup sur l'augmentation de celle-ci grâce à la miniaturisation. Les contextes employés sont souvent liés à la localisation, l'instant et les personnes alentour. Les buts de ces systèmes sont très variés : assistants professionnels, aide-mémoire, guide (touristiques ou autres)...

L'un des premiers systèmes opérationnels est l'*Active Badge* du laboratoire de recherche de Olivetti [Want *et al.*, 1992]. Chaque employé porte un badge actif émettant dans l'infrarouge. Des capteurs placés sur le pourtour du bâtiment permettent de localiser le badge de tout porteur actuellement dans les locaux. Ce système permet de savoir où se trouvent les personnes présentes sur le site (pour, par exemple, transférer un appel sur le poste le plus proche) et de déterminer les rapprochements de personnes. Le système *ParcTab* mis en place un peu plus tard à Xerox PARC [Want, 1995] est plus complexe. Les utilisateurs ne portent plus un simple badge, mais un assistant personnel électronique qui communique (toujours par infrarouge) non plus avec des capteurs associés au bâtiment dans son ensemble, mais avec un réseau local formé par les dispositifs de la salle où se trouve l'utilisateur. Outre les fonctions classiques et non-sensibles au contexte des PDA (calendrier, gestionnaire de tâches, carnet d'adresses...), les ParcTab offrent des fonctionnalités nouvelles et adaptées à la situation : information sur la salle, recherche de ressources disponibles les plus proches (par exemple les photocopieuses), lecture/modification de répertoires informatiques dédiés à la salle (pour laisser des notes aux collègues qui vont y passer par exemple), localisation de collègues, télécommande de différents appareils présents dans la pièce... Dans ces deux applications, le contexte permet de localiser le porteur (soit du badge, soit du ParcTab) et donc de pister les différents employés. Dans le cas du système ParcTab, le contexte adapte les applications disponibles et permet de trier les informations.

Les applications de guide sont beaucoup plus nombreuses, car elles sont sur un créneau porteur en association avec les téléphones portables et les systèmes de navigation routiers qui sont en pleine expansion commerciale. Ceux-ci localisent l'utilisateur (soit par GPS, soit par les cellules des réseaux de communication) et lui renvoient des informations (touristiques, environnementales, commerciales...) en fonction de ses préférences et de son environnement et adaptent leurs fonctionnalités. Le système GUIDE développé à l'Université de Lancaster [Davies *et al.*, 1998] est même capable de faire retrouver son chemin à un touriste égaré ou d'indiquer qu'il est déjà passé par le lieu courant. Le contexte permet d'adapter les

informations aux volontés de l'utilisateur, de sélectionner quelle information fournir, d'en déclencher d'autres...

Les aide-mémoire ont beaucoup à gagner en basant leur système sur le contexte. En effet, la mémoire humaine retient des éléments de connaissance en association avec le contexte dans lequel ces éléments ont été rencontrés. Par exemple où, quand, comment, avec qui tel événement a eu lieu. Un aide-mémoire est utile quand on ne se souvient plus d'un fait précis. Par contre, dans ces situations, il est fréquent de se souvenir de détails contextuels associés à ce fait. Les aide-mémoire, s'ils retiennent eux aussi le contexte en même temps que les faits, peuvent déterminer quel fait a eu lieu en fonction des éléments contextuels dont se souvient l'utilisateur. Tel est le cas notamment des systèmes *Forget-Me-Not* [Lamming et Flynn, 1994], *StartleCam* [Healey et Picard, 1998]...

Les *context-aware systems*, tels que ceux que nous venons de présenter, forment une grande famille d'applications concrètes de travaux de recherche sur les divers rôles que joue le contexte sur nos activités quotidiennes. L'augmentation des capacités des PDA en tout genre est de bonne augure quant à la prospérité de cette branche d'activité. Les premiers systèmes proposés ne prennent en compte qu'une infime partie du contexte, mais nous verrons bientôt arriver des systèmes plus complexes. Les contextes actuellement mis en œuvre sont pauvres, et leur structuration n'est pas nécessaire. A l'avenir, les concepteurs de tels systèmes, prenant en compte de plus en plus d'éléments contextuels, devront structurer les contextes mis en œuvre dans les systèmes et rejoindre ainsi les questions soulevées dans les trois exemples traités précédemment.

Section 5 :

Système d'Aide Intelligent en Contexte

Dans la section 3 de ce chapitre, nous avons souligné l'importance du contexte dans nos activités quotidiennes. Nous avons également pointé du doigt les difficultés rencontrées dès que l'on cherche à modéliser le contexte. D'abord sur le manque de consensus sur sa définition, ensuite sur l'aspect infini de cet ensemble. Dans cette section, nous exposons notre vision du contexte et comment nous allons le modéliser. Nous exposons ici ce que nous entendons par contexte, puis nous analysons son comportement au cours de processus complexes tels que la gestion d'incidents. Nous insistons notamment sur l'aspect dynamique du contexte et des états des éléments qui le compose. Enfin, nous présentons une extension du modèle de systèmes d'assistance intelligents intégrant de plus la notion de contexte à tous les niveaux où celui-ci intervient.

5.1 Modélisation du contexte

Nous avons vu précédemment que le contexte est défini de diverses manières. Etymologiquement, le contexte est ce qui entoure le texte, c'est à dire l'ensemble des éléments non-compris dans le texte lui-même mais qui influe sur son sens. Ainsi, la définition la plus communément acceptée est que le contexte est ce qui n'est pas directement lié au centre d'intérêt d'une entité (personne, groupe de personnes, institution, agent informatique...), mais qui a une influence potentielle sur le sens que peut donner l'entité à cet objet ou sur le traitement qu'elle veut en faire. Le contexte est donc défini par rapport à un centre d'intérêt particulier et pour une entité précise : Pour le même centre d'intérêt, deux entités différentes ne considèrent pas le même contexte ; Pour une même entité, deux centres d'intérêt, même très proches, ne font pas forcément référence au même contexte.

La notion d'intervention potentielle apporte une dimension particulière au contexte : tous les éléments du contexte ne sont pas utiles pour traiter un centre d'intérêt, et les éléments qui ne sont pas utiles ne sont pas forcément identifiés par l'entité. Certains éléments du contexte sont assez courants pour être ressentis bien qu'ils n'interviennent pas dans le traitement, d'autres passent inaperçus. Le contexte est donc un ensemble d'éléments pas forcément tous identifiés, dans lequel l'entité peut puiser afin d'affiner le traitement de son centre d'intérêt. C'est un ensemble dont l'entité ne connaît pas les limites. Chaque entité peut avoir conscience des éléments de contexte qu'elle a déjà rencontrés, mais des éléments nouveaux peuvent être nécessaires pour le traitement d'un centre d'intérêt particulier.

L'ensemble des objets d'attention est, pour chaque entité et à un instant donné, fini, mais nous rencontrons régulièrement des objets d'attention originaux. Ainsi l'ensemble des objets ayant captivé l'attention d'une entité est de taille finie, mais croissante. Il en est donc de même pour l'ensemble des éléments ayant déjà été identifiés dans les contextes de ces objets d'attention et pouvant former le contexte des prochains objets d'attention.

5.1.1 Connaissances contextuelles et connaissances externes

Le contexte du centre d'intérêt d'une entité est ainsi défini comme l'ensemble des éléments potentiellement utiles pour son traitement. Le mot clé ici est *potentiellement* ; il souligne le fait que tous les éléments ne sont pas effectivement utiles. L'ensemble des éléments du contexte qui interviennent effectivement lors du traitement du centre d'intérêt par l'entité joue, bien sûr, un rôle particulier. C'est dans cet ensemble que se trouvent toutes les informations qui sont utiles pour l'analyse de ce centre d'intérêt et que l'entité puise les connaissances sur lesquelles elle base le traitement. Nous désignons ce sous-ensemble du contexte sous le terme d'ensemble des *connaissances contextuelles*. Les connaissances contextuelles sont ainsi les éléments du contexte qui sont utiles ou peuvent être utiles à une

entité pour comprendre, analyser, expliquer, prendre des décisions, résoudre un problème donné, bref, qui sont effectivement utiles dans la situation présente [Brézillon, Pomerol et Saker, 1998]. Les éléments restant du contexte sont appelés *connaissances externes*.

L'ensemble des connaissances contextuelles se construit ainsi petit à petit au cours du traitement. L'ensemble définitif des connaissances contextuelles intervenant dans le traitement n'est connu qu'après coup. Toutefois, quand une entité porte son attention sur un objet, elle identifie rapidement les objets similaires déjà traités et se construit une première idée du nouveau centre d'intérêt. La construction de cette idée fait appel à des éléments de contexte en rapport avec les objets précédemment rencontrés ayant été mis en relation avec son centre d'intérêt. L'ensemble des connaissances contextuelles est ainsi initialisé avec des éléments déjà rencontrés utilisés principalement, au moins au début, pour la compréhension de l'objet en cours de traitement. Cet ensemble est ensuite enrichi pendant le traitement par l'ajout de connaissances contextuelles complémentaires. Ces connaissances sont soit des éléments ayant déjà été rencontrés par ailleurs (et pour lesquels l'entité a une expérience plus ou moins grande pour le recueil et l'utilisation), soit originales. Dans ce dernier cas, l'entité doit alors construire son expérience autour de ces connaissances contextuelles nouvelles. Une fois le traitement terminé, les connaissances contextuelles originales sont conservées dans l'ensemble des éléments déjà rencontrés et seront peut être de nouveau intégrées dans le contexte d'un centre d'intérêt futur. L'expérience qui a été faite par l'entité du recueil et de l'utilisation de ces connaissances déjà rencontrées facilite ainsi sa réutilisation future.

A chaque nouveau problème rencontré, une entité donnée rassemble l'ensemble des connaissances contextuelles qui lui sont utiles pour comprendre, analyser, expliquer et traiter ce problème. Le même problème devant être traité par une autre entité révélera un autre ensemble de connaissances contextuelles. Dans notre domaine d'application, les personnes chargées de la gestion d'incident travaillent plus ou moins en groupe (voir le second chapitre). Les connaissances concernant la gestion d'incident sont partagées par tous les intervenants. Toutefois, lorsqu'une nouvelle connaissance émerge d'une situation originale et inconnue du groupe, cette connaissance est rapidement propagée à tout le groupe grâce aux sessions de retour d'expérience et aux récits entre collègues lors des relèves. Nous pouvons ainsi considérer l'ensemble des agents chargés de la gestion des incidents d'une ligne comme formant une entité à part entière⁴.

⁴ Le fait de traiter les connaissances et raisonnements au niveau de l'ensemble des opérateurs plutôt qu'au niveau des opérateurs même met en évidence des éléments objectifs. Ceci évite de plus de traiter des informations personnelles pouvant être mal perçues par les opérateurs concernés

5.1.2 Contexte procéduralisé

L'ensemble des connaissances contextuelles activées par un centre d'intérêt donné peut être de taille considérable. Toutes ces connaissances contextuelles ne sont toutefois pas utiles à chaque instant du processus de traitement de ce centre d'intérêt. Aussi définissons nous le *contexte procéduralisé* comme le sous-ensemble des connaissances contextuelles utile à une étape donnée du processus de traitement par l'entité [Brézillon et Pomerol, 1998]. Cette opération permet de mettre en évidence les éléments du contexte effectivement responsables des décisions prises pour chaque étape du traitement du centre d'intérêt.

Nous ajoutons ainsi un nouveau niveau de dynamisme dans le contexte : le dynamisme de l'état des connaissances contextuelles. A chaque étape du traitement du centre d'intérêt, certaines connaissances contextuelles captent l'attention et passent ainsi dans le contexte procéduralisé et inversement, des éléments du contexte procéduralisé n'étant plus directement utiles pour l'étape courante redeviennent de simples connaissances contextuelles.

5.1.3 Liens entre les différents états des éléments du contexte

Le contexte est l'ensemble des éléments potentiellement utiles pour la compréhension et le traitement d'un centre d'intérêt donnée d'une entité donnée. Dès que l'entité se focalise sur cet objet, des connaissances contextuelles sont mises en œuvre explicitement. Au cours de l'analyse et du traitement, de nouvelles connaissances contextuelles complètent cet ensemble. Les éléments de contexte passent ainsi à l'état de connaissances contextuelles. A chaque étape de l'analyse et du traitement, toutes les connaissances contextuelles ne sont pas explicitement utiles. Celles qui le sont, sont alors dans un état procéduralisé. La Figure 1-1 montre une image arrêtée de l'imbrication des différents ensembles d'éléments de contexte définis ci-dessus (c'est une image arrêtée car le contexte procéduralisé évolue à chaque instant).

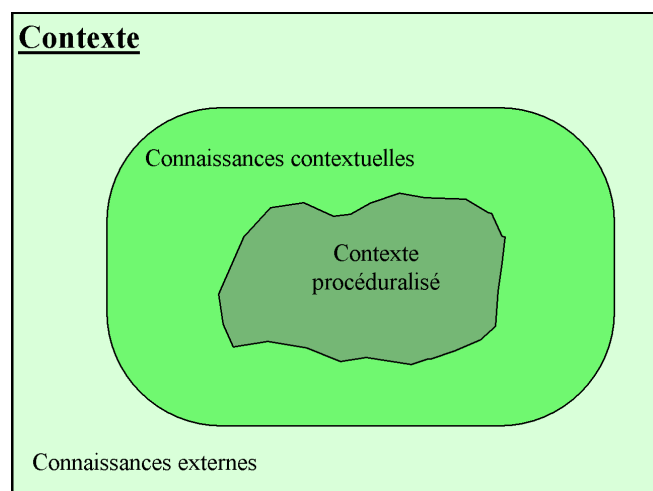


Figure 1-1: Contexte, connaissances contextuelles et contexte procéduralisé

Pour résumer, le contexte procéduralisé est composé des connaissances contextuelles qui sont explicitement exploitées à une étape donnée du traitement d'un centre d'intérêt ; les connaissances contextuelles étant les éléments du contexte concernant le centre d'intérêt dans son ensemble. Ceci peut être interprété, selon les termes de Carenini et Moore [1993], en contextes global et local. Le contexte global fait référence au centre d'intérêt et à ce qui s'y rapporte de manière générale ; il correspond ainsi à notre ensemble des connaissances contextuelles. Le contexte local est composé des éléments les plus récents qui contraignent effectivement la compréhension et peut être comparé à ce que nous nommons contexte procéduralisé.

Par ailleurs l'ensemble des éléments ayant déjà été identifiés lors de traitement d'objets d'attention forment un ensemble de taille croissante dans lequel l'entité peut puiser des éléments utiles pour se faire une première idée du centre d'intérêt. Cet ensemble est en perpétuelle évolution. Il n'est pas composé d'éléments du contexte proprement dit, mais d'éléments potentiellement inclus dans le contexte lié à un centre d'intérêt qui sont associés à une certaine expérience de l'entité. Il est important de garder en mémoire les expériences vécues ou apprises autour de ces éléments pour en faciliter l'utilisation lorsqu'ils font de nouveau partie d'un contexte.

5.2 Dynamiques du contexte

La définition du contexte que nous adoptons et la mise en avant des états contextuel et procéduralisé d'éléments du contexte nous place dans une vision hautement dynamique du contexte [Pomerol et Brézillon, 1999]. D'une part un contexte est activé pour chaque nouvel objet d'attention et constitué d'éléments potentiellement utiles à son traitement soit ayant déjà été identifiés lors de traitements d'objets d'attention précédents, soit originaux. Ceci met en avant le dynamisme des contextes en fonction des activités.

D'autre part, certains éléments du contexte entrent progressivement dans l'ensemble des connaissances contextuelles qui regroupe les connaissances effectivement utiles pour l'analyse, la compréhension, le traitement d'un objet d'attention. Ceci met en avant la dynamique liée à l'ensemble des connaissances contextuelles. Ce sous-ensemble du contexte étend ses frontières dans le contexte. Tout élément du contexte qui, au cours de l'analyse et du traitement de l'objet d'attention, devient connaissance contextuelle, garde ce statut jusqu'à la fin.

Enfin le contexte procéduralisé est composé des connaissances contextuelles contraignant explicitement l'étape courante du traitement de cet objet. C'est un sous-ensemble des connaissances contextuelles dont la frontière est totalement libre : lors de l'étape courante du

traitement, des connaissances contextuelles sont procéduralisées si besoin est et des connaissances contextuelles procéduralisées sont déprocéduralisées le cas échéant... Le dynamisme de procéduralisation/déprocéduralisation des connaissances contextuelles suit en réalité le dynamisme du point de concentration de l'attention. Nous notons également ici que tout élément du contexte procéduralisé est partie intégrante des connaissances contextuelles. Si, lors d'une étape d'analyse ou de traitement, un élément de contexte doit intervenir directement dans le processus, il entre par la même occasion dans l'ensemble des connaissances contextuelles s'il n'en faisait pas encore partie.

5.3 Définition d'un SAIC

Comme nous le voyons, le contexte joue un rôle important dans les raisonnements évolués. Dans la section 3.2, nous avons rappelé les différents rôles que joue le contexte dans les quatre principaux champs concernant les systèmes d'aide : les connaissances, les raisonnements, l'acquisition de connaissances et les explications. Il nous semble primordial, dans le cadre du développement de systèmes visant à aider un (ou des) opérateur(s) humain(s), d'en tenir compte.

Pour des systèmes d'aide dans des domaines fortement contraints par le contexte, nous proposons le développement de Système d'Aide Intelligent en Contexte (SAIC) [Brézillon 1999a]. Ceux-ci sont des SAI tels que ceux préconisés par Boy [1991] qui ont de plus la particularité de baser leur fonctionnement sur le traitement du contexte et de sa dynamique, et ce à tous les niveaux impliqués.

Comme pour le système ORCA de Turner [1995], le contexte n'est pas simplement ajouté au système, mais constitue la pierre angulaire de l'édifice. Le système ne se contente pas d'utiliser le contexte ; son fonctionnement est basé sur celui-ci. Le modèle du contexte proposé dans la section précédente est alors intéressant à deux niveaux importants. Premièrement, il permet de bien délimiter ce qu'est le contexte et comment il intervient au niveau des connaissances, des raisonnements, de l'acquisition des connaissances et des explications. Deuxièmement, il propose un système de gestion du contexte et de ses sous-ensembles opérationnels.

En ce qui concerne les raisonnements, les éléments définis précédemment permettent de mettre en évidence les rôles joués par chacun d'eux pour chaque étape. L'explicitation du contexte procéduralisé permet de bien voir quels sont les éléments qui assoient les décisions prises. Le contexte procéduralisé permet également d'expliquer ces choix. Les connaissances contextuelles sont, quant à elles, principalement associées aux connaissances du domaine et à l'acquisition de connaissances. Elles permettent d'établir une liste d'éléments du contexte

ayant joué un rôle particulier lors de traitements précédents et constituent ainsi un ensemble d'éléments pouvant être surveillés, à des fins préventives par exemple. L'acquisition de connaissances est concernée par les connaissances contextuelles, car à chaque fois qu'une nouvelle connaissance contextuelle est détectée, les objets du domaine pouvant être concernés par celle-ci doivent en être informés.

En cours d'opération, le système d'aide doit être capable de décharger l'opérateur de certaines tâches et de surveiller ses activités. Pour cela il est très important que le système d'aide soit conscient du centre d'intérêt de l'utilisateur, de son contexte, des connaissances contextuelles de l'étape courante de traitement et du contexte procéduralisé à cet instant. Ces éléments permettent de suivre l'activité de l'utilisateur et de détecter une déviance par rapport à ce qui est attendu. Cette déviance, qui peut être normale dans la mesure où l'utilisateur peut développer une stratégie nouvelle et donc inconnue du système, doit être détectée pour : premièrement en avertir l'utilisateur (au cas où il ne serait pas au courant de sa déviance parce qu'un élément de connaissance lui aurait malencontreusement échappé), deuxièmement, faciliter l'acquisition par le système de la nouvelle stratégie quand il s'avère que l'utilisateur maîtrise la situation.

Section 6 :

Conclusion

Les Systèmes Experts de première génération, généralement basés sur des règles de production, permettent de modéliser des raisonnements fortement structurés. Ce type de raisonnements est une grande classe parmi les raisonnements humains. Toutefois, en réalité, ils sont souvent modulés par des logiques non-monotones. Les résultats obtenus étant très encourageants, la voie continue d'être explorée. La seconde génération de Systèmes Experts est plus souple tant au niveau des logiques adoptées (utilisations de logiques floues, temporelles...) qu'au niveau de la structuration des connaissances introduites dans le système. Toutefois, les systèmes obtenus sont souvent mal perçus par les utilisateurs : les systèmes répondent à une très large majorité des problèmes rencontrés, mais ce sont les problèmes pour lesquels les utilisateurs n'ont pas de doutes. De plus, bien que prévus pour une évolution de leurs capacités, les Systèmes Experts se heurtent à l'impossibilité de l'énumération, *a priori*, de l'ensemble des circonstances dans lesquelles une décision donnée ou une action peuvent être réalisées. Il est donc nécessaire de repenser entièrement l'architecture des Systèmes Experts pour les rendre plus ouverts et proposer des mécanismes d'acquisition incrémentales de connaissances. Enfin, la plupart des systèmes entretiennent des relations de type « expert à novice » avec l'utilisateur et les explications données sont souvent pauvres et peu ciblées.

Tout ceci relègue finalement la plupart des Systèmes Experts dans des rôles subalternes de système d'entraînement plutôt que de système d'aide.

La tendance actuelle est au développement de systèmes d'aide. Ceux-ci ne cherchent pas à avoir la même expertise que l'utilisateur mais plutôt à la compléter par des compétences particulières auxquelles les ordinateurs sont bien adaptés. Les Systèmes Interactifs d'Aide à la Décision proposent ainsi tout un panel de fonctionnalités de pré-traitement d'informations afin de construire une vue simple et efficace des relations de cause à effet des décisions sur les faits et ce afin de faciliter la prise de décision de l'utilisateur. L'utilisateur reste maître de la situation et se sert du système pour construire un modèle concernant ses décisions en se déchargeant de calculs et autres opérations répétitives de bas niveau. Dans cette optique, il n'est pas rare de concevoir une architecture logicielle basée sur l'utilisation de petits Systèmes Experts parfaitement ciblés pour le traitement des diverses tâches requises.

Dans d'autres domaines que l'aide à la décision, nous voyons l'émergence de système assistant un opérateur dans ses activités quotidiennes. Les Systèmes d'Assistance Intelligents apportent notamment une vision intéressante de la relation utilisateur/système. Le but est d'assister l'utilisateur en le déchargeant d'opérations secondaires et de surveiller soit le processus sur lequel travaille l'opérateur, soit les activités de l'opérateur lui-même pour signaler toute manœuvre ou événement inattendu et donc potentiellement à risque. Ces systèmes sont fondés sur une forte interactivité permettant de faire évoluer le Système d'Assistance Intelligent et donc d'améliorer à la fois l'efficacité du système mais aussi celle du couple système-opérateur.

Par ailleurs, le contexte joue un rôle primordial sur de nombreux points importants pour un système intelligent. Il contraint notamment les raisonnements et les explications liés à une tâche particulière. De plus la compréhension même de la tâche est fortement liée au contexte. Les nombreux travaux autour du contexte ont eu tendance à compliquer le sens donné à cette notion. Toutefois un consensus semble s'établir lentement autour de l'idée que le contexte est ce qui n'est pas directement lié au centre d'intérêt d'une entité mais qui a une influence potentielle sur le sens que peut donner l'entité à cet objet ou sur le traitement qu'elle veut en faire. Le contexte intervient ainsi à tous les niveaux dans le développement de système d'aide interactifs. Les connaissances peuvent changer de sens en fonction du contexte, les raisonnements donner des résultats très différents ; les explications sont particulièrement sensibles aux circonstances dans lesquelles elles doivent être données et du niveau de l'utilisateur ; l'acquisition des connaissances est, quant à elle, indissociable du contexte dans lequel ces connaissances sont acquises.

Le contexte joue un rôle particulier dans la gestion quotidienne de systèmes dynamiques complexes. La gestion de tels systèmes est une tâche difficile. Les contraintes techniques et

temporelles sont fortes et les opérateurs chargés de cette mission doivent à tout instant concilier des objectifs parfois antagonistes. Pour faciliter leur tâche, les responsables des systèmes concernés, parfois aidés des concepteurs, éditent des procédures. Celles-ci consistent en des structures organisant le diagnostic et les actions à entreprendre pour gérer des états particuliers du système. Les procédures sont toutefois trop générales et les opérateurs préfèrent reconstruire des stratégies de gestion des situations à partir de celles-ci et de leurs expériences personnelles afin de s'adapter au mieux à la réalité. Cette opération correspond à une recontextualisation de la procédure. Le seul résultat tangible de cette opération est la pratique réellement appliquée par les opérateurs. Celle-ci consiste en une séquence d'actions contextualisées associées aux divers éléments du contexte ayant guidé les choix stratégiques et tactiques de l'opérateur au cours de la gestion de la situation. Ces pratiques contiennent une forme compilée du contexte et peuvent être réutilisées, après analyse et restructuration, pour augmenter l'acuité des procédures face au contexte.

De nombreux systèmes intègrent désormais la notion de contexte, plus ou moins profondément. Le formalisme de McCarthy [1979] autorise la manipulation de contextes abstraits par les systèmes basés sur la logique du premier ordre. Il introduit ainsi une certaine non-monotonie fort intéressante dans ces systèmes de logique. Son formalisme définit de plus les contextes les uns par rapport aux autres et amène des réflexions fructueuses sur le rôle du contexte, sa dimension infinie et les méthodes pour la généralisation de modèles définis dans un contexte particulier. Le système ORCA de Turner [1995] est une application performante basant le contrôle de véhicules submersibles autonomes sur le contexte des dits véhicules. Elle utilise des schémas contextuels pour définir des lignes de conduites du système et des réflexes adaptés au contexte du véhicule. Des schémas procéduraux prennent le relais pour la réalisation de tâches ne dépendant pas du contexte. L'approche de Gonzalez et Ahlers [1994 ; 1995] ajoute de plus une hiérarchisation des contextes séparant ceux qui relèvent de la mission dans son ensemble des contextes intervenant à un niveau stratégique ou à un niveau tactique. Enfin, les *Context-Aware Systems* envahissent de plus en plus notre quotidien. Ces systèmes se basent sur des données contextuelles pour adapter leur comportement et leurs fonctionnalités à la situation réelle.

Le contexte joue un rôle particulier dans notre vie quotidienne. Il semble donc logique de l'inclure dans les systèmes cherchant à nous simplifier la vie. Nous avons analysé le contexte et sa dimension infinie nous oblige à faire la part des choses. La première propriété du contexte est qu'il n'est défini que par rapport à un centre d'intérêt particulier. Une fois cet objet d'attention défini, le contexte peut être analysé. Nous différencions ensuite les éléments du contexte qui sont utiles de ceux qui ne le sont pas. Nous définissons ainsi deux niveaux d'utilité : premièrement les éléments utiles pour l'ensemble de la tâche que nous considérons, que nous appelons connaissances contextuelles, et deuxièmement les éléments qui sont utiles

à une étape donnée de cette tâche, que nous appelons contexte procéduralisé. Cette sélection de deux sous-ensembles imbriqués du contexte induit deux dynamismes particuliers. Premièrement l'ensemble des connaissances contextuelles est un sous-ensemble du contexte qui se construit petit à petit au cours du traitement de la tâche. Tout élément du contexte qui a été utile pour comprendre, analyser, décider est inclus dans cet ensemble et n'en sort plus. Deuxièmement, le contexte procéduralisé est un sous-ensemble de celui des connaissances contextuelles formé par celles qui sont utiles à une étape précise du traitement de la tâche. Les éléments de cet ensemble peuvent en sortir dès lors qu'ils n'interviennent plus explicitement dans l'étape en question.

Cette dynamique de l'état des éléments du contexte associée à l'accroissement régulier des connaissances ayant été un jour contextualisées permet d'explicitier le rôle du contexte dans un système d'aide interactif. Nous proposons donc le modèle de Systèmes d'Aide Intelligent en Contexte pour étendre les compétences des Systèmes d'Assistance Intelligents proposés par Boy en basant son fonctionnement sur le contexte. Le contexte intervient ainsi aux quatre niveaux précédemment mis en avant. Cette explicitation du contexte et la structuration que nous proposons permet d'améliorer considérablement l'efficacité du système d'aide en focalisant le SAIC sur le contexte de la tâche de l'opérateur et sur son évolution, afin d'obtenir un assistant compétent et capable d'adaptations à la situation et à l'évolution des connaissances du domaine mais également stratégiques.

Deuxième chapitre :

Gestion d'incidents sur une ligne de métro

Le métro parisien est un système ferré de transport de voyageurs. La majorité des trajets sont souterrains, ce qui rend la gestion des incidents particulièrement difficile. Riche d'un siècle d'histoire et d'expérience, la Régie Autonome des Transports Parisiens, héritière de l'exploitation du métro, cherche sans cesse à améliorer la gestion des incidents qui y surviennent.

Les agents de la régie chargés de la régulation sont des experts : ils connaissent parfaitement le terrain et les règles à suivre pour gérer les incidents. Leur tâche consiste à la fois à réguler le trafic et à gérer les incidents. L'activité qui en résulte est complexe et fortement liée au contexte. L'analyse de cette activité met en évidence l'utilisation de concepts pragmatiques et de schèmes d'action de la part des opérateurs.

Dans ce chapitre, nous présentons le métro parisien : son histoire, sa structure, son organisation. La suite de l'exposé traite plus particulièrement de la gestion des incidents, de la tâche et de l'activité des agents qui en sont chargés. Nous étudions tout particulièrement le rôle que joue le contexte dans ces activités. Nous présentons enfin les diverses voies déjà explorées par la régie pour aider les agents de la régulation dans leur travail quotidien.

Section 1 :

Le Métro Parisien

Le chemin de fer métropolitain, plus communément appelé *métro*, est un système de transport ferroviaire urbain de passagers. Ce système voit le jour à la fin du XIX^{ème} siècle à Londres, puis dans d'autres grandes villes européennes, dont Paris. Nous présentons dans cette section une brève histoire du métro parisien et du contexte dans lequel ce système est né, le transport de passagers au tournant du XX^{ème} siècle et le rôle que joue le métro naissant à cette époque. Nous ne présentons ici qu'une esquisse afin de situer notre étude, ces points sont détaillés par Tricoire [1999] et d'un point de vue plus technique par Robert [1983]. Nous insistons ensuite sur la structure du réseau, sur ce qu'est une ligne de métro et sur les spécificités du réseau parisien. Pour finir, nous présentons l'exploitation du métro parisien, à l'heure actuelle, son organisation en temps normal et la gestion des incidents.

La complexité du système « métro » est telle que nous ne pourrions pas ici présenter tous ses détails et ceux des activités de régulation qui en découlent (l'étude du système serait une thèse à elle seule). Nous nous contentons d'un niveau de détails permettant de donner un cadre général et d'illustrer les propos soutenus dans la suite de la thèse.

1.1 Un peu d'histoire

Au cours du XIX^{ème} siècle, les grandes villes s'étouffent devant la forte croissance tant démographique qu'économique et industrielle. Haussmann, Préfet de la Seine, entreprends la construction de grands boulevards sillonnant Paris afin de fluidifier le trafic. Les déplacements sont de plus en plus nombreux : marchandises et personnes empruntent des transports pédestres, hippomobiles ou ferrés. Depuis 1828, date de la création de l'*Entreprise Générale des Omnibus* par Baudry, de nombreuses compagnies proposent des services de transport soit par des omnibus hippomobiles, soit par des tramways à traction diverse (funiculaire, air comprimé, accumulateurs inertiels...). En 1855, Haussmann impose la fusion des différentes compagnies d'omnibus afin de clarifier l'offre, mais la croissance de la ville impose une restructuration profonde de l'offre de transport.

Le transport de voyageurs est un élément crucial pour le développement des villes. Les rues sont saturées, et les offres de transports de surface ne suffisent pas à fluidifier le trafic. Les omnibus et surtout les tramways offrent une grande capacité de transport intra-muros, mais sont tributaires de la circulation de surface. La nécessité d'un transport indépendant de la circulation, soit souterrain soit sur des viaducs, se fait sentir et de nombreux projets sont proposés.

Les premiers systèmes de transports ferrés de voyageurs permettent principalement d'atteindre les campagnes environnantes de Paris (la ligne de Sceaux, par exemple). C'est justement sur ce point que la ville de Paris défend son projet de chemin de fer métropolitain face aux liaisons de gares à gares soutenues par l'Etat. En effet, la réalisation d'un système de transport réalisant un réseau d'interconnexions entre les différentes gares permettrait aux travailleurs de vivre à l'extérieur de Paris et risquerait de vider la ville de ses habitants, transformant ainsi Paris en une ville morte la nuit, à l'image de ce qui se passe à Londres. L'enjeu des débats est donc de taille ! La ville de Paris et l'Etat rentrent alors dans un conflit d'intérêt quant à la création et l'exploitation d'un réseau ferré parisien. L'Etat préconise la création de liaisons entre les différentes gares parisiennes existantes afin, notamment, de simplifier les déplacements de personnes et de marchandises au travers de Paris. La municipalité préfère un transport orienté voyageurs dont le rôle est de mieux répartir les habitants dans la ville et de désenclaver les quartiers les moins prisés.

Outre les tergiversations politiques entre l'Etat et Paris, la création d'un système de transport ferré métropolitain soulève de nombreuses questions techniques. Pour augmenter la capacité de circulation il faut doubler les grands boulevards de la ville, soit en sous-sol, soit au-dessus des voiries. Aux débats techniques s'ajoutent alors les débats esthétiques et concernant la salubrité. De plus, la traction à vapeur pose de nombreux problèmes et limite l'utilisation des tunnels (donc qui plus est d'un réseau principalement souterrain). Les discussions s'éternisent, mais l'exposition universelle de 1900 met un terme aux tergiversations. En 1895, l'Etat cède et donne carte blanche à la ville pour la réalisation d'un réseau métropolitain. De nombreux projets ont été proposés. Le projet retenu est un transport ferré principalement souterrain, de petit gabarit (pour limiter les coûts de réalisation), à traction électrique.

Les premières lignes sont réalisées sous la direction de Fulgence Bienvenüe : La Ligne 1, *Porte de Vincennes – Porte Maillot*, est inaugurée le 19 juillet 1900, juste à temps pour l'exposition universelle.

1.2 Structure du réseau

Le réseau du métro de Paris est initialement conçu comme un réseau de tramway, dont la particularité principale est qu'il est souterrain. Ainsi, il n'y a pas de ligne tel que nous le concevons aujourd'hui. Les voies sont interconnectés autorisant des trajets divers aux trains. La "ligne" 1 est ainsi composée d'un grand axe *Porte de Vincennes – Porte Maillot*, mais également de deux embranchements : *Etoile – Porte Dauphine* d'une part et *Etoile – Trocadéro* d'autre part. Ces deux embranchements ont été prolongés pour aboutir à l'ouverture de la *ligne 2 nord* (avril 1903) et *2 sud* (5 novembre 1905 jusqu'à *Passy*, puis 24 avril 1906 jusqu'à *Place d'Italie*). Le réseau prend de l'ampleur et les lignes sont de plus en plus différenciées.

Désormais, le réseau est constitué de 14 lignes dont l'exploitation est réalisée indépendamment les unes des autres. Nous détaillons maintenant la structure du réseau actuel.

1.2.1 Qu'est-ce qu'une ligne de métro ?

Une ligne de métro peut être définie selon différents points de vue. On peut d'abord distinguer les voies *principales* des voies *secondaires*. Les premières sont les voies constituant la "ligne" en exploitation normale, où circulent toutes les rames avec voyageurs. Les voies secondaires sont les autres voies : voies de raccordement entre deux lignes, voies d'évitement, voies en direction des ateliers. Désormais chaque ligne a ses voies propres (ses voies principales) et est reliée aux autres lignes grâce aux voies de raccordement, qui sont les vestiges du réseau fortement interconnecté du début du métro parisien.

Une ligne peut ensuite se définir comme la succession de **stations**, plus ou moins distantes. Ce point de vue, celui des voyageurs, permet de définir les stations mais aussi les **interstations** qui sont les portions de lignes comprises entre deux stations consécutives. La plupart des interstations sont sous tunnel, bien qu'un certain nombre soient aériennes et quelques-unes mixtes. Dans le même ordre d'idées, on distingue des stations particulières, les **terminus**, qui jouent un rôle administratif de gestion des conducteurs et des matériels roulants. Chaque ligne porte au moins deux terminus, dont l'un est qualifié de **principal**, les autres étant qualifiés de **secondaires**. Chaque terminus est équipé de voies secondaires permettant de gérer les trains (garage, dégarage, circulation en direction des ateliers...). Les terminus sont également équipés d'un **Poste de Manœuvre Locale** (PML) qui permet d'aiguiller les trains dans la zone du terminus, de communiquer avec les conducteurs des trains situés dans la zone du terminus et avec les autres centres opérationnels de la ligne, d'organiser les départs des rames et de gérer les conducteurs en réserve dans le terminus ou en changement de service. Suite aux diverses extensions des lignes, certains terminus ne sont plus en fin de ligne ; ils sont alors qualifiés d'**intermédiaires**.

Les voies principales de chaque ligne sont numérotées : les voies impaires partent du terminus principal vers le (ou les) terminus secondaire(s), les voies paires sont dédiées au trajet inverse. En général les lignes sont constituées uniquement des voies 1 et 2, mais dans certaines zones, d'autres voies principales peuvent exister (notamment en terminus).

Une ligne peut être également décrite d'un point de vue technique, en fonction de l'**organisation de l'alimentation électrique**. La puissance électrique nécessaire à la traction des trains présents sur une ligne est importante et ne peut être fournie que par plusieurs **Postes de Redressement** (PR). Les PR convertissent l'électricité haute tension alternative en un courant continu de moyenne tension exploitable par les trains circulants sur la ligne. La présence de plusieurs points d'alimentation autorise la segmentation de la ligne en **sections d'alimentation**. Les sections sont des parties de ligne pouvant être, le cas échéant, exploitées indépendamment du reste de la ligne. Pour cela elles sont pourvues d'au moins un poste de redressement et d'un ou plusieurs **appareils de voie**⁵. En temps normal, chaque section est alimentée indépendamment des sections l'encadrant, mais peut être alimentée par une section voisine par la fermeture d'un **Contacteur de Sectionnement** (CS). Les sections sont segmentées en **sous-sections**. Deux types de sous-sections existent : certaines sous-sections sont liées à un poste de redressement et sont donc directement alimentées, les autres sont alimentées par la fermeture des **Sectionneurs d'Isolement Télécommandés** (SIT) en temps

⁵ L'appareil de voie est un dispositif constitué de deux aiguillages tête-bêche autorisant le passage d'un train d'une voie principale à l'autre

normal ou mises hors-tension par l'ouverture de ces SIT si nécessaire. Les sous-sections sont elles-mêmes découpées en *sections élémentaires* : ces portions de ligne peuvent être isolées de leurs voisines en ouvrant des sectionneurs situés le long des voies. Elles sont utilisées exceptionnellement lorsque le courant doit être coupé en une zone précise et peu étendue pendant une longue période.

D'un point de vue systémique, une ligne peut être considérée comme un système ouvert (flux de voyageurs entre l'enceinte du métro et l'extérieur), hautement technique (alliant la complexité d'un système ferroviaire à celle des tunnels), complexe et où le maître-mot est la *sécurité*, à cause de la proximité entre les voyageurs et les installations électriques de forte intensité.

1.2.2 Particularité du métro parisien

Le métro de Paris se distingue principalement par sa densité. La volonté de la municipalité de Paris au XIX^{ème} siècle est de desservir un grand nombre de quartiers afin de faciliter les déplacements d'une majorité de parisiens. Cette politique a conduit à la réalisation d'un réseau constitué de nombreuses lignes très rapprochées et fortement interconnectées. De plus les interstations sont courtes (800m maximum et 600m de moyenne dans Paris intra-muros). De ce fait, un plus grand nombre de rames peuvent être mises en exploitation simultanément, aboutissant à un intervalle minimal de 90 secondes sur la ligne 4 en heure de pointe.

Sur une ligne de métro toutes les rames en circulation sont omnibus et roulent à la même vitesse. Ainsi un voyageur désirant se rendre d'une station à une autre monte dans la première rame arrivant à quai et ne se soucie que du temps de parcours entre les deux stations concernées. Ce temps de parcours est constant (sauf incidents ou perturbations) et dépend principalement de la longueur et du profil des interstations.

Toutes les lignes de métro ne sont pas équipées du même genre de matériel roulant. Deux grandes classes de trains existent : les trains à roulement fer et les trains à pneus. Les trains à pneus sont toutefois équipés de roues fer pour pouvoir circuler sur les zones non spécifiques et en cas de crevaison. Une ligne exploite, en général, un seul type de trains.

1.2.3 Eléments constitutifs d'une ligne

Les voies sont équipées d'un système permettant de suivre le trajet d'un train. Ce système est réalisé par le découpage des voies en portions de longueurs plus ou moins importantes et l'équipement de ces portions de voies d'un système électrique informant de la présence d'un train. Ces portions de voies sont appelés *Circuits De Voie* (CDV). Les CDV n'informent que

de la présence d'un train en un lieu précis ; ces informations sont relayées par un calculateur afin de suivre le trajet des trains.

Les voies sont constituées de deux *rails de roulement*, doublés par des *pistes de roulement* et des *guides verticaux* pour les lignes accueillant des trains à roulement pneumatique. L'un des rails de roulement est relié à la masse du courant de traction, le +720V étant quant à lui distribué soit par un rail supplémentaire (appelé *rail de traction*), soit par l'un des guides pour les zones accueillant des trains à roulement pneumatique.

La sécurité est primordiale. La ligne est équipée d'un système de coupure de courant par section. Des *Rupteurs d'Alarme* (RA) sont régulièrement espacés le long des voies (environ tous les 50m) et équipent les quais (à chaque bout et au milieu du quai). Toute personne peut couper en urgence le courant de traction sur la section correspondant au RA actionné, ainsi que sur la section voisine dans les zones frontières entre deux sections.

1.3 Exploitation du réseau

L'*exploitation* d'une ligne consiste en la proposition d'une offre de transport par le passage régulier de trains aptes au transport de voyageurs, appelés *rames*. Cette offre est adaptée à la demande par l'ajustement de la fréquence des rames. Les chefs de départ (au PML, dans les terminus) sont chargés de respecter au mieux l'offre de transport prévue. Cette offre prévue dépend du jour (jour ouvrable ou férié, vacances, saison...) et de l'heure (heure creuse ou heure de pointe, matin, après-midi ou soir...) et est compilée dans un tableau appelé *garde-temps*. Le garde-temps donne l'heure théorique de départ des rames dans les terminus, calculée en fonction des fréquentations observées les années précédentes aux mêmes périodes, dans les même conditions, adaptées aux particularités prévues pour l'année en cours. Ce tableau représente une forme très compacte d'une quantité importante de données concernant l'exploitation.

1.3.1 Organisation de l'exploitation

L'exploitation de la ligne en temps normal est réalisée en deux types de lieux. Premièrement aux terminus, dans les Postes de Manœuvre Locaux, des opérateurs régulent la circulation des trains situés dans la zone de contrôle du terminus et gèrent les conducteurs et les rames afin de respecter au mieux le garde-temps. Deuxièmement, au Poste de Commande Centralisé, les opérateurs ont en charge la régulation des trains situés en dehors des zones des terminus et la gestion des incidents d'exploitation.

Dans chaque PML un opérateur, appelé *chef de départ*, gère les trains et les conducteurs disponibles afin de respecter au mieux l'offre théorique de transport du garde-temps. Ceci

comprend plusieurs tâches. Premièrement, ils doivent tenir compte des conducteurs présents au terminus, qu'ils soient rattachés à ce terminus ou à un autre, du rattachement des conducteurs présents et de leurs horaires de travail. Ce problème n'est pas simple, il s'agit de calculer au plus juste les temps de parcours de la ligne à chaque instant et d'estimer l'heure d'arrivée des rames à l'autre bout de la ligne, de comparer cet horaire avec ceux des conducteurs et de choisir les conducteurs afin d'optimiser le nombre de trajet effectué par chacun d'eux. Deuxièmement, les chefs de départ choisissent les trains à envoyer en ligne. Tous les trains ne sont pas identiques : outre les trains pour le service voyageur, il existe des trains des services de maintenance. Parmi le premier type de trains, certains sont pourvus d'un système de graissage des rails afin de limiter l'usure et les nuisances sonores (notamment dans les courbes). La plupart des trains envoyés en ligne sont dédiés au service voyageur et régulièrement un train graisseur est intercalé (la fréquence dépend des lignes, sur la ligne 4 elle est d'un train sur 5). De plus, des trains des services de maintenance peuvent être envoyés et garés sur des voies secondaires afin de préparer les travaux de la nuit suivante. Tout ces choix (conducteurs et trains) tentent de suivre au mieux l'offre théorique de transport et sont adaptés à chaque instant aux conditions réelles de l'exploitation (incidents en cours, retards cumulés au départ et à l'arrivée, fluidité du trafic sur les deux voies...).



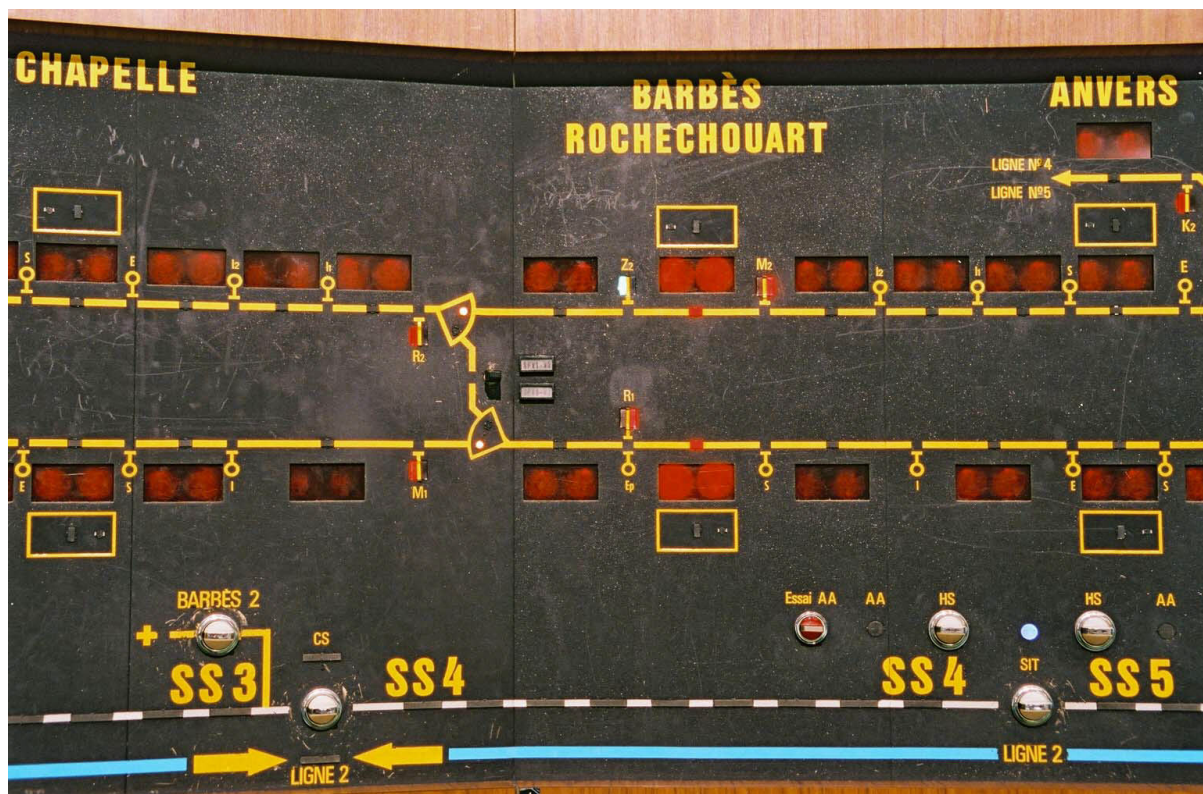
© RATP - DUMAX Gérard

232992 - 24/03/1998

Figure 2–1: Photo des TCO et consoles des lignes 7-7bis (console de gauche, TCO du haut) et de la ligne 1 (console de droite, TCO du bas)

La régulation des trains en ligne est réalisée par les *Chefs de Régulation* (CREG), qui travaillent au *Poste de Commande Centralisé* (PCC) (Figure 2–1). Chaque ligne est associée à un PCC. Tous les PCC étaient rassemblés, jusqu'en novembre 2001, en un même lieu. Sur



la Figure 2–1, nous pouvons voir deux postes côte à côte, correspondant à deux lignes distinctes. Pour chaque poste, le CREG fait face à un *Tableau de Contrôle Optique* (TCO) (arrière-plan) et à une *console* (plan intermédiaire). Les CREG sont les relais entre les différents acteurs de l'exploitation de la ligne. Ils sont en contact avec les conducteurs par la *Téléphonie Haute Fréquence* (THF) (sur l'aile extérieure de la console), avec les agents de station et l'*Inspecteur Principal d'Exploitation* (IPEX, intermédiaire pour les services extérieurs tels que les pompiers, la police, le déminage) par des téléphones directs, avec les personnes sur les quais par des téléphones automatiques. Sur le bloc entre les deux consoles on peut voir les téléphones direct et automatique. De plus, les CREG peuvent donner l'ordre à une rame de stationner jusqu'à nouvel ordre grâce au *Départ Sur Ordre* (DSO), sorte de signal semi-automatique disposé en tête de chaque quai. Les commandes de DSO sont situées au milieu de la console.



© RATP - DUMAX Gérard

201563 - 17/02/1992

Figure 2–2: Photo d'un détail du TCO de la ligne 2

Le TCO (dont la Figure 2–2 présente un détail) représente la ligne, l'état d'alimentation des sections et sous-sections, l'état des signaux de manœuvre et des aiguillages, la position et les numéros des trains qui y circulent. La partie supérieure représente les voies, les stations et les trains. Les voies sont représentées par les lignes jaunes. Elles sont représentées par CDV, chaque élément ainsi identifié est limité par les signaux l'encadrant (soit des signaux d'espacement, représentés par les , soit des signaux de manœuvre représentés par des )

et est muni d'un indicateur lumineux rouge précisant la présence d'un train. Les appareils de voie (paires d'aiguillages tête-bêche permettant le passage d'une voie à l'autre) sont représentés par un CDV passant d'une voie à une autre. La position de chacun des aiguillages est représentée par un indicateur lumineux blanc. Les appareils de voie équipés pour un service provisoire télécommandé sont associés à une commande du SP à trois positions (« SP de voie 1 à voie 2 », « service normal » et « SP de voie 2 à voie 1 ») placée sur le TCO à droite de l'appareil de voie concerné. Les quais des stations sont représentés par des rectangles jaunes (en général les quais encadrent les deux voies et sont donc représentés à l'extérieur des lignes de CDV). Les trains sont identifiés par des numéros qui sont affichés entre les signaux encadrant un CDV. Le TCO porte également, dans sa partie inférieure, une représentation du sectionnement électrique, de l'état d'alimentation de chaque sous-section et les commandes de gestion d'énergie de traction (boutons CS entre SS3 et SS4, en bas à gauche et SIT entre SS4 et SS5, en bas à droite), des indicateurs de signal d'alarme (AA, qui s'allument dans une section mise hors tension par l'actionnement d'un rupteur d'alarme) et des commandes liées au circuit de rupteurs d'alarme (Essais AA et HS). Les CREG utilisent les moyens à disposition (DSO, télécommande au TCO, moyens de communication divers...) pour réguler le trafic, gérer les incidents et informer les différents acteurs de l'exploitation.

Jusqu'en novembre 2001, date de la mise en service du nouveau PCC de la ligne 4, les centres de régulations étaient organisés de la manière suivante. Chaque ligne a un PML dans chacun de ses terminus, afin de gérer les trains et les conducteurs qui s'y trouvent. Ce sont des postes de régulation situés sur le terrain. Les PCC de toutes les lignes étaient rassemblés en un lieu unique, permettant ainsi aux CREG d'une ligne de venir aider lors d'un incident conséquent. Cette organisation, associée à un double système de roulement des équipes au PCC, facilite la diffusion d'informations, d'un CREG à un autre, à propos d'incidents particuliers. Ainsi les CREG forment une communauté de pratique telle qu'elle est décrite par Brown et Duguid [1991].

Le nouveau PCC de la ligne 4 propose une organisation très différente, séparant son PCC de ceux des autres lignes, et rassemblant le PCC et les deux PML dans une même salle. La mise en place de ce nouveau type de salle de contrôle effectue un double mouvement de *décentralisation* des PCC et de *relocalisation* des centres de régulation au plus près de la ligne.

Le système construit sur les principes exposés dans cette thèse étant testé en marge du nouveau PCC de la ligne 4 et celui-ci ayant été mis en service en novembre dernier, nos propos concernent par défaut les PCC centralisés, et nous précisons quand ceux-ci ne concernent que le nouveau poste.

1.3.2 Incidents d'exploitation et gestion des incidents

Dans un système ouvert de cette complexité les incidents d'exploitation sont inévitables. Ils sont de natures et de gravités diverses. Quand ceux-ci surviennent sur les voies principales, il incombe au CREG de gérer ces incidents d'exploitation. Il utilise alors les outils mis à sa disposition pour limiter l'étendue de l'incident, rechercher la cause de celui-ci et l'éliminer.

Les outils à la disposition du CREG sont le TCO et le pupitre de commande. Outre les informations sur l'état de la ligne le TCO porte les boutons de télécommande des CS, des SIT et des appareils de voies autorisant le retournement d'un train. Le pupitre rassemble les différents moyens de téléphonie (THF, téléphones directs), les boutons de commande des DSO, un bouton de coupure générale d'urgence du courant (ou DG pour *disjonction générale*) plus d'autres commandes pour l'exploitation spécifique de chaque ligne (par exemple les télécommandes de dégivrage des rails sur les portions aériennes).

Suivant la gravité de l'incident le CREG est aidé d'un à trois assesseurs pour les incidents les plus importants. Les assesseurs sont ses collègues des autres lignes qui viennent décharger le CREG de la ligne concernée (alors appelé *chef d'incident*) de l'actionnement des différents outils, lui laissant la part décisionnelle et communicative de la gestion de l'incident.

Les CREG ont la possibilité de couper le courant immédiatement et sans intermédiaire sur toute la ligne, une section ou une sous-section, afin d'autoriser des actions à *pied d'œuvre* (c'est-à-dire sur place). Ils peuvent également retenir des trains en station par le biais des DSO. Ils ont aussi la possibilité d'organiser le retournement de trains en ligne. Grâce au THF, ils communiquent avec le conducteur d'une rame particulière ou avec les conducteurs de plusieurs rames (situées sur une même section ou tous les conducteurs actuellement sur la ligne). Ils peuvent également communiquer avec les chefs de station et les personnels à pied d'œuvre. Toutes ces communications sont utilisées pour informer les acteurs locaux de l'exploitation, mais également pour obtenir des informations précises sur la situation locale. De plus, les CREG sont en liaison directe avec l'IPEX pour la demande et la gestion de services extérieurs tels que les pompiers, la police...

Section 2 :

Tâche et activité des chefs de régulation

2.1 Tâche des chefs de régulation

Les CREG sont chargés de la régulation du trafic sur la ligne et de la gestion des incidents éventuels. Leur tâche se décompose en trois sous-tâches, en fonction de l'état du trafic : premièrement, lorsque le trafic est normal, les CREG doivent suivre l'évolution de celui-ci afin qu'il reste dans un certain intervalle de normalité ; deuxièmement, lorsque la situation se dégrade, ils doivent prendre des mesures de régulation pour ramener le trafic dans les intervalles normaux ; Enfin, en cas d'incident, ils doivent rechercher la cause de celui-ci et l'éliminer tout en surveillant le trafic sur le reste de la ligne (aussi appelé *trafic résiduel*).

Cette tâche s'apparente à la régulation d'un mode de transport guidé (train, tramway...), mais s'en distingue principalement par le fait que les trains sont omnibus, qu'ils vont à la même vitesse et que le but n'est pas de respecter des horaires, mais de maintenir un intervalle régulier entre les trains. Ces spécificités autorisent un certain nombre de stratégies de gestion de situations perturbées ou incidentelles, comme nous le montrons dans la section 3.

2.2 Activité des chefs de régulation

Cette tâche donne naissance à une activité de type *gestion de trafic* telle qu'elle est décrite par Cellier, Keyser et Valot [1996]. C'est un cas particulier de la *gestion d'environnements dynamiques* [Hoc, 1996]. L'activité de gestion de trafic se caractérise par une distance physique importante entre le contrôleur (généralement appelé *régulateur*⁶) et le système à contrôler (formé dans notre cas de la ligne, des trains...) : il n'y a pas de relation directe entre l'opérateur et le système à réguler. Le régulateur ne peut que communiquer avec les acteurs sur le terrain et télécommander certaines installations (ce qui amène un certain degré d'incertitude sur l'effectivité de ces commandes). Un autre caractère important de l'activité de gestion de trafic est l'importance de la dimension spatio-temporelle [Cellier, 1996] : l'objectif de la régulation du trafic est d'optimiser les déplacements de mobiles, ce qui mêle à la fois les positions, les vitesses et les durées liées à ces mobiles.

⁶ Le terme régulateur est à prendre ici dans son acception générale. Nous n'employons, dans cette sous-section, le terme CREG que pour les régulateurs du trafic d'une ligne de métro parisien, afin de différencier ce qui est spécifique à notre domaine de ce qui est plus généralement observable.

Selon Danielou [1986] et Spérandio [1993], l'activité de régulation de trafic se décompose en trois grandes activités : la *surveillance*, le *diagnostic de dysfonctionnement* et la *décision/planification*. En ce qui concerne l'activité des CREG en particulier, on observe des activités hybrides de ces trois grandes classes d'activités, assujetties à la situation.

Premièrement, en situation normale, le CREG surveille le trafic. La *surveillance* consiste en l'observation des intervalles entre les trains, à la régularité du trafic et des conditions de circulation. Dès qu'une déviance est observée, le CREG prend des mesures compensatoires (que nous analysons dans la section suivante) afin de maintenir un trafic régulier et fluide. Cette activité correspond à celle décrite par Samurçay et Hoc [1996] et Senach [1984].

Quand un incident se produit, le CREG mène de front deux activités : premièrement il cherche à déterminer puis éliminer les causes de l'incident (gestion de l'incident à proprement parler), deuxièmement il *gère le trafic résiduel*. Cette activité de gestion du trafic résiduel consiste à *surveiller* la position des trains situés en dehors de la zone d'influence de l'incident, leurs intervalles, leurs vitesses. Elle est en ce point semblable à l'activité de surveillance en temps normal. Le CREG prend également les mesures nécessaires pour éviter que les trains ne s'accumulent aux frontières de la zone d'influence de l'incident en maintenant un service voyageur correct et, si l'incident vient à durer, organise l'exploitation partielle des parties de lignes non concernées par l'incident en retournant des trains en ligne. Cette partie de l'activité est en relation avec les activités de *décision et planification* décrites par Bainbridge [1980] et Decortis et Cacciabue [1991]. La gestion du trafic résiduel par les CREG est donc une activité mêlant les activités « classiques » de *surveillance* et de *décision/planification* observée chez les régulateurs.

L'activité de *gestion de l'incident* est une composition de deux activités : le *diagnostic*, qui correspond à la recherche de la cause de l'incident, et la *planification* et l'élimination des causes de l'incident. Ces deux activités sont intimement imbriquées dans la gestion de l'incident par un CREG et ne sont pas identifiables l'une sans l'autre. En effet, le diagnostic dirige la décision et la planification (comme à l'accoutumée), mais également se fonde dans celle-ci. Des plans d'action sont partiellement dédiés au diagnostic lui-même et ce afin d'obtenir assez d'informations sur la situation à traiter. Le traitement de l'incident se fait conjointement au diagnostic, à tel point que dans de nombreux cas d'incidents le diagnostic n'est réellement établi qu'à la fin du traitement de l'incident. Nous ne pouvons pas dissocier le diagnostic de l'action et emploierons donc le terme de *plan de diagnostic/action* pour les décrire.

Nous venons d'établir le lien entre les activités des CREG et les activités habituellement rencontrées dans d'autres domaines liés à la régulation de trafic. Nous allons maintenant analyser cette activité dans le cas des CREG et plus particulièrement la gestion en exploitation

normale puis la gestion de situations incidentelles que nous avons à peine esquissées jusqu'ici.

2.3 Gestion en exploitation normale

En situation normale d'exploitation les CREG surveillent le trafic des trains grâce aux indications du TCO. Ils observent notamment la régularité de l'intervalle entre les différentes rames, afin d'atteindre l'objectif de régularité du service offert aux voyageurs.

Dès qu'une déviance est perceptible et constitue un risque de prise de retard, les opérateurs peuvent "jouer" sur l'intervalle entre les trains grâce au DSO. Ce dispositif consiste en un signal particulier, situé en tête de quai, adressé aux conducteurs. La mise en œuvre du DSO allume le signal obligeant le conducteur à rester à quai jusqu'à l'extinction du signal. Par ce moyen, le CREG augmente l'intervalle avec le train précédent et diminue l'intervalle avec le train suivant. En utilisant plusieurs DSO à bon escient le CREG est en mesure de contenir une déviance raisonnable et donc de maintenir une certaine régularité des rames sur la ligne.

2.4 Gestion d'une situation incidentelle

2.4.1 Qu'est-ce qu'une situation incidentelle ?

Une situation incidentelle se caractérise par la survenue d'un incident. Il s'agit d'une rupture d'exploitation pouvant être de nature diverse. Cette situation empêche la réalisation normale de l'offre de transport prévue et, si elle n'est pas prise en considération, bloque le trafic.

2.4.2 Evénement incidentel

Le premier élément caractéristique d'un incident (du point de vue du CREG) est l'événement lui indiquant qu'un incident vient de se produire. Nous qualifions cet événement d'*incidentel*. Nous distinguons cet événement des autres événements vécus au PCC car, pour un même type d'incident dans des conditions similaires, la nature de l'événement incidentel influe grandement sur les décisions qui vont suivre et sur le plan d'actions menées pour éliminer la cause de l'incident. Nous avons identifié 8 types d'événements incidentels :

1. La **Disjonction d'Intensité** (DI) correspond à une coupure automatique de courant sur une section, réalisée par des disjoncteurs d'intensité réagissant à tout court-circuit. Elle se manifeste par une alarme au PCC et l'allumage en rouge de la (les) section(s) concernée(s) au TCO. Le courant est rétabli automatiquement.

2. L'**Incident Ligne** (IL) correspond à une coupure du courant automatique sur une section suite à 3 DI. La section reste alors hors-tension.
3. La **Disjonction d'Alarme** (DA) correspond à une coupure du courant sur une (parfois deux) section(s) due à l'actionnement d'un rupteur d'alarme. Elle se manifeste au PCC par l'allumage en rouge de la (les) section(s) concernée(s) et d'un voyant au TCO.
4. La **demande de Disjonction Générale** (DG) correspond à un appel par la THF d'un conducteur demandant une coupure immédiate du courant sur l'ensemble de la ligne.
5. L'appel d'un conducteur informant qu'un **signal d'alarme** (KSA) a été actionné dans une voiture de sa rame. Le conducteur doit alors stationner pour réarmer le signal d'alarme et se renseigner sur la cause de l'actionnement.
6. L'**appel d'un conducteur par la THF** pour une autre raison que la demande de DG et l'actionnement d'un KSA.
7. Un **appel autre** que par le THF, par exemple un chef de secteur par radiotéléphonie ou un agent de station par téléphonie directe.
8. Le **stationnement** d'une rame (STA). Ce stationnement est visible sur le TCO et est apprécié par le CREG en observant les durées d'occupations des CDV.

La disjonction d'intensité, l'incident ligne, la disjonction d'alarme et le stationnement sont des événements incidentels observable par le CREG au TCO ; la demande de disjonction générale, l'appel pour actionnement d'un frein de secours et les autres appels par THF sont signalés par le conducteur concerné par le biais du THF. Nous avons extrait la demande de DG et l'appel pour KSA des autres appels par THF, car la gestion des incidents est fortement différenciée dès les premières secondes pour ces deux événements incidentels.

En général un seul événement incidentel est observé par incident, mais il peut arriver que plusieurs événements simultanés annoncent le même incident : ainsi une personne sur un quai peut tirer un rupteur d'alarme (réalisant ainsi une DA) en même temps qu'un conducteur demander une DG à la vue d'un voyageur descendant sur les voies. La conjonction de ces deux événements est différente de la survenue de l'un ou de l'autre seul. Elle se distingue d'une DA simple par le fait que le conducteur demandant la DG permet de connaître immédiatement la cause de la DA. Elle se distingue d'une simple demande de DG par le fait que la coupure de courant est déjà confirmée à pied d'œuvre par la DA.

Nous remarquons ici que cette liste peut être complétée le cas échéant lors d'ajout de nouvelles fonctionnalités dans le PCC. Ainsi cette liste, réalisée à partir des données des PCC de génération précédente, doit être complétée par quelques alarmes arrivant directement au PCC nouvelle génération de la ligne 4 telles que l'**alarme pneu porteur** et l'**alarme frotteurs**.

Ces deux alarmes préviennent les chefs de régulation et les chefs de départs que le train se présentant en terminus a un type particulier d'avarie (concernant les pneus ou les frotteurs, en fonction de l'alarme), leur permettant ainsi de prendre immédiatement des mesures appropriées (l'échange de matériel avant de faire tourner le train ou la prévision d'un tel échange dans l'autre terminus). La liste des événements incidentels est fixe pour un système d'exploitation donné (ensemble formé par : la ligne, les ressources, les outils à disposition du contrôleur d'exploitation, les outils de communications, les systèmes de détection des anomalies...). Tout changement ou ajout de nouveaux éléments est susceptible de modifier cette liste.

2.4.3 Gestion en deux phases

Dès que le CREG est averti de l'occurrence d'un incident, celui-ci prend les mesures nécessaires à l'élimination des causes de l'incident. Suivant le type d'événement incidentel le CREG prend des mesures immédiates. Par exemple, en cas de demande de DG, il coupe immédiatement le courant de traction sur toute la ligne, avant même de se renseigner sur la source de la demande ou même du bien-fondé de celle-ci. En effet, il est préférable de couper l'alimentation alors que ceci n'était pas nécessaire plutôt que l'inverse. Les premières mesures d'urgence étant mises en place, le CREG recherche la cause de l'incident.

2.4.3.1 Phase de diagnostic et de gestion dans l'incertain

La recherche de la cause de l'incident est plus ou moins complexe. Certains événements incidentels sont accompagnés de messages permettant de l'identifier rapidement, notamment les appels venant du terrain (d'un conducteur par le THF, d'un agent de station par le téléphone directe ou d'un gradé de la ligne par radiotéléphonie). En général les objets de ces appels contiennent l'information nécessaire à l'identification de la cause de l'incident.

Les autres événements incidentels (la disjonction d'intensité, l'incident ligne, la disjonction d'alarme et le stationnement d'une rame) ne sont pas accompagnés de messages. Le CREG doit alors interroger les agents sur le terrain pour obtenir des informations permettant de cerner la cause de l'incident. Cette ***phase de diagnostic*** est plus ou moins longue. Les trains étant très rapprochés, le CREG n'a que quelques instants pour réagir : en heure d'affluence l'intervalle entre les trains est de 90 secondes, le CREG a donc environ une minute de marge avant de prendre les premières mesures de gestion de l'incident. Si la cause de l'incident n'est pas identifiée à temps, le CREG doit commencer à gérer l'incident sans connaître la cause de celui-ci. Nous parlons alors de ***gestion dans l'incertain***.

La phase de gestion dans l'incertain peut durer jusqu'à la fin de la gestion de l'incident. L'incident restera alors de cause inconnue. En règle générale, la cause est toutefois identifiée

« après coup ». Dès que le CREG en prend connaissance, il passe dans la seconde phase de gestion de l'incident.

2.4.3.2 Phase de gestion d'un incident de type connu

Un incident de type connu est plus simple à gérer, car les mesures sont ciblées et de nombreuses procédures précises ont été établies. Le CREG est en mesure d'éliminer directement la cause de l'incident ou de minimiser les conséquences sur le trafic. Il ne faut pas oublier que le but premier d'une ligne de métro est le transport de voyageur et que l'élimination de certaines causes d'incidents ont plus de conséquences néfastes sur l'offre de transport que d'autres stratégies telles que le rapatriement en terminus d'une rame préalablement vidée de ses voyageurs. C'est le cas notamment pour les problèmes de portes, de motrices inactives ou de freinage, suivant la situation.

2.4.4 Retour à la normale

Une fois l'incident maîtrisé, le CREG doit rétablir un trafic fluide sur la ligne. En effet, les mesures prises pour éliminer la cause ou la source de l'incident ont des conséquences sur le trafic résiduel. Des trains ont été arrêtés plus ou moins longtemps, des stations temporairement non desservies, les trains ne sont plus répartis de manière homogène sur la ligne et des rames sont surchargées, rendant le service voyageur moins efficace et donc augmentant encore le retard.

Le CREG a deux principaux moyens d'action sur la régularité du trafic. Premièrement il peut retenir un train en station sous DSO afin d'augmenter l'intervalle avec le train précédent et diminuer l'intervalle avec le train suivant. Quand l'intervalle entre deux trains est trop important et que les conditions sont favorables, le CREG peut également retourner un train en ligne, le faisant passer d'une voie à l'autre, en établissant un **Service Provisoire** (SP). Ceci est une stratégie payante surtout quand une grande dissymétrie s'est installée entre les deux voies : l'une étant surchargée (de nombreux trains y circulent) l'autre ayant un fort déficit de trains. L'établissement d'un service provisoire rééquilibre le trafic en réorganisant la répartition des trains sur les deux voies.

En combinant ces deux types d'actions, le CREG est en mesure de rétablir un service nominal sur la ligne. En fonction de la durée de l'incident, le CREG a déjà pu prendre des mesures pour limiter les conséquences de l'incident sur le trafic résiduel. En parallèle des deux phases de gestion de l'incident proprement dit, le CREG gère le trafic résiduel.

Pendant cette phase de retour à la normale, le CREG doit également rédiger son rapport d'incident. Ce rapport retrace l'ensemble des événements et mesures prises lors de la gestion

de l'incident. En ce qui concerne la gestion du trafic résiduel, seul l'établissement des SP est noté. Ce rapport journalier est archivé et mis à disposition de différents services de l'entreprise. Nous pouvons représenter l'activité des CREG en situation d'incident par la Figure 2-3.

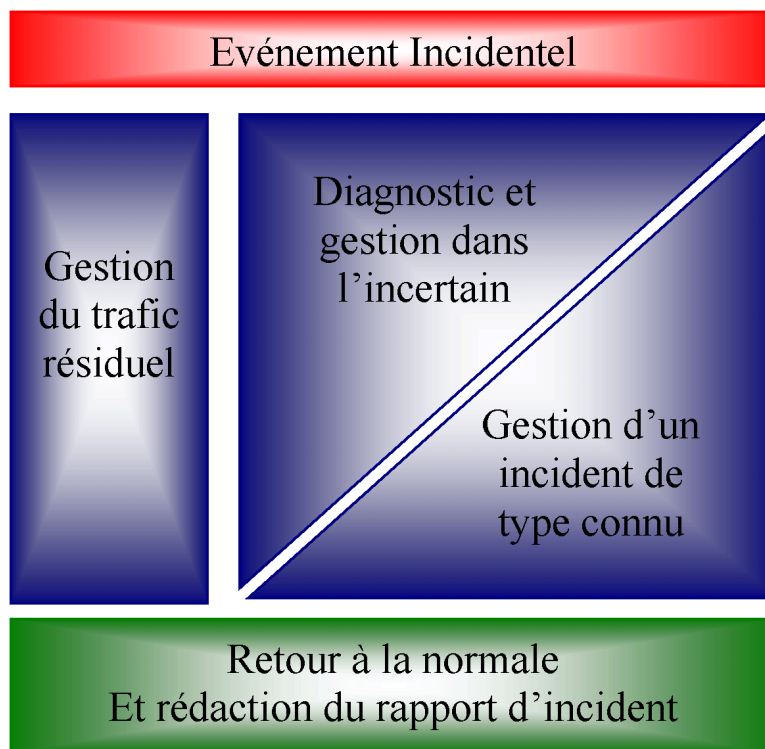


Figure 2-3: Phases de gestion d'un incident

Section 3 :

Rôle du contexte dans la gestion d'incidents

Pour gérer les incidents, de nombreuses procédures ont été développées par l'entreprise. Ces procédures sont générales et ne tiennent pas compte des particularités d'un incident précis. Le CREG doit adapter ces procédures à la situation réelle, en veillant à respecter les consignes de sécurité ainsi que les obligations techniques liées aux infrastructures et aux matériels. Le contexte joue donc un rôle important dans la gestion des incidents. Nous présentons dans cette section l'importance du contexte dans la gestion d'incidents. Dans un premier temps, nous étudions le contexte à l'instant où un incident survient. Nous verrons ensuite comment ce contexte évolue au cours de la gestion de cet incident. La dernière partie est consacrée à l'adaptation des procédures au contexte de l'incident.

3.1 Contexte de l'incident

L'heure, le lieu, l'événement incidentel, l'état du trafic sur la ligne, sa déviance par rapport à la situation d'exploitation attendue à cet instant sont autant d'éléments capitaux qui influent sur les choix réalisés par le CREG pour gérer un incident. Ces éléments, avec de nombreux autres, forment le *contexte de l'incident*. Le contexte de l'exploitation évolue continuellement et s'enrichit des différents événements survenus au cours de la journée. Nous appelons ce contexte : *contexte nominal*. Le contexte de l'incident peut être comparé à une image du contexte nominal à l'instant où l'incident se produit.

Tant que l'exploitation est normale, le contexte nominal évolue continûment et le CREG maintient à chaque instant ses connaissances sur la situation. Il garde à l'esprit le jour (jour ouvrable ou week-end, en période de vacances ou non...), l'heure, et donc le type d'exploitation attendue (heure pleine ou creuse, garage ou dégarage de trains en terminus...), l'affluence attendue, la charge des trains, leur répartition sur les deux voies de la ligne, les incidents passés, les particularités de la ligne, de l'exploitation...

Dès qu'un incident survient, le CREG, qui, rappelons le, n'a que quelques instants pour réagir à l'annonce d'un incident, consacre toute son attention à l'incident, à la recherche de la cause de celui-ci et à l'élaboration d'une stratégie pour le gérer. Il ne peut plus suivre, pendant un court instant, l'évolution du contexte nominal. Les décisions qui vont être prises le sont donc en fonction de l'image du contexte arrêtée à l'instant où l'incident est annoncé. Le contexte de l'incident joue donc un rôle primordial sur les premiers instants et sur l'orientation de la stratégie construite par le CREG à ce moment précis.

C'est ainsi que deux incidents de même type survenant dans des contextes différents peuvent être gérés de manières très différentes. C'est le cas par exemple de problèmes de non-fermeture de portes. Dans un contexte de forte affluence, cet incident sera généralement traité par l'élimination de la rame du trafic voyageur en évacuant les voyageurs et la faisant rentrer Haut Le Pied⁷ (HLP) en terminus ; alors que si l'affluence est plus faible, la porte sera fermée manuellement puis condamnée par le conducteur pour reprendre le service voyageur. Le choix stratégique, qui à première vue semble illogique (perte d'une unité de transport en cas d'affluence et maintient en heure creuse), est en réalité basé sur la capacité de transport non pas d'une rame, mais de l'ensemble du système. Ainsi il est préférable d'éliminer de

⁷ Envoyer une rame Haut Le Pied consiste à évacuer les voyageurs de la rame et de faire rentrer cette rame au prochain terminus sans marquer l'arrêt aux stations sur le trajet. Cette procédure est fréquemment utilisée pour les problèmes sur le matériel roulant.

l'exploitation une rame posant problème en heure de pointe afin de fluidifier le trafic. La perte de la capacité de transport est alors largement comblée par une régularité plus importante des rames restant en exploitation. A l'inverse, la suppression d'une rame en heure creuse est plus contraignante, car le nombre de rames en exploitation est inférieur et la capacité de transport de la rame avariée ne peut pas être comblée. De plus, la faible affluence permet de mieux gérer la redirection des voyageurs vers les portes fonctionnant correctement.

Le contexte de l'incident joue donc un rôle important sur les grandes orientations adoptées par le CREG pour la gestion de l'incident. Toutefois, le contexte continue à évoluer pendant un incident.

3.2 Gestion de l'incident et évolution du contexte

Plus on avance dans la gestion de l'incident, plus le contexte nominal devient prépondérant par rapport au contexte de l'incident. Le contexte nominal continue d'évoluer et contient de plus en plus d'informations concernant l'incident et sa gestion. Il archive notamment l'historique des coupures de l'alimentation, des confirmations à pied d'œuvre de ses coupures, la position des trains et leurs particularités... Toutes ces informations, conséquences des décisions prises par le CREG pour gérer l'incident sont postérieures à l'occurrence de l'incident et ne sont donc pas des éléments du contexte de l'incident.

Une fois passée la période de forte charge suite à un incident (dont la durée dépend de la nature de l'incident, de l'événement incidentel et du contexte de l'incident), le CREG peut de nouveau prendre le temps suivre l'évolution du contexte nominal de manière à affiner sa stratégie en fonction de la situation réelle. Ceci est d'autant plus important pour la gestion du trafic résiduel. Certaines actions, nécessaires à l'élimination de la cause d'un incident ont des conséquences sur le trafic de partie ou de l'ensemble de la ligne. L'état du trafic peut être fortement dégradé lors de la gestion d'un incident. L'élimination de la cause étant prioritaire, le trafic résiduel est relégué au second plan, même si des mesures préventives et/ou palliatives sont prises en parallèle de la gestion proprement dite. Le contexte nominal est alors un élément capital pour la gestion de l'incident, mais surtout pour la régulation du trafic résiduel.

Pour la gestion d'un incident, par exemple, dès que le courant de traction a été coupé, le contexte nominal intervient fortement à chaque instant. Il est en effet capital pour la sécurité des voyageurs et des personnels à pied d'œuvre de connaître très précisément l'état de l'alimentation de la traction : alimentée, coupée, coupée avec confirmation à pied d'œuvre, sections ou sous-sections concernées par les coupures... Ces informations sont postérieures à l'événement incidentel et sont donc parties intégrantes du contexte nominal. Evidemment, dans cet exemple, puisque le CREG est la source (sinon le réalisateur) des actions de gestion

du courant, il est parfaitement informé en temps réel de l'état exact de l'alimentation de traction.

La distance (matérielle et opératoire) entre le PCC et le terrain, complétée par le passage obligé par des intervenants à pied d'œuvre pour de nombreuses actions, fait que le contexte nominal n'est pas toujours immédiatement ou directement connu du CREG. Ainsi une personne sur le terrain peut réaliser des actions non demandées par le CREG qui auront des conséquences sur la gestion de l'incident et/ou du trafic résiduel. Dans ce cas, les éléments de contexte nominal qui seraient modifiés ne sont pas forcément connus du CREG.

3.3 Procédures et Pratiques réelles

Le contexte joue un rôle prépondérant pour la gestion des incidents. Certains éléments de contexte, les plus techniques en général, sont intégrés dans les procédures (voir le Premier chapitre, Section 3.3.1, page 40). En effet, les *procédures* déclinent les opérations à réaliser pour atteindre un but en fonction de la valeur de certains paramètres ou variables du système ou d'une partie du système (par exemple état de l'alimentation, d'une rame, d'une motrice...), dans des conditions de sécurité optimales. Ces procédures sont générales, et ne tiennent pas compte de la richesse des diverses situations pouvant être rencontrées par les opérateurs.

Nous parlons ici d'opérateurs plutôt que de CREG, car le phénomène dépasse largement le cadre de leur métier et même du domaine. En effet, dans de nombreux domaines, nous pouvons observer une multiplication et une spécialisation des procédures, à tel point que, dans le domaine de la gestion d'incidents aéronautiques, les opérateurs préfèrent reconstruire une stratégie à partir d'une procédure plutôt que de l'appliquer à la lettre, ceci pour rester bien attentif aux particularités de la situation [De Brito et Boy, 1999].

Les procédures sont des formes de connaissances compilées issues de l'expérience centenaire de l'entreprise. Elles se sont construites petit à petit grâce à la répétition des incidents et à l'amélioration des stratégies appliquées pour les résoudre. Elles ont été ensuite formalisées et unifiées par l'entreprise. Une procédure est ainsi une simplification des stratégies obtenue par élimination des éléments de contexte secondaires afin de créer une stratégie applicable, pour un incident donné, dans toutes les situations actuellement connues. Le principe même d'élaboration des procédures rend impossible la prise en compte de la richesse du contexte.

Comme dans l'aéronautique, les CREG préfèrent reconstruire une stratégie plutôt que d'appliquer une procédure à la lettre. Ainsi les pratiques, qui sont les raisonnements et les actions réellement réalisées pour gérer l'incident, sont quelque peu différentes des procédures officielles. Ces pratiques sont à la base du cycle d'amélioration des procédures, et sont donc

importante pour l'entreprise. En effet, les CREG connaissent parfaitement leur métier et sont les mieux placés pour proposer de nouvelles voies de gestion des incidents, du moment qu'elles respectent entièrement les règles de sécurité. De plus des situations non prévues par les procédures peuvent émerger, le CREG devant alors s'inspirer des procédures connues pour construire une stratégie nouvelle.

Les procédures et les pratiques entretiennent des relations particulières : les premières sont issues des pratiques précédemment appliquées ; les secondes étant le résultat de l'activité des opérateurs guidée par les procédures.

Section 4 :

Résultats de l'analyse de l'activité

Dans un premier temps [Pasquier, 1998] nous avons analysé l'activité des chefs de régulation. Nous avons observé les CREG dans leurs situations de travail et les avons interrogés sur leurs actions, sur les éléments appuyant leurs décisions, sur les raisonnements qu'ils ont suivis, sur les procédures officielles et les pratiques réelles. Nous leur avons également demandé de répondre à un questionnaire, afin de mieux les connaître, leur formation, leur expérience, leurs attentes vis-à-vis d'un système d'aide à la régulation du trafic et à la gestion d'incident [Brézillon, Pasquier et Saker, 1999b]. Ces premiers travaux nous ont permis de cibler le système d'aide par rapport aux attentes des utilisateurs et d'établir un type de modélisation compréhensible par les opérateurs et facilement traitable par le système. Dans un second temps, nous avons travaillé avec Zanarelli [Zanarelli, Saker et Pasquier, 1999 ; Pasquier et Zanarelli, 2001], doctorante en ergonomie cognitive à la RATP, dont l'étude porte sur l'analyse de l'activité des chefs de régulation et des chefs de départ, en situation incidentelle ou non, et de leurs interactions [Zanarelli, 2002].

Nous résumons, dans cette section, les résultats obtenus dans cette phase d'analyse de l'activité des CREG. Nous montrons dans un premier temps l'importance de l'expérience de CREG pour mener à bien leur tâche, ainsi que du retour d'expérience, qui permet de maintenir leur haute performance. Nous présentons ensuite deux concepts issus de l'ergonomie cognitive qui sont nécessaires pour la description de l'activité de CREG : les concepts pragmatiques et les schèmes d'action.

4.1 Importance de l'expérience et du retour d'expérience

Les CREG, en ce qui concerne le métro parisien, ont une formation initiale principalement technique. Après plusieurs années sur le terrain en tant qu'agent de maîtrise polyvalent (conducteur, agent de manœuvre...), le futur CREG suit une formation à la régulation de trafic et à la gestion d'incident sur les lignes de métro. Pendant cette formation, les CREG apprennent les procédures puis les mettent en pratique sur un simulateur de ligne, réplique conforme des PCC actuels. Les futurs CREG sont également en alternance au PCC. Après cette période de formation, les nouveaux CREG ont en charge la gestion d'une ligne et sont aidés par un de leurs collègues pendant les heures d'exploitation voyageur. Cette formation initiale fournit aux CREG les éléments nécessaires à la régulation de la ligne et à la gestion des incidents.

Cette formation initiale est complétée par l'expérience acquise pendant leur service. Chaque nouvel incident est l'occasion de se perfectionner. Les assesseurs⁸ jouent un rôle capital dans cette formation continue. Pendant l'incident les CREG échangent leurs points de vues, proposent diverses solutions possibles. Le chef d'incident élabore sa stratégie à partir des procédures, de ses expériences et de celles de ses assesseurs. Une autre période importante d'échange d'expérience entre CREG a lieu après l'incident, et surtout lors des relèves. Outre le résumé des incidents d'exploitation ayant eu lieu les CREG se relevant échangent des anecdotes sur des incidents particuliers ou des situations spécifiques. Cet échange, conforté par le double système de rotation des CREG. Les CREG sont répartis en plusieurs équipes, de deux types : premièrement les repos dit 5/2 correspondant à un service de 5 jours en matin (5h-12h) ou en soir (12h-19h), suivis de deux jours de repos ; deuxièmement les repos 6/4 correspondant à un service de 6 jours en matin (6h-12h), en soir (12h-19h) ou en nuit (19h-5h), suivis de 4 jours de repos. Des équipes de périodicité différentes se relayent au PCC, ce qui permet une diffusion rapide des incidents et situations importants récemment rencontrés

L'ensemble de l'expérience acquise par les CREG, lors de la formation initiale, lors des incidents qu'ils ont vécus ou lors d'échanges entre collègues, garantit l'autonomie décisionnelle laissée aux CREG. Ainsi les CREG sont des experts de haut niveau en régulation de trafic et en gestion d'incident.

⁸ Lors d'un incident, un à trois (suivant l'importance de l'incident) CREG des lignes voisines viennent aider le CREG de la ligne concernée, et sont alors appelés assesseurs, le CREG de la ligne concernée prenant le titre de chef d'incident. Le rôle des assesseurs est de surveiller et de conseiller le chef d'incident et d'actionner les commandes au TCO sur ordre du chef d'incident.

4.2 Concepts pragmatiques

L'expertise des CREG est difficile à cerner. Zanarelli [2002] a étudié l'activité des CREG, entre autres, en situation de travail nominale et incidentelle. Ses études sont basées sur l'analyse et l'auto-confrontation des CREG avec des enregistrements vidéo de périodes de gestion d'incidents ou de fortes perturbations de trafic qu'ils ont eu à gérer.

Ces études font ressortir deux éléments qui sont essentiels en ce qui nous concerne. Le premier de ses éléments est la notion de *concepts pragmatiques*. Il s'agit d'objets mentaux construits par les CREG qui leur permettent de se faire une idée de la situation présente, à partir d'éléments plus ou moins techniques dont ils ont connaissance.

Il en est ainsi du concept de *fluidité*. Aucun outil à la disposition des CREG ne présente la fluidité du trafic. Ils se construisent une image mentale de la fluidité en combinant divers éléments de la situation : l'heure (et surtout le type de marche suivie par les rames), le nombre de rames en ligne, les intervalles entre ces rames, les vitesses relatives des rames... Aucun outil ne leur donne cette information, pourtant c'est en grâce à la fluidité que le CREG sait si le trafic est perturbé ou non.

Les concepts pragmatiques sont mal identifiés par l'entreprise et ne sont donc pas enseignés lors de la formation initiale des CREG. Ces concepts sont construits petit à petit par chacun d'eux, en fonction de son expérience et de sa subjectivité. Il est donc très difficile de prendre en compte ses concepts qui sont pourtant à la base de nombreuses décisions, notamment en ce qui concerne la gestion du trafic résiduel.

La notion de concepts pragmatiques est importante car elle permet d'accéder à des éléments support des décisions des CREG alors qu'elles ne sont référencées en aucun lieu. Les CREG utilisent des concepts pragmatiques pour établir leurs stratégies de gestion de situations perturbées ou incidentelles. Il est donc capital pour un système d'aide intelligent s'adressant à eux de connaître l'existence de tels concepts et de pouvoir les intégrer dans ses modèles de connaissances stratégiques.

4.3 Schèmes d'action

Le deuxième concept mis en évidence au PCC suite à l'analyse de l'activité des chefs de régulation par Zanarelli [1999] est celui des *schèmes d'action* (nous reviendrons plus en détail sur cette notion dans le chapitre 4, section 2.5, page 122). Les schèmes d'action sont des unités mentales guidant l'action. Ils organisent l'activité des personnes en fonction de buts à atteindre, d'outils à leur disposition, de la situation. Ce qu'il est important de savoir sur cette notion pour les propos que nous tenons dans cette section est qu'un schème d'action est

composé d'une structure d'actions permettant d'atteindre un but, et des moyens utilisés pour réaliser ces actions. La structure d'actions peut également faire appel à d'autres schèmes pour réaliser des sous-buts. Vergnaud [1985] décrit les schèmes d'action comme étant un tout organisé de manière dynamique. Selon lui les schèmes d'action sont composés de quatre éléments : 1) des invariants opératoires : objets, propriétés, relations et processus ; 2) des règles d'action, qui créent et guide l'action ; 3) des inférences et des calculs et 4) des prédictions, qui sont des buts ou des étapes intermédiaires attendues.

Les schèmes d'action identifiés dans la communauté de pratique des CREG correspondent à divers types d'activités. Premièrement nous observons des schèmes de gestion de situations dégradées. Ces schèmes guident des actions de régulation du trafic telles que la mise en place et le retrait de DSO ou le retournement de trains en ligne (Service Provisoire), en fonction de la situation. Les schèmes de gestion de situations incidentelles font appel à des schèmes de gestion du trafic résiduel et à une recherche de la cause de l'incident. Quand la cause est identifiée par le CREG, celui-ci prend des mesures ciblées et organise son activité autour de schèmes spécifiques à la cause de l'incident.

La notion de schème nous intéresse tout particulièrement car elle représente une structuration mentale de l'activité. Il nous semble donc important, dans l'optique de la réalisation d'un outil d'aide intelligent, de nous pencher sur cette notion et sur les fonctionnements psychologiques sous-tendant l'activité des CREG. L'avantage principal de cette notion est que la structure organise des actions en fonction du contexte et que ces structures sont construites autour de leur évolution.

Section 5 :

Systèmes d'aide à la gestion du trafic à la RATP

Tout au long de ce premier siècle du métro, les entreprises qui ont eu en charge son exploitation ont cherché à gérer de mieux en mieux les incidents qui s'y produisent, construisant, dans un premier temps, des postes de gestion locale du courant de traction et organisant le trafic grâce à des signaux lumineux transmis de station en station par les chefs de gares. Puis, dans les années 1960, arrive la centralisation de la régulation des rames en ligne et de la gestion des incidents grâce aux outils installés dans le PCC, créant ainsi le métier de chef de régulation et les activités que nous venons d'étudier.

Ce métier est déjà fortement assisté par des outils, pas toujours informatiques mais au moins automatiques et/ou électroniques. Ainsi le suivi des trains en ligne est en fait calculé (et enregistré) par un ordinateur central à partir des données brutes sur l'occupation des CDV à

chaque instant. Deux trains ne pouvant pas partager le même CDV et un train ne pouvant se déplacer d'un CDV qu'en direction d'un CDV contigu, l'algorithme est simple mais efficace.

Petit à petit, à mesure que l'informatique se développait, des systèmes ont été ajoutés dans l'environnement des chefs de régulation. Ces systèmes n'interviennent pas directement sur l'activité de ceux-ci. Il s'agit principalement de trois progrès : premièrement la rédaction du rapport d'incident a été simplifiée par le développement d'un système de saisie de type « texte à trou », d'une base de données sur les incidents accessibles à divers acteurs de l'entreprise et d'un système d'information voyageur centralisée en liaison directe avec les PCC. Ces apports, toutefois, ne concernent pas l'activité principale des chefs de régulation, mais des activités secondaires.

La décentralisation du PCC de la ligne 4 et la relocalisation au plus proche de la ligne combinées à l'agrégation avec les poste des chefs de départ des deux terminus de la ligne a été l'occasion de modifications beaucoup plus importantes (voire sans commune mesure). D'une part les deux métiers concernés (chef de régulation et chef de départ) se trouvent modifiés. Les trois agents en charge du respect du programme d'exploitation, qui étaient auparavant séparés, se trouvent dans la même pièce. Ainsi chacun sait ce que les autres font ou projettent de faire et peuvent désormais agir de concert, rendant leur travail plus efficace. Outre cet aspect organisationnel, les outils proposés sont foncièrement différents.

Le nouveau PCC reprend les principales caractéristiques des PCC précédents, en remplaçant toutefois la console électromécanique par une interface informatisée. Le système informatique à disposition des agents rassemble les commandes précédemment placées sur la console et celles situées sur le TCO. Une image globale de la ligne, de son état d'alimentation, de la position et des identifications des trains est toutefois présente à des fins de mémoire collective de travail. Tout a été fait pour que les opérateurs retrouvent l'ambiance de travail précédente, mais ceci se limite à l'apparence.

Des avancées significatives ont été réalisées concernant l'aide à la régulation de trafic. En ce qui concerne d'abord la gestion des départs, le système calcule automatiquement les rames à expédier, les agents qui les conduiront et les heures de départ en fonction du programme théorique de la journée, des disponibilités des différentes ressources, des temps d'arrêt des conducteurs, des rames arrivant dans le terminus sur l'autre voie et de divers paramètres configurables par le chef de départ (par exemple le battement minimum à respecter). Pour la régulation de trafic en ligne, le système régule les rames en fonction soit de leur écart par rapport à leur horaire théorique, soit par rapport aux intervalles, soit une composition des deux. Ce système de régulation automatique retient également automatiquement tous les trains en station en cas de coupure de courant sur une section ou sur l'ensemble de la ligne, autour des zones de manœuvre en cas d'activation d'une commande de service provisoire...

Cette nouvelle génération de PCC est inspirée, pour certains points, du PCC de la ligne 14 (METEOR), à rames automatisées. Le PCC de METEOR ne gérant pas de conducteurs, la gestion des départs est foncièrement différente (et originale) au nouveau PCC de la ligne 4. Pour la gestion des trains en lignes, par contre, une bonne partie du système et des algorithmes a pu être réutilisé.

Section 6 :

Conclusion

Le chemin de fer métropolitain de Paris est un moyen de transport urbain de voyageurs de grand débit. Riche d'un siècle d'histoire et d'exploitation, le métro subit régulièrement des incidents, fort heureusement mineurs pour la plupart. La Compagnie du Métropolitain de Paris a rapidement mis en place la structure que nous connaissons aujourd'hui, ainsi que les procédures de gestion des incidents qui les accompagnent. La RATP, héritière de l'exploitation du métro, continue cette œuvre en s'appuyant sur l'expérience des agents chargés de la régulation du trafic et de la gestion des incidents.

Ces agents, ayant une forte expérience du terrain et connaissant les procédures, doivent jongler sans cesse entre le prescrit et la réalité. L'entreprise compte sur eux pour gérer au mieux les perturbations et incidents survenant sur la ligne qu'ils ont en charge, et leur laisse une grande liberté décisionnelle. Ces agents doivent concilier les procédures, souvent très techniques et peu regardantes des détails de la situation, et le contexte actuel de l'exploitation.

L'activité des chefs de régulation est ainsi fortement orientée vers la mentalisation du contexte et l'intégration des diverses données qu'ils reçoivent soit des outils à leur disposition (du tableau de contrôle optique notamment), soit du terrain (conducteurs, gradés de ligne, agent de station), soit d'autres agents (inspecteur principal d'exploitation, autres chefs de régulation). Ce processus construit et tient à jour les concepts pragmatiques, images mentales du système à réguler et de son contexte nominal d'exploitation. Ces concepts pragmatiques sont directement utiles au chef de régulation lors de situations perturbées et plus encore lors d'incident d'exploitation. Ils complètent ainsi le contexte de l'incident et offre des éléments supplémentaires pour les prises de décisions.

Aux choix difficiles s'ajoute la complexité de la tâche demandée lors d'un incident. Le chef de régulation doit non seulement identifier rapidement et traiter les causes de l'incident, mais également gérer le trafic résiduel. Ses choix sont guidés principalement par le souci du respect des règles strictes de sécurité et de la minimisation des conséquences sur la qualité de service rendu aux voyageurs. De plus, les actions sur le terrain n'étant pas toujours suivies des

conséquences attendues, le chef de régulation peut envisager et mener de front plusieurs stratégies.

Ces stratégies sont construites sur la base des procédures enseignées et surtout de l'expérience acquise sur le terrain et auprès des collègues. Ces stratégies sont fortement liées au contexte de l'incident et tentent d'en extraire toute la richesse. On observe alors l'émergence de schèmes d'action. Ces structures cognitives organisent l'activité et les choix décisionnels en tenant compte du contexte, des outils à disposition, de règles d'action sur ces outils, d'inférences sur les buts et réalisations intermédiaires.

Les pratiques réelles appliquées lors d'incident sont ensuite étudiées pour le retour d'expérience et pour l'amélioration des procédures. Les procédures et les pratiques sont donc les supports d'un cycle d'évolution continu des stratégies de gestion des incidents, dont les chefs de régulation sont le principal moteur.

Troisième chapitre :

Le modèle de Graphes Contextuels

La richesse du contexte est difficilement prise en compte par les systèmes artificiels, notamment des outils informatiques. Ces outils, peu adaptés à la diversité des situations rencontrées en réalité, ont des difficultés d'intégration dans un système complexe tel que le métro. A l'occasion de la conception d'un SAIC dédié à la régulation du trafic d'une ligne de métro, la notion de contexte a été introduite dans le modèle des objets du domaine. Nous proposons maintenant d'inclure cette notion fondamentale dans les modèles de raisonnements.

La structuration du raisonnement humain a souvent été basée sur des règles, organisées en bases de règles. Ce modèle des raisonnements humains pêche sur de nombreux points, notamment sur la maintenance difficile des bases de règles. Une structuration de ces bases peut être réalisée par l'utilisation d'arbres de décision particuliers. Toutefois, la finesse du contexte multiplie rapidement la taille des arbres construits dès que le modèle est appliqué à des systèmes réels.

Nous proposons un type particulier de graphes pour modéliser les raisonnements humains fortement dépendant du contexte, tels que ceux observés lors de la gestion d'incidents sur une ligne de métro. Ces graphes, orientés acycliques, offrent des garanties de finitude des algorithmes. Inspirés des arbres sus-cités, ils en conservent la facilité d'utilisation tout en limitant l'explosion combinatoire. Nous les appelons *graphes contextuels*.

Nous présentons dans ce chapitre le modèle des graphes contextuels : ses éléments constitutifs et sa structure. Nous présentons également un algorithme d'acquisition incrémentale de connaissances adapté aux graphes contextuels, permettant ainsi de compléter un raisonnement par l'introduction de nouvelles stratégies possibles. Nous esquissons enfin l'utilisation des graphes contextuels pour la représentation d'activités complexes.

Section 1 :

Motivations et historique de la construction du modèle

Les Systèmes d'Aide Intelligent en Contexte sont issus des Systèmes Interactifs d'Aide à la Décision. Ces derniers sont basés sur des modèles support des raisonnements à la fois des décideurs et des systèmes d'aide (voir le premier chapitre). Le cas des décisions prises par les

chefs de régulation (CREG) dans les postes de contrôle des lignes de métro de Paris est très particulier. La quantité d'informations contextuelles intervenant lors de la prise de décision est considérable et les opérateurs construisent, pour chaque incident, un plan de gestion à partir des procédures, de leurs expériences et de celles de leurs collègues. Ainsi chaque incident aboutit à une pratique de la part des CREG (voir deuxième chapitre).

La prise en compte des pratiques réelles par un système d'aide se heurte à l'aspect statique des modélisations des connaissances de niveau stratégique actuellement proposées par les méthodologies de conception de systèmes à base de connaissances. Les modèles pouvant être obtenus par des méthodes telles que KADS sont constitués de modules de raisonnement fixés dès la conception du système, pouvant être organisés de manière plus ou moins dynamique. Hors, dans notre cas, ces modules guidant le raisonnement évoluent avec l'expérience des utilisateurs. L'aspect dynamique de l'organisation de ces éléments dans les modèles actuellement proposés n'autorise pas l'ajout de nouvelles stratégies d'organisation dynamique.

Les pratiques des CREG, quand elles apportent une nouveauté stratégique, modifient le modèle de raisonnement des opérateurs. Il est donc logique de répercuter ces modifications dans les modèles de raisonnement des outils d'aide. Pour cela il est nécessaire de créer un modèle de « modèles de raisonnements », autrement dit une modélisation dynamique des connaissances de niveau stratégique.

Construction du modèle

Dans un premier temps nous avons analysé différentes stratégies de résolution d'un même incident afin de percevoir comment celles-ci s'organisent les unes par rapport aux autres. A partir des procédures officielles et des récits d'expériences des CREG nous avons construit un arbre représentant les actions à réaliser pour gérer l'incident en fonction d'éléments du contexte (Figure 3–1). Cet arbre, issu des arbres de décision [Raïffa, 1968], représente les différentes manières de gérer un incident particulier. Il se lit de gauche à droite. Les rectangles représentent des actions réalisées par le CREG ou des agents à pied d'œuvre (conducteurs, cadres d'exploitation...). Les cercles représentent des divergences entre les stratégies (nœuds de décision pour les CREG) où un élément de la situation est analysé pour choisir entre les différentes stratégies de gestion possibles.

Le cas présenté ci-dessous (Figure 3–1) correspond à une rame en détresse ayant une ou plusieurs motrice(s) inactive(s). Cet exemple est le fil conducteur de ce chapitre : toutes les structures présentées par la suite représentent le processus de décision associé à sa gestion. La

signification des sigles des actions et des éléments contextuels est donnée respectivement dans le Tableau 3-1 et le Tableau 3-2. Un détail des stratégies représentées suit la Figure 3-1.

1	Régulation du trafic résiduel
2	Reprise du service voyageurs
3	Reprise du service voyageur jusqu'à la prochaine forte déclivité
4	Départ du train sans voyageurs
5	Garer le train en terminus
6	Réinitialiser le train
7	Evacuation des voyageurs du train avarié
8	Evacuation des voyageurs du train de secours
9	Evacuation des voyageurs du train avarié par les portes d'intercommunication
10	Evacuation des voyageurs du train de secours par les portes d'intercommunication
11	Evacuation des voyageurs du train avarié par les voies
12	Evacuation des voyageurs du train de secours par les voies
13	Le train de secours rejoint le train avarié
14	Liaison des trains
15	Le convoi rejoint le terminus
16	Dissociation du convoi
17	Le train de secours rejoint la station suivante

Tableau 3-1: Actions

C1	Réinitialisation immédiate du train possible
C1.1	Oui
C1.2	Non
C2	Puissance de traction suffisante
C2.1	Oui
C2.2	Non
C3	Présence d'une forte pente avant le terminus
C3.1	Non
C3.2	Oui
C4	Position du train avarié par rapport à la station
C4.1	A quai
C4.2	Sous tunnel
C4.3	Partiellement à quai
C5	Position du train de secours par rapport à la station
C5.1	A quai
C5.2	Sous tunnel
C5.3	Partiellement à quai
C6	Présence d'une station entre le train de secours et le train avarié
C6.1	Oui
C6.2	Non

Tableau 3-2: Eléments contextuels

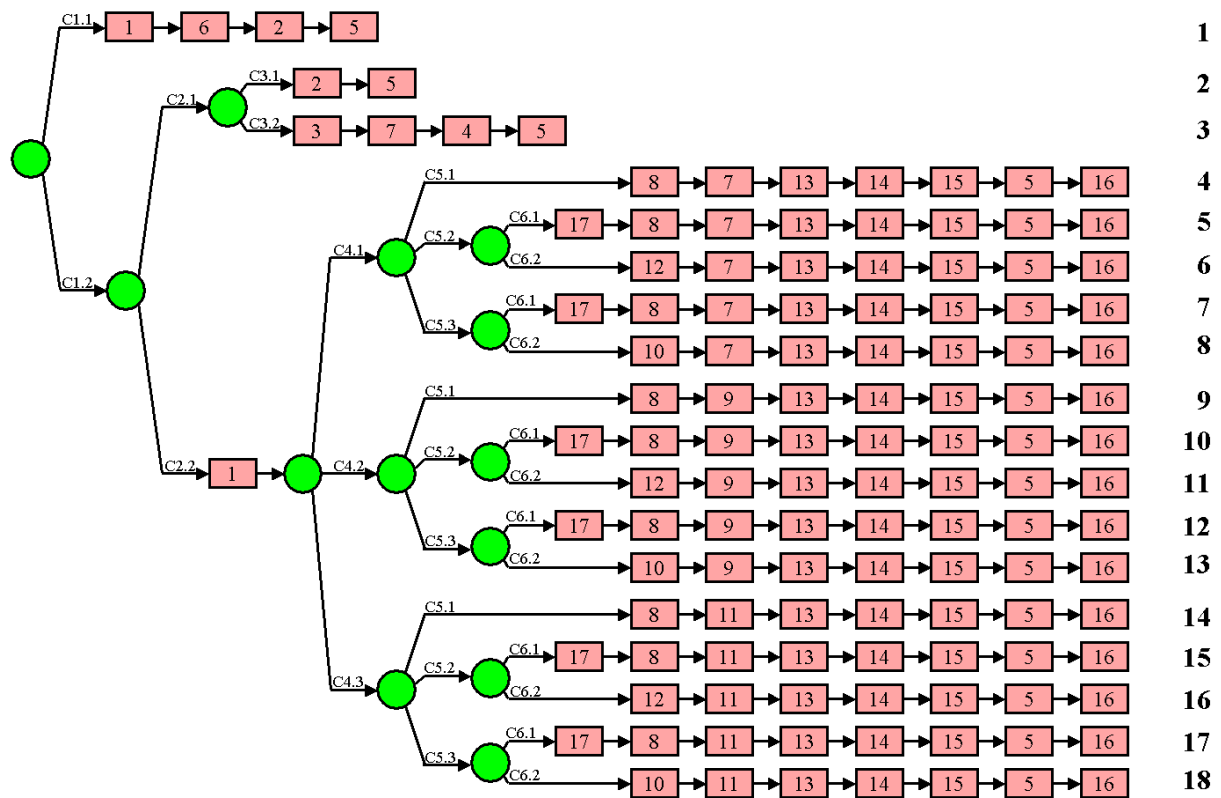


Figure 3-1 : Arbre de gestion d'un incident

La première question correspond à la possibilité de réinitialisation du train par le conducteur : si ce dernier peut réinitialiser son train, il le fait. Cette opération permet de reprendre la course normale (ligne 1 de l'arbre). Dans le cas contraire, un deuxième élément entre en considération : le nombre de motrices inactives. Si ce nombre le permet, la rame peut être ramenée en terminus de manière autonome, soit en continuant le service voyageur jusqu'au terminus (ligne 2 de l'arbre), soit en arrêtant le service voyageur en cours de route (notamment s'il y a une zone de forte déclivité⁹) et en repartant sans voyageur (ligne 3 de l'arbre). Enfin, si la rame ne peut pas être ramenée de manière autonome, un train opérationnel est utilisé pour pousser (exceptionnellement tirer) la rame avariée. Pour ce faire, les deux rames doivent être vides. L'évacuation des voyageurs de chaque rame est réalisée de manière différente en fonction de la position de celle-ci. Si la rame est à quai, il suffit de prier les voyageurs de bien vouloir descendre. Si elle est partiellement à quai, il faut ouvrir les portes d'intercommunication des voitures engagées sous tunnel et faire passer les voyageurs vers les voitures à quai. Enfin si la rame est entièrement sous tunnel, les voyageurs descendent sur les voies et sont dirigés vers la station la plus proche (après que le PCC ait

⁹ La déclivité correspond à la pente d'une voie. Quand la déclivité est importante, les rames ont besoin d'une puissance supérieure pour monter la pente.

réalisé une coupure de courant confirmée par du personnel à pied d'œuvre, bien sûr). Ces deux dernières procédures d'évacuation sont longues et, dans le cas de l'évacuation de la rame de secours, il est préférable, si une station est libre entre les deux rames, d'évacuer les voyageurs à cette station. Ainsi le CREG dispose de 3 manières d'évacuer la rame avariée et 5 cas pour l'évacuation de la rame de secours. Une fois les deux rames vides, la procédure est identique. Les 15 dernières lignes de l'arbre représentent la procédure de secours en fonction de la position des deux rames en jeu.

Cette représentation arborescente est une vue compilée d'une base de règles [Buchanan, 1984, pp. 23-24] qui contiendrait entre autres les règles suivantes :

Si C11	Faire 1, 6, 2 et 5
Si C12 & C21 & C31	Faire 2 et 5
Si C12 & C21 & C32	Faire 3, 7, 4 et 5
Si C12 & C22 & C41 & C51	Faire 1, 8, 7, 13, 14, 15, 5 et 16
Si C12 & C22 & C41 & C52 & C61	Faire 1, 17, 8, 7, 13, 14, 15, 5 et 16
Si C12 & C22 & C41 & C52 & C62	Faire 1, 12, 7, 13, 14, 15, 5 et 16
...	

Cette présentation est plus simple à maintenir que la base de règles correspondante, car les connaissances sont organisées et regroupées en fonction des prémisses. Le contexte sert alors de règle écran, comme le préconise Clancey [1979], pour séparer les règles en fonction de leur domaine d'application. Il est ainsi plus simple d'éviter les conflits et les redondances entre règles. Nous observons également une hiérarchisation des éléments de contexte intervenant dans la sélection des actions : allant des plus généraux vers les plus spécifiques. Nous rejoignons ainsi l'idée de Watson et Perera [1998]. Toutefois cette représentation est de taille comparable à la base de règles correspondante.

La représentation de la gestion de l'incident par l'arbre ci-dessus est sommaire par rapport à la réalité observable (et observée) au poste de contrôle. Dans certaines conditions les procédures sont adaptées et donnent lieu à une différenciation des stratégies accessibles à partir de l'arbre de la Figure 3-1. Pour tenir compte de ces spécialisations, il est nécessaire de multiplier les branches de l'arbre. Ainsi, dans certains cas, l'action 1, qui précède la divergence des 15 branches correspondants aux différentes conditions de l'application de la sous-procédure « secours » peut ne pas être réalisée. Suivant cette représentation arborescente (et son équivalent sous forme de règles), il faut rajouter une condition à la réalisation de cette action puis recopier la partie de l'arborescence qui la suit. Cette opération alourdit grandement l'arbre.

Dans notre domaine d'application de nombreux éléments de contexte d'un incident influencent grandement la gestion de celui-ci. Les arbres ainsi construits prennent rapidement

une ampleur considérable. Nous devons trouver une représentation moins sensible à l'explosion combinatoire. Pour cela, nous commençons par une analyse des spécificités des arbres que nous obtenons. Les caractéristiques de notre domaine d'application font que les arbres ainsi obtenus sont très particuliers :

1. Les branches représentent différents moyens de retourner à une situation normale d'exploitation, à partir de situations différentes.
2. On observe une accumulation des actions sur le bout des branches. Ceci est dû au fait que, face à un incident, les CREG cherchent d'abord à recueillir un maximum d'informations avant de choisir une stratégie. Quelques actions sont toutefois effectuées avant une connaissance totale de la situation (par exemple l'action 1 dans l'arbre présenté, qui correspond à une action de régulation des autres trains).
3. Des séquences d'actions identiques sont portées par différentes branches. Ceci révèle l'existence d'invariants de l'action, qui sont des petites procédures telles que « partir Haut Le Pied », « évacuer les voyageurs »... Celles-ci sont bien connues des CREG et partagées par l'ensemble de ceux-ci.
4. De nombreuses branches se terminent par les mêmes séries d'actions. Par exemple, lorsqu'une rame a besoin de secours, la stratégie consiste à évacuer la rame avariée et la rame de secours (le mode d'évacuation dépend de la position des différentes rames concernées, on a alors plusieurs branches dans l'arbre), puis à faire un convoi (celui-ci est réalisé toujours de la même façon, quelle que soit la position des rames). En réalité, l'exécution des actions précédentes a permis de se ramener à une situation commune. La même série d'actions permet alors de retrouver une situation d'exploitation normale.
5. Dans l'arbre de la Figure 3-1, nous n'avons pas montré les inversions possibles d'actions, notamment celles qui décrivent l'évacuation des voyageurs des deux rames concernées par la procédure de secours, le cas échéant. Ce modèle d'arbre ne peut pas représenter ceci en tant que tel et la seule solution dans ce modèle est la multiplication des branches concernées.

Dans un premier temps, nous avons allégé l'arbre, sans modifier sa structure, en remplaçant les séquences d'actions identifiables comme « petites procédures » par les CREG par un seul objet représentant la dite procédure, appelé macro-action (voir Tableau 3-3). Les macro-actions sont basées sur l'expérience des opérateurs et reprennent une part du plan de diagnostic et d'action [Pomerol, 1997]. Cette étape prend en compte le troisième point sus-cité. Il nous permet, de plus, de mieux appréhender les raisonnements sous-jacents.

Macro-actions		Séquence d'actions
MA 1	Reprise du service voyageur par le train avarié	Actions 2 et 5
MA 2	Arrêt du service voyageur par le train avarié	Actions 7, 4 et 5
MA 3	Faire un convoi	Actions 13, 14, 15, 5 et 16

Tableau 3-3: Macro-actions et leur équivalent en terme de séquence d'actions

Les points 1, 2 et 4 montrent que le raisonnement employé par les CREG est plutôt guidé par la situation et l'évolution du contexte. En raisonnant sur les situations et la possibilité de se ramener à une situation commune à partir de situations différentes, il semble logique de baser la représentation sur cet élément clé. Nous avons donc décidé de rassembler les branches dès lors qu'elles partagent la même série terminale d'actions. C'est le point clé de notre changement de représentation, car nous passons d'une représentation basée sur le contexte de l'incident à une représentation basées sur le *contexte de résolution de l'incident*. La première est statique alors que la seconde est hautement dynamique. De plus nous obtenons un graphe beaucoup plus compact que l'arbre correspondant (voir Figure 3-2).

La dernière étape correspond à la prise en compte des séquences d'actions et de décision pouvant être réalisées en parallèle. Nous introduisons un branchement temporel représentant ces indépendances temporelles. Par défaut les actions et décisions se suivent sans chevauchement, le branchement temporel indique que les parties de graphe portées par ses branches doivent être réalisées après celles qui précèdent le branchement, avant le début de celles qui le suivent et qu'entre les deux les différentes parties peuvent être menés de front.

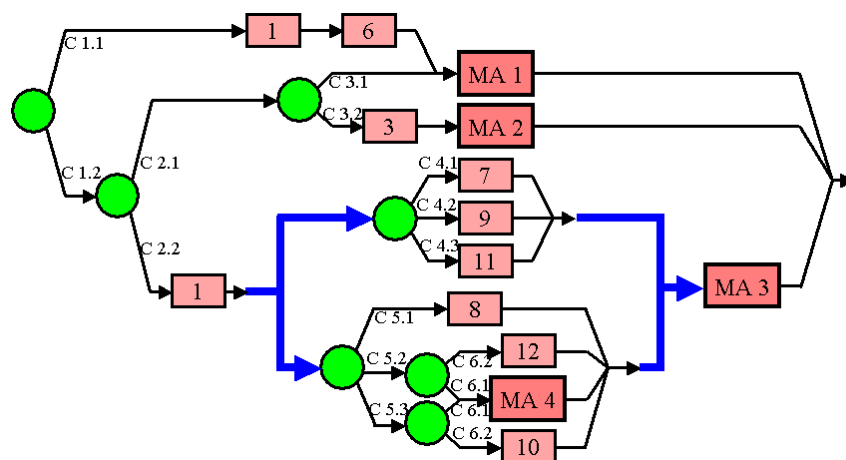


Figure 3-2: Graphe de gestion d'un incident

Le graphe que nous obtenons ici est la dernière étape avant la définition du modèle de *graphe contextuel*. Il n'en est pas un car il manque une des principales caractéristiques des graphes contextuels, à savoir l'organisation en composants, qui n'est qu'esquissée ici avec la notion de macro-actions. Toutefois la structure même de ce graphe est le point de départ de celle des graphes contextuels que nous présentons maintenant.

Section 2 :

Éléments constitutifs

Les graphes contextuels sont issus de ceux qui viennent d'être présentés (Figure 3–2) : la structure est identique, les principes sous-jacents aussi, mais une différence majeure est introduite. Les graphes contextuels sont conçus pour évoluer au cours de leur utilisation. Les macro-actions ne pouvant être composées que de simples séquences linéaires d'actions, incompatibles avec l'évolution possible de leur structure, nous les remplaçons par une nouvelle structure d'activité acceptant une évolution future de sa structure, notamment l'ajout de nœuds contextuels et de solutions alternatives. Cette section présente les différents types d'éléments constitutifs des graphes contextuels. Ceux-ci sont composés de cinq principaux éléments : les actions élémentaires, les activités, les nœuds contextuels, les nœuds de recombinaison et les branchements temporels.

2.1 Actions élémentaires

Les actions élémentaires des graphes contextuels correspondent aux plus simples actions pouvant être identifiées dans le domaine concerné. Sur ce point elle sont similaires aux tâches génériques de Chandrasekaran [1986]. C'est à partir de ces tâches de base que nous décrivons les tâches plus complexes en les organisant dans un graphe contextuel. Elles sont ainsi la brique élémentaire de l'édifice.

Toute action peut toujours être détaillée. Aussi nous considérons comme élémentaire les actions qui le sont pour l'utilisateur final du système. Dans notre application, par exemple, nous prenons comme action élémentaire les actions réalisées par les différents acteurs (qu'ils soient au poste de contrôle ou sur le terrain), dont la réalisation peut être réduite à une simple explication (en une phrase courte) de la marche à suivre, en prenant le point de vue de l'opérateur du poste de contrôle. Par exemple l'action *réinitialiser la rame* est élémentaire du point de vue de ce dernier, mais est en réalité une action pouvant être détaillée pour le conducteur de la rame qui doit réaliser cette réinitialisation. Seul le résultat de cette activité du conducteur importe pour le chef de régulation ; ce dernier n'établit aucune stratégie pour réaliser cette action. C'est donc une action élémentaire de son point de vue, bien qu'elle corresponde à une activité complexe pour le conducteur.

Une action élémentaire dans notre domaine est, par exemple, de faire retirer ou de faire remettre une barrette de rupteur d'alarme par un acteur sur le terrain. Ces actions sont élémentaires car leur décomposition n'apporte pas d'élément supplémentaire quant à leur mise en œuvre.

Les éléments du graphe représentant ces actions élémentaires ont un unique précédent et un unique suivant. Ces éléments font référence à un descripteur d'action composé du nom de l'action, d'un but, de pré-conditions devant être vérifiées pour toute réalisation de l'action (dont, éventuellement, des disponibilités de ressources), d'une description simple de l'action et de conséquences sur la situation (post-conditions). Dans nos figures, les actions élémentaires sont représentées par des rectangles rouges.

2.2 Activités¹⁰

Les tâches plus complexes requièrent une structuration de l'activité qui leur est associée. Nous introduisons également des objets « activité » dans notre modèle pour leur faire référence. Ces activités sont similaires aux actions élémentaires si ce n'est que leur réalisation est détaillée et induit une part de planification. Le détail consiste à présenter les étapes intermédiaires, les actions et activités permettant d'y arriver et les éléments permettant de choisir, le cas échéant, entre diverses stratégies. Le modèle utilisé pour la représentation des détails de ces activités seront discutés lors de notre proposition d'un modèle à base de graphe contextuels dans la section 4 et surtout dans le cinquième chapitre. Pour les propos tenus dans cette section, il est suffisant de retenir que les objets « activité » sont similaires à des objets « action », mais qu'ils font référence à une structure plus complexe.

Les macro-actions précédemment introduites sont des cas particuliers d'activités qui se réduisent à une simple séquence d'actions. L'évolution potentielle de ces plans d'actions peut modifier la structure et ajouter des embranchements divers. Ces structures ne rentreraient alors plus dans le cadre très strict des macro-actions. Il est donc logique de les considérer dès le départ comme des activités plus complexes, bien qu'elles soient encore simples.

Une activité dans notre domaine d'application est, par exemple, la déconsignation. Cette activité consiste à rendre la remise sous tension possible en remplaçant un rupteur d'alarme précédemment actionné. Elle fait appel à l'action « faire remettre une barrette de rupteur d'alarme ». Si cette action est concluante, la sous-section concernée est déconsignée, dans le cas contraire la personne à pied d'œuvre doit retirer la barrette, enlever les débris éventuels et replacer la barrette. Si ceci ne donne toujours pas le résultat escompté, une procédure de vérification du circuit des rupteurs d'alarme sur la section entière est nécessaire. Cet exemple

¹⁰ Le terme « activité » s'entend ici dans son sens le plus général. Nous l'avons introduit dans la définition du modèle au même titre que le terme « action » précédent : dans ce modèle, ces termes correspondent à des composants particuliers des graphes, dont le nom est lié à la sémantique que nous avons appliquée au modèle (voir le cinquième chapitre).

montre la différence entre l'action *remettre une barrette de rupteur d'alarme* et l'activité *déconsignation* : cette dernière repose sur un plan dynamique d'actions en fonction de la situation et de son évolution.

Les éléments du graphe représentant ces activités ont un unique précédent et un unique suivant. Ces éléments font référence à un descripteur d'activité composé, comme les descripteurs d'actions, d'un nom, d'un but, de pré-conditions et de conséquences sur la situation. En plus des descripteurs d'action, les descripteurs d'activité contiennent les différents plans possibles permettant de mener à bien cette activité. Dans nos figures, les activités sont représentées par des rectangles oranges et sont nommées.

2.3 Nœuds contextuels

Le but des graphes contextuels est de représenter l'ensemble des stratégies connues pour atteindre un but déterminé. L'existence de plusieurs solutions possibles implique une représentation des décisions établissant le choix de la stratégie qui sera appliquée. Ces choix sont réalisés en analysant les valeurs de différents éléments de la situation. Ces éléments sont soit directement liés à la tâche en cours, soit appartiennent au contexte. Ces derniers étant majoritaires, nous rassemblons, par extension, tous ces éléments de choix sous le terme de « nœud contextuel ».

Les nœuds contextuels correspondent à des éléments représentant les choix. Le but de notre modèle est de représenter des raisonnements hautement liés au contexte et d'accumuler un nombre important de nœuds de décision relatifs aux éléments contextuels. Nous appelons donc nœud contextuel tout nœud représentant une décision, qu'elle soit appuyée sur un élément contextuel ou, plus rarement, sur un élément directement lié à la tâche. Lors de l'analyse d'un élément de contexte consistant par rapport au choix de la méthode employée pour la réalisation de la tâche en cours, plusieurs valeurs sont possibles. Pour chaque valeur possible la stratégie peut être différente. Nous observons ainsi une divergence des stratégies lors de nœuds contextuels.

Cette divergence a pour conséquence qu'un nœud contextuel a un unique précédent et plusieurs suivants (autant que de valeurs actuellement connues pour l'élément du contexte analysé). Les éléments du graphe correspondant à ces divergences sont représentés par des cercles verts. Nous soulignons ici le caractère non-définitif de la liste des valeurs admises pour chaque élément analysé. Pour certains cette liste est connue à l'avance (par exemple quand on considère si le jour est férié ou non), pour d'autres cette liste est incomplète et ne peut pas être fixée *a priori* (par exemple pour la cause de l'incident ou pour les raisons d'un

appel d'un conducteur). Ceci joue un rôle important lors de l'implémentation du système, comme nous le verrons dans le chapitre 6.

2.4 Nœuds de recombinaison

Le but de chaque opérateur est de revenir à un état normal du système (élimination des causes de l'incident et retour à un trafic nominal). Pour cela, il réalise des actions et des activités afin de modifier l'état du système. De plus certains éléments du contexte peuvent varier dans le laps de temps concerné. Nous observons donc une évolution de la situation en cours de traitement. La politique des opérateurs étant, de plus, de se ramener le plus rapidement à des situations communément rencontrées, nous observons une tendance à la réduction au plus tôt des différences entre les stratégies. Cette politique pousse les opérateurs à prendre des mesures permettant de se ramener à une situation commune à partir de situations initialement différentes. La réalisation de ses actions réduit l'écart entre les stratégies, si bien que, plus ou moins rapidement, les stratégies sont similaires. L'influence, sur les décisions, des éléments contextuels analysés lors de la séparation des stratégies est alors rendue obsolète. Les nœuds de recombinaison permettent ainsi de représenter cette convergence de stratégie guidée par la politique globale des opérateurs.

Les éléments du graphe représentant ces recombinaisons de stratégies sont représentés par une simple convergence d'arcs. Ces éléments du graphe ont plusieurs précédents et un unique suivant.

2.5 Branchements temporels

Les branchements temporels représentent des groupes de décisions/actions pouvant être menés en parallèle. Ces éléments se présentent comme nécessaires dans les représentations modernes et nous la retrouvons (entre autres) dans UML [OMG 1999]. Dans nos figures, chaque branchement temporel est représenté par une paire de crochets épais bleus. Ils représentent les limites temporelles des groupes temporellement indépendants. Le premier élément de la paire a un unique précédent et plusieurs suivants ; le deuxième élément de la paire a exactement ce même nombre de précédents et un seul suivant.

Les branchements temporels permettent d'enrichir grandement le pouvoir de représentation des graphes contextuels et augmentent la généralité de ceux-ci. Par exemple, dans notre fil conducteur, les actions concernant l'évacuation des rames en détresse et de secours sont indépendantes temporellement : elles peuvent être réalisées dans des ordres indifférents. Les structures de décision associées au mode d'évacuation sont également indépendants. Sans la représentation de l'indépendance temporelle, il aurait été nécessaire de procéduraliser un

élément du contexte pour faire apparaître une branche montrant un ordonnancement différent des actions réalisées. Pourtant il n'existe pas un unique élément de contexte responsable d'un tel ordonnancement. Il est hors de question de distinguer tous les cas possibles, ne serait-ce que par la faiblesse des liens entre ces éléments de contexte et les choix réalisés. Il est préférable, puisque l'ordonnancement de ces actions importe peu, de laisser le choix à l'opérateur d'organiser lui-même celui-ci en fonction d'éléments de contextes divers et variés.

Section 3 :

Graphes contextuels

3.1 Structure des Graphes Contextuels

Un *graphe contextuel* est un graphe *orienté*, *acyclique* ayant une *unique source* et un *unique puits*. Les sommets du graphe sont des instances des éléments décrits ci-dessus. Ils sont reliés par des arcs montrant la *précédence temporelle*. Les différents chemins possibles pour passer de la source au puits représentent les différentes stratégies possibles pour la réalisation d'un but prédéfini, les nœuds contextuels identifient le contexte dans lequel ces stratégies sont applicables. La Figure 3-3 montre un graphe contextuel possible représentant l'exemple que nous suivons dans ce chapitre (ce graphe ne présente pas encore d'activités, pour en discuter en détail dans la section 5).

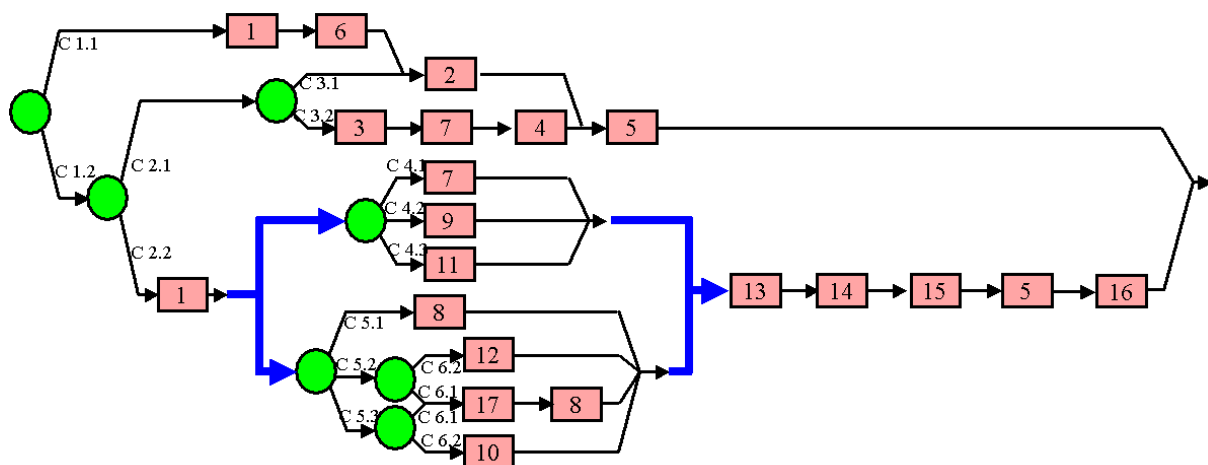


Figure 3-3: Graphe contextuel

Sous cette forme, un graphe contextuel se comporte comme l'arbre précédent. Le choix d'une stratégie est réalisé de la manière suivante. Entrant par la source du graphe, on suit les arcs en répertoriant les actions rencontrées. Lors de la divergence d'un nœud contextuel, on suit la branche appropriée à la situation courante. Lors d'un nœud de recombinaison, on

poursuit simplement le chemin. Enfin, lors d'un branchement temporel, on suit toutes les branches en parallèle. Le parcours s'arrête quand on arrive à l'unique puits. La séquence d'actions ainsi constituée correspond à une stratégie possible pour atteindre le but souhaité dans le contexte courant. Le fait que le graphe soit acyclique garantit la finitude de cet algorithme de parcours.

3.2 Analyse de la structure

Cette structure permet de représenter à la fois le diagnostic sur la situation (grâce aux nœuds contextuels) et les actions réalisées dans les différents cas. De cette manière, les graphes contextuels permettent de représenter différentes stratégies possibles pour atteindre un but souhaité en fonction du contexte et de les organiser dans une structure de diagnostic/action semblable à celle utilisée pour les procédures de gestion de systèmes dynamiques complexes.

L'orientation des arcs montre la précédence temporelle, ce qui structure fortement l'ordonnement des actions. Ainsi les graphes contextuels sont des représentations très structurées du diagnostic et des actions. La réalisation d'actions et d'activités modifient la situation courante. A chaque point du graphe, nous savons quels éléments de contexte sont actifs quant aux décisions qui sont prises et qu'elles actions ont été réalisées. Nous pouvons ainsi définir quelle situation correspond à cette étape. La description de cette situation n'est bien sûr pas complète, mais elle est opérationnelle : tous les éléments actuellement nécessaires à l'établissement de la stratégie y sont présents. Les graphes contextuels permettent donc de représenter également la situation et son évolution au cours du traitement avec la finesse nécessaire aux choix stratégiques actuels.

L'orientation des arcs représente l'ordonnement temporel des actions et des activités. Comme aucune action ou activité ne peut se précéder elle-même¹¹, il est logique que le graphe soit acyclique. Le fait que les graphes soient orientés et acycliques garantit la terminaison des algorithmes de recherche de chemins. Ces algorithmes sont à la base de l'utilisation de ces graphes, car chaque chemin correspond à une stratégie différente.

¹¹ Quand une action ou une activité doit être répétée, nous faisons face à deux applications différentes d'un même descripteur d'action ou d'activité (celles-ci diffèrent, ne serait-ce que par leur heure d'exécution). Pour représenter ceci dans un graphe contextuel, nous utilisons deux instances différentes faisant appel à un même descripteur d'action ou d'activité.

Un graphe contextuel a une structure en fuseaux : plusieurs chemins, partant d'une unique racine et allant à un unique puits, se séparent et se rejoignent. Plusieurs fuseaux peuvent se suivre, montrant ainsi une réalisation en plusieurs étapes où tous les chemins se rejoignent en un seul sur une partie intermédiaire avant de se séparer de nouveau. L'unique racine se justifie par le fait que la structure représente à la fois le diagnostic et l'action. Tant que le diagnostic n'est pas réalisé, les éléments permettant de choisir entre les différents chemins ne sont pas encore mis en avant : la situation n'est pas encore détaillée. Inversement, quand toutes les actions sont terminées, nous avons atteint le but fixé. La situation est donc identique (au niveau de détail mis en oeuvre) à la situation finale souhaitée. Ceci justifie le point de sortie unique de la structure.

Un graphe contextuel représente ainsi un raisonnement contextuel local construisant une stratégie pour la réalisation d'un but en fonction de la situation et de son évolution. Le raisonnement est local, car les connaissances contextuelles intervenant dans les décisions relatives à la réalisation du but souhaité ne dépendent que de ce but et de l'évolution de la situation relativement à la réalisation de ce but. La situation peut, en effet, évoluer indépendamment des actions entreprises pour la réalisation de ce but, mais seules les conséquences de ces évolutions sur la réalisation du but et sur les moyens mis en oeuvre pour celle-ci interviennent dans les décisions considérées. Le contexte ainsi mis à contribution se limite aux connaissances contextuelles impliquant une adaptation des actions.

Les graphes contextuels présentent également des sous-structures en fuseau (ayant une racine et un puits uniques). Ces sous-structures représentent ainsi divers chemins d'action permettant de passer d'une situation à une autre en impliquant uniquement des connaissances contextuelles locales. Quand la situation intermédiaire atteinte par ces sous-structures peut être identifiée à un but précis, celles-ci sont susceptibles de représenter en réalité une activité secondaire. Nous reviendrons sur ce point dans la section 4 de ce chapitre.

3.3 Algorithme d'acquisition incrémentale de connaissances¹²

L'un des principaux objectifs de la définition du modèle des graphes contextuels est la possibilité de prise en compte de l'évolution des connaissances de niveau stratégique lors de la définition d'un système à base de connaissances. L'originalité des graphes contextuels réside dans leur capacité d'évolution et d'assimilation des pratiques réelles des opérateurs afin

¹² Nous n'entrons pas ici dans une discussion entre l'acquisition de connaissances et l'apprentissage. Nous utilisons dans cette section les deux termes de manière interchangeable. Nous discutons plus précisément les liens et les différences entre ces deux notions dans le cinquième chapitre, section 3.3, page 155.

d'augmenter leurs capacités représentatives. L'acquisition incrémentale de connaissances est donc un point capital du modèle.

Nous proposons ici de décrire un algorithme simple permettant cette acquisition, grâce à l'interactivité entre le système et l'opérateur. Cette étape est nécessaire quand une nouvelle pratique est appliquée par l'opérateur pour la gestion d'un état particulier du système dans un contexte donné. Nous suivons pas à pas la description de l'algorithme sur un exemple d'acquisition d'une nouvelle pratique dans notre fil conducteur (dont nous rappelons dans la Figure 3-4 le graphe contextuel initial).

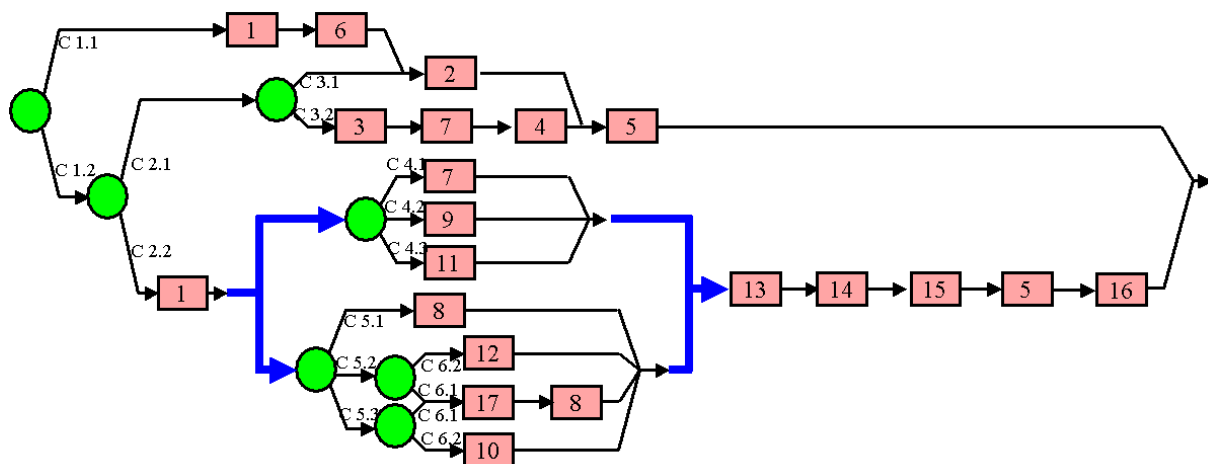


Figure 3-4: Graphe contextuel avant acquisition de la nouvelle pratique

Le système ayant pris connaissance de la séquence d'actions réalisée, une première phase d'interaction avec l'opérateur permet de déterminer quelle séquence d'actions aurait dû être appliquée. Cette opération est réalisée par la sélection des valeurs prises par les éléments contextuels déjà identifiés dans le graphe contextuel, ce qui correspond à la recherche du chemin (d'actions) suivi. Ceci est réalisé pas à pas en parcourant le graphe de la source jusqu'au puits. A chaque nœud contextuel rencontré, le système demande à l'opérateur quelle valeur correspond à la situation courante. L'algorithme continue alors en suivant la branche correspondante et se poursuit jusqu'au puits.

Lors du parcours du graphe contextuel à la recherche du chemin d'action attendu, un nœud contextuel peut ne pas proposer la valeur pour la situation courante. L'utilisateur ajoute alors cette valeur et une nouvelle branche au nœud contextuel. Cette nouvelle branche est reliée à un nœud de recombinaison placé à la fin du graphe contextuel, afin de ne pas perdre la structure en fuseau du graphe contextuel. La recherche du chemin d'actions est interrompue.

Les branches non suivies ne sont pas concernées par la suite de l'algorithme (et sont représentées en estompé dans la Figure 3-5).

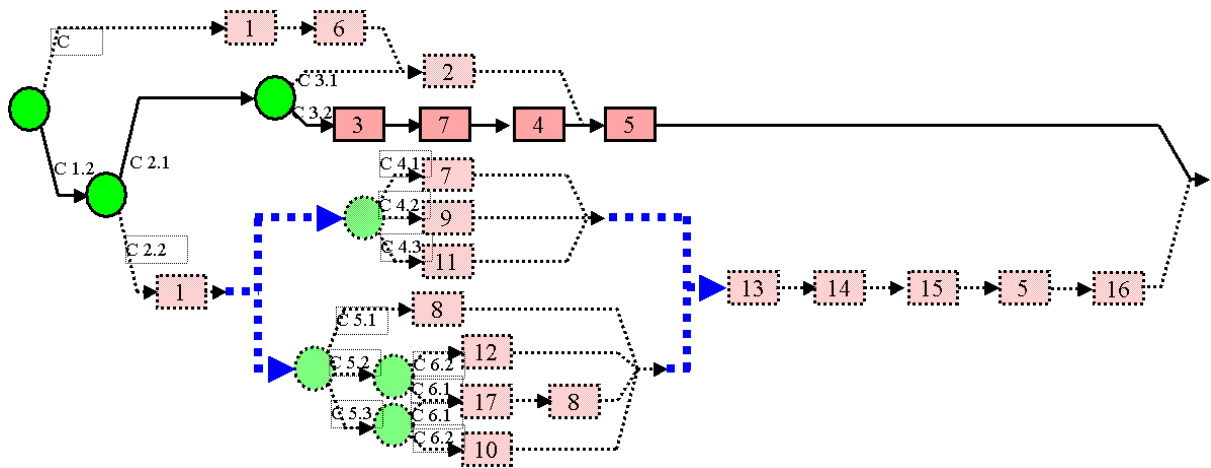


Figure 3-5: Sélection de la stratégie ayant dû être appliquée

Le chemin ainsi sélectionné correspond à la stratégie attendue pour gérer cet état du système dans la situation courante. L'algorithme compare ensuite la stratégie attendue à la séquence d'actions effectivement réalisées (la pratique), à la recherche de différences stratégiques. Tant que la séquence d'action et d'activité correspond à celles prévues par la stratégie attendue, celles-ci sont simplement marquées comme communes. Tant que la pratique correspond à la stratégie, le graphe n'est pas modifié (il n'est pas utile d'apprendre ce que l'on sait déjà).

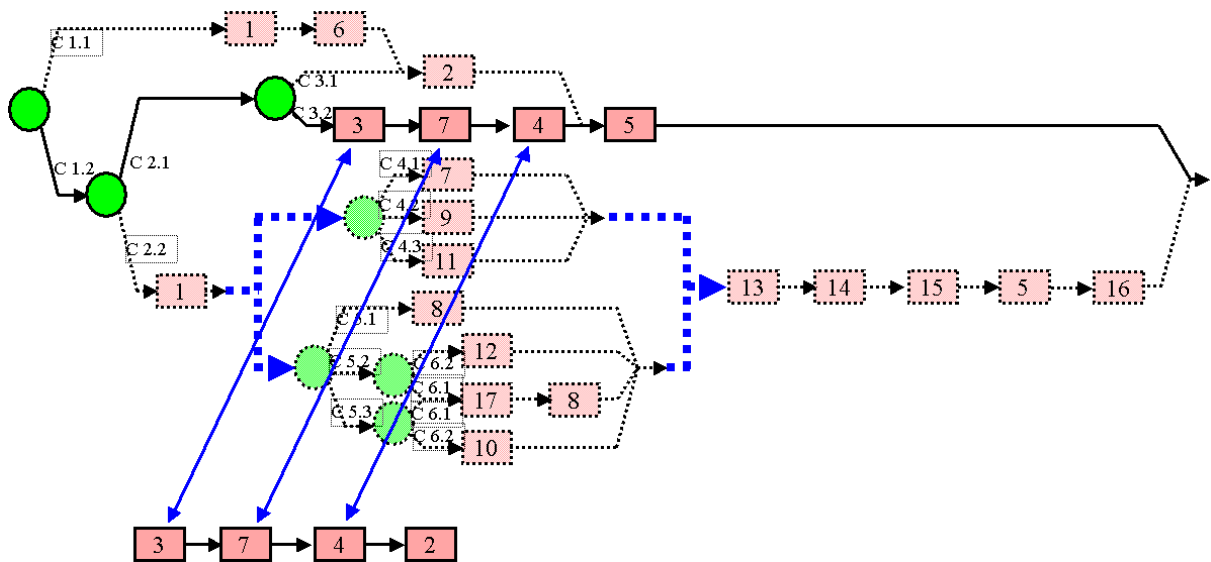


Figure 3-6: Appariement des actions attendues et des actions réalisées

Quand des actions ont été réalisées en pratique sans être prévues par la stratégie attendue (ou non réalisées en étant prévues), l'algorithme demande à l'opérateur d'expliquer son choix et d'explicitier les éléments de la situation ayant supporté cette décision. Cette deuxième phase d'interaction est, en réalité, le point principal dans l'acquisition de l'expertise de l'opérateur.

Ce dernier doit faire un effort d'introspection (au demeurant raisonnable) pour identifier les éléments de contexte en cause, qui l'ont amené à développer une pratique différente de celles connues par le système. Ces éléments de contexte sont alors insérés dans le graphe par l'ajout d'un nouveau nœud contextuel. La première branche de celui-ci correspond à la stratégie prévue (le chemin précédent) et la nouvelle branche correspond à la pratique.

Il peut arriver que la divergence de la stratégie et de la pratique se passe dans une zone du graphe correspondant à un nœud contextuel. Dans ce cas l'algorithme demande à l'utilisateur si l'élément contextuel correspondant est le support de sa décision. Si tel est le cas, c'est qu'une nouvelle valeur est enregistrée pour l'élément en question. Dans ce cas, aucun nœud contextuel n'est ajouté, seulement une nouvelle branche pour le nœud correspondant qui continue avec la suite de la séquence d'actions et d'activités de la pratique.

Quand la divergence de la pratique et de la stratégie est prise en considération par le graphe contextuel, l'algorithme recherche le point de rebranchement de la pratique et de la stratégie. Cette recherche est réalisée en parcourant en sens inverse la pratique et la stratégie tant que les actions et les activités correspondent. Dès qu'une divergence est repérée, l'algorithme insert un nœud de recombinaison.

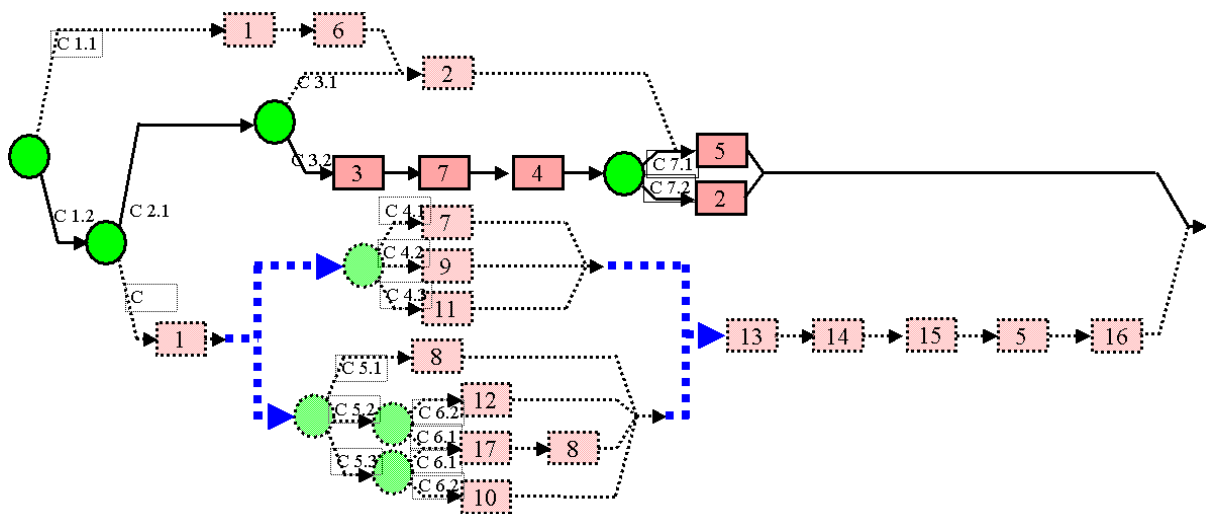


Figure 3-7: Graphe contextuel complet après acquisition de la nouvelle pratique

Quand la stratégie attendue passe par un branchement temporel, la recherche de similarité avec la pratique doit suivre en parallèle toutes les branches temporellement indépendantes. Pour chaque branche l'algorithme suit pas à pas son chemin comme à l'accoutumé. Si l'algorithme détecte, une divergence au sein du branchement temporel (une action est réalisée alors qu'elle ne correspond à aucune des actions attendues), alors celui-ci doit déterminer quelle est la nature de cette divergence. Deux cas sont en effet possibles : soit la divergence doit être traitée localement (elle correspond à une modification d'une des structures de

diagnostic/action portée par l'une des branches), soit elle correspond à une rupture plus générale de la pratique par rapport à la stratégie attendue. Pour établir la nature de la divergence, des interactions sont nécessaires avec l'utilisateur. Dans le premier cas, il doit déterminer quelle partie de la pratique correspond à quelle branche, afin de modifier la (ou les) structures de diagnostic/action concernées. Dans le second cas, la rupture implique une divergence sur l'ensemble des branches et sur une partie de la stratégie suivant le branchement temporel. Dans ce cas, l'utilisateur est requis pour déterminer quelles séquences d'actions correspondent aux différentes parties du graphe concernées.

Cet algorithme simple permet d'ajouter facilement des pratiques à un ensemble de graphes contextuels. Le point crucial à retenir pour l'acquisition de l'expertise des opérateurs est l'identification des éléments de contexte ayant induit les adaptations proposées par la nouvelle pratique. Ces informations permettent de positionner la nouvelle pratique par rapport aux stratégies déjà enregistrées dans le graphe. Les explications fournies par l'utilisateur à cet instant sont conservées pour expliquer les futurs choix stratégiques opérés par le système d'aide.

3.4 Autres algorithmes associés

Après une session d'acquisition d'une nouvelle pratique il est important de vérifier la validité des graphes modifiés. L'ajout d'une nouvelle pratique introduit des nouveaux nœuds contextuels et de recombinaison. L'introduction de ces nœuds doit respecter quelques règles pour éviter de rendre le graphe cyclique. Ces vérifications peuvent être réalisées au sein même de l'algorithme d'acquisition incrémentale de connaissances stratégiques. Il suffit notamment de ne pas insérer de nœud de recombinaison avant le nœud contextuel qui lui correspond. Cette opération est simple à mettre en œuvre : il suffit de parcourir le graphe du point d'insertion désiré du nœud de recombinaison et de suivre tous les chemins qui en partent et de vérifier que l'on ne passe pas par l'élément contextuel correspondant. Si un bouclage est repéré, il doit être éliminé. Normalement un tel bouclage est impossible, car la divergence entre la pratique et la stratégie a lieu lors de la première différence entre les séquences d'actions/activités et la recombinaison lors de la dernière.

Un autre algorithme utile recherche les sous-graphes temporellement indépendants. Cet algorithme repère les sous-graphes qui sont répétés sur différents chemins du graphe et qui présentent une inversion les uns par rapport aux autres. Si de tels cas se présentent (ils sont extrêmement rares), l'algorithme demande à l'utilisateur si cette inversion est strictement réglementée par des contraintes techniques ou si ces sous-structures de diagnostic/action sont temporellement indépendantes. Dans ce dernier cas, l'algorithme introduit un branchement temporel dans le graphe, ce qui augmente la généralité du graphe contextuel ainsi obtenu.

3.5 Analyse de la complexité

Les graphes contextuels sont une forme très compacte des arbres construit précédemment. Ainsi le graphe de la Figure 3–3 contient plus de stratégies (grâce à la combinatoire liée à l’ordonnement libre des actions portées par le branchement temporel) que l’arbre de la Figure 3–1. Le graphe est composé de 22 actions, 7 nœuds contextuels, 5 nœuds de recombinaison et un branchement temporel, soit 35 éléments. L’arbre correspondant est composé de 122 actions et 13 nœuds de décision, soit 135 éléments ; sans compter que pour représenter tous les ordonnancements possibles des actions portées par le branchement temporel, il est nécessaire de tripler la partie de l’arbre correspondant à la procédure « porter secours » soit un surcoût de 222 actions et 20 nœuds de décision (plus encore ceux représentant les supports de choix entre les trois ordonnancement possibles). Il est évident que plus le nombre de détails contextuels entre en ligne de compte plus l’avantage des graphes sur les arbres de décision se fait sentir.

La complexité en mémoire des graphes contextuels est très faible. L’ajout de la considération d’un élément de contexte supplémentaire ajoute un certain nombre d’éléments mais en aucun cas n’oblige à copier un pan de la structure.

Les calculs effectués sur ces structures sont soit des parcours de chemins (linéaires en la profondeur du graphe), soit des recherches sur l’ensemble des chemins (proportionnels à $n*m$, où n et m sont respectivement la largeur et la profondeur du graphe).

3.6 Graphes Contextuels et dynamisme du contexte

Nous avons souligné, dans le premier chapitre, le rôle du contexte et le dynamisme des connaissances contextuelles au cours d’un processus de décision/action tel que celui que nous considérons. Les connaissances contextuelles ne sont pas toutes utiles à chaque instant du processus. Le modèle des Graphes Contextuels permet de mettre en valeur cette caractéristique.

La Figure 3–8 explique le phénomène de procéduralisation/déprocéduralisation et sa représentation grâce aux graphes contextuels. Cette figure est un extrait du graphe contextuel précédent (c’est la partie du graphe portée par la première branche du branchement temporel). Elle montre comment l’élément de contexte *position de la rame avariée* (C4) influence sur son évacuation. Tant que l’évacuation de la rame avariée n’est pas d’actualité, cet élément de contexte n’est d’aucune utilité. Par contre dès que cette rame doit être évacuée, alors cet élément de contexte devient primordial. A ce moment la connaissance contextuelle correspondant à la situation courante (C4.1, C4.2 ou C4.3) est procéduralisée (voir premier

chapitre, section 5.1.2, page 53). Quand l'évacuation est terminée, l'élément de contexte est de nouveau inutile et la connaissance contextuelle précédemment procéduralisée est déprocéduralisée.

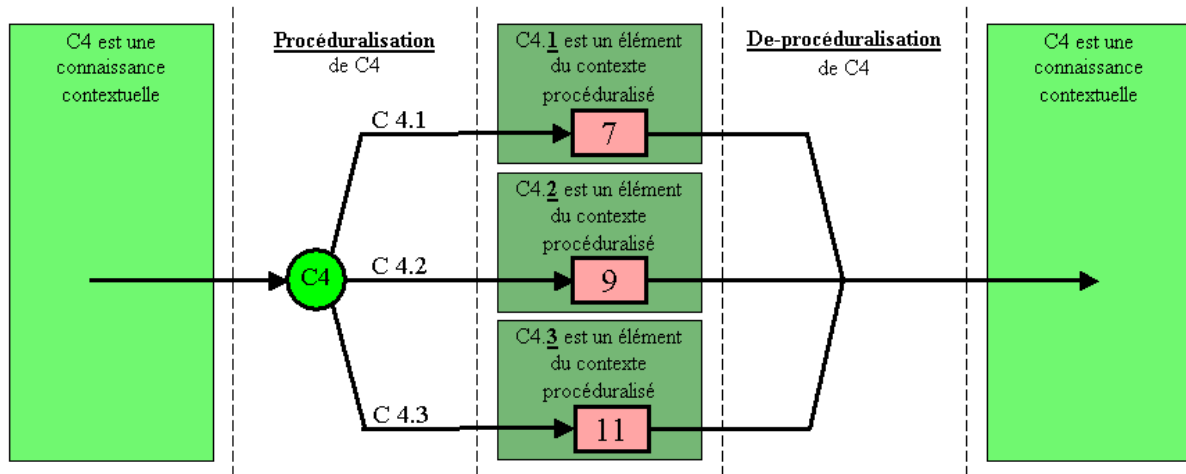


Figure 3-8: Mécanisme de procéduralisation/déprocéduralisation du contexte

Les graphes contextuels permettent ainsi de savoir immédiatement quelles sont les connaissances contextuelles procéduralisées à chaque instant. Il s'agit en fait des éléments de contexte qui ont été analysés pour sélectionner la branche courante et qui ne sont pas encore sortis de l'attention de l'opérateur.

Les graphes contextuels sont un modèle des connaissances de niveau stratégique dont l'organisation est basée sur l'évolution de l'état de procéduralisation des connaissances contextuelles en jeu. Ils sont intimement liés au contexte de résolution de l'incident dont ils suivent l'évolution.

Section 4 :

Graphes contextuels et activités¹³

4.1 Principes

La construction des graphes contextuels correspondants aux procédures de gestion de différents incidents fait ressortir des similitudes entre les graphes contextuels obtenus. Nous observons notamment qu'un grand nombre de sous-structures en fuseau ou en série de fuseaux (voir la section 3.2 de ce chapitre) des graphes contextuels sont communes aux différents graphes. Dans le graphe de la Figure 3-3, nous pouvons notamment identifier les sous-graphes contextuels présentés ci-dessous (Figure 3-9).

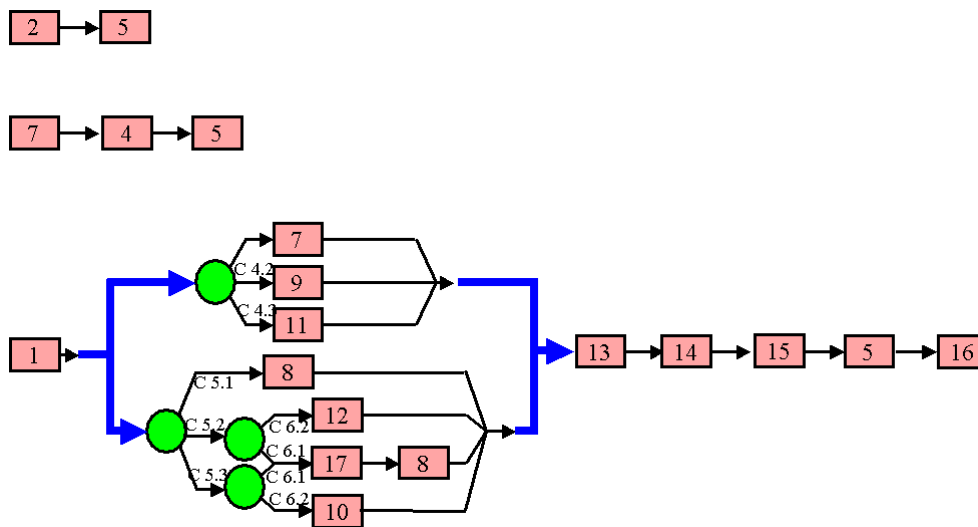


Figure 3-9: Sous-graphes contextuels

L'étude de ces sous-graphes contextuels avec les opérateurs révèle qu'en réalité de nombreuses procédures et pratiques sont communes à la gestion de nombreux incidents et qu'elles sont, qui plus est, explicables hors du contexte d'un graphe contextuel plus général. Ces structures sont identifiables par les opérateurs comme de petites procédures, qu'ils sont capables de nommer, analyser, expliquer et dont ils peuvent détailler les stratégies en fonction de différents contextes, de manière autonome, sans avoir recours au contexte de la gestion des incidents. Ces structures représentent des raisonnements contextuels locaux organisant des procédures et pratiques communes aux différents graphes construits.

¹³ Nous rappelons ici que le terme « activité » est à prendre dans son sens le plus général (voir note n°10, page 81).

Les structures ainsi identifiées respectent les règles de construction des graphes contextuels, ont une entrée et une sortie, et représentent des raisonnements locaux indépendants du graphe contextuel dont elles font partie. Elles ont ainsi toutes les qualités nécessaires pour être considérées comme des graphes contextuels à part entière, représentant un raisonnement contextuel local offrant une structure de diagnostic/action relative à la réalisation d'une sous-tâche particulière. Ainsi, d'un point de vue sémantique que nous détaillons dans le cinquième chapitre (page 142), ces sous-graphes contextuels organisent le plan de diagnostic/action correspondant à une activité secondaire. Nous extrayons donc chacun de ces sous-graphes communs et les remplaçons dans les graphes dont ils sont issus par des activités (au sens de composant des graphes contextuels, défini dans la section 2.2, page 95). Les sous-graphes ainsi extraits sont alors associés aux descripteurs des activités correspondantes. Nous obtenons ainsi un ensemble d'activités qui font référence les unes aux autres, chacune décrivant les stratégies possibles pour atteindre un but précis en fonction de la situation et de son évolution. La Figure 3–10 représente l'exemple que nous suivons dans ce chapitre suivant le modèle complet des graphes contextuels (y compris les composants « activité »).

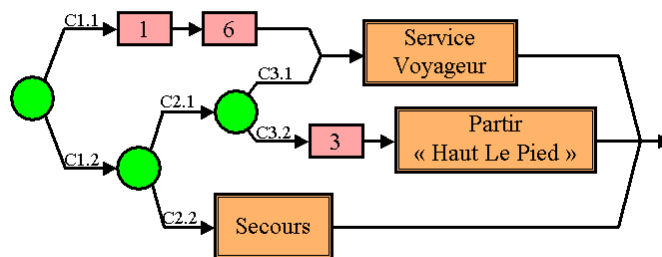


Figure 3–10: Graphe contextuel avec activités

Cette opération d'extraction des sous-graphes contextuels communs représentant des raisonnements locaux structurant les diagnostics et les actions mis en œuvre pour la réalisation de buts intermédiaires peut également être étendue aux graphes contextuels les plus généraux. Chaque graphe contextuel est alors inclus dans un descripteur d'activité pouvant être appelé par le graphe contextuel inclus dans un autre descripteur d'activité par le biais d'un composant « activité ». Nous obtenons ainsi un ensemble de descripteur d'activité, chacun contenant un graphe contextuel structurant le diagnostic, les actions et les activités secondaires nécessaires à la réalisation d'un but en fonction de la situation.

Les macro-actions étaient les prémisses de cette opération d'extraction des sous-procédures. La différence avec les macro-actions se situe dans la structure contenue dans ces activités. Il ne s'agit plus de simples séquences linéaires d'actions, mais de graphes contextuels à part entière. Les macro-actions précédemment utilisées ne sont en fait que des cas particuliers de ces activités où l'ensemble des chemins d'action actuellement connus se

réduit à un singleton et qu'aucun diagnostic supplémentaire n'était donc nécessaire pour choisir quelles actions entreprendre. Cette étape supplémentaire règle du coup le problème soulevé précédemment avec l'évolution des pratiques et l'incapacité des macro-actions à tenir compte de ces évolutions. En effet, en remplaçant les macro-actions par ces activités, même ne contenant qu'une séquence d'actions, nous prévoyons l'acquisition de nouvelles pratiques possibles pour atteindre le but souhaité.

4.2 Structuration de l'ensemble des activités

Lors d'un parcours du graphe à la recherche d'une stratégie on ajoute une séquence d'actions obtenue par le parcours du graphe contextuel contenu par les descripteurs d'activité rencontrés, et ce de manière récursive jusqu'à ce qu'aucune activité ne reste non développée. La finitude de l'algorithme n'est alors plus garantie sauf si on limite l'appel des activités entre elles. Une activité ne doit ainsi pas être appelée ni par elle-même ni par une activité appelée par la suite. Cette contrainte semble lourde, mais une structuration de l'ensemble des descripteurs d'activité permet de garantir l'absence de tels appels récursifs ou croisés. Une structure possible consiste en un graphe orienté acyclique dont les sommets sont les descripteurs d'activité et les arcs correspondent à la relation « fait appel à ». La Figure 3–11 montre ce graphe correspondant à notre exemple récurrent complété par le descripteur de l'activité *gestion de l'incident « freinage »*, qui ressemble beaucoup à notre cas d'étude.

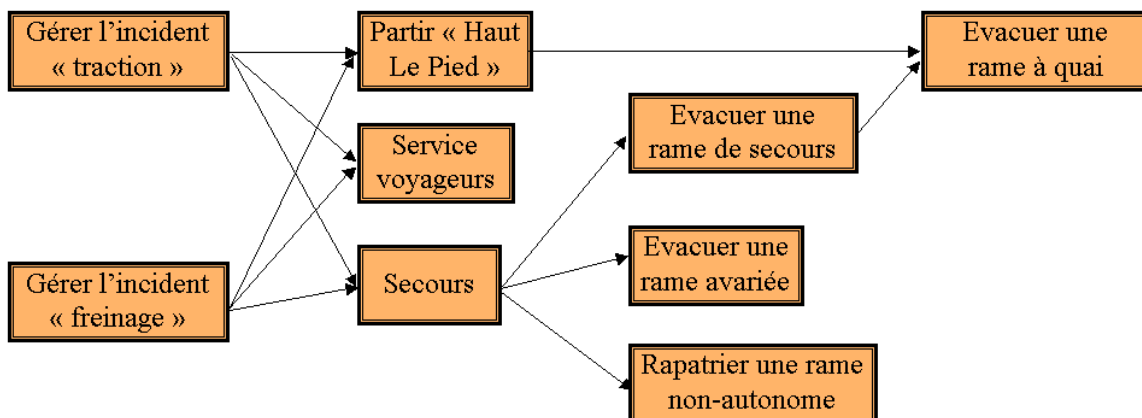


Figure 3–11: Structuration de l'ensemble des activités

Les descripteurs d'activité ainsi mis en exergue montrent toute la dimension du modèle. En réalité chaque action élémentaire pourrait être détaillée sous cette forme. Ceci met en avant le niveau de précision de la modélisation et l'importance du choix de la limite entre activités et actions élémentaires. Ce choix doit être réalisé de manière judicieuse et se limiter aux actions considérées comme élémentaires pour les utilisateurs concernés par le système. On retrouve ici les questions soulevées par Chandrasekaran [1986] concernant les tâches génériques.

4.3 Conséquences sur l'acquisition incrémentale de connaissances

L'utilisation de graphes contextuels pour organiser les différentes stratégies liées aux descripteurs d'activité introduit une certaine réflexivité dans la définition des deux structures : chaque descripteur d'activité est associé de manière bijective à un graphe contextuel et les graphes contextuels peuvent faire appel à des descripteurs d'activité pour réaliser des étapes intermédiaires. L'introduction de descripteurs d'activités dans un graphe contextuel pose la question de l'acquisition incrémentale de connaissances stratégiques et de sa réalisation en pratique dans des descripteurs d'activité emboîtés.

L'acquisition de connaissances stratégiques peut être réalisée suivant deux grands principes : soit elle est réalisée d'une manière globale en considérant que la pratique est un tout et concerne l'ensemble des descripteurs d'activités impliquées ; soit elle est réalisée localement par descripteur d'activité en décomposant la pratique suivant les activités impliquées.

Le premier principe peut être mis en œuvre, par exemple, en remplaçant dans le graphe contextuel de départ, toutes les activités (au sens composant du graphe) par le graphe contextuel correspondant, puis en exécutant l'algorithme précédemment décrit. Cette méthode peut être améliorée en ne remplaçant chaque activité par son graphe contextuel que lorsqu'elle est rencontrée sur le chemin établissant la stratégie. Une fois l'algorithme appliqué, il faut « ré-extraire » les graphes contextuels correspondants aux différentes activités. Les méthodes établies suivant ce premier principe rencontrent alors deux grandes difficultés : premièrement que faire des actions qui ont été introduites à la frontière des activités ; deuxièmement (et beaucoup plus grave) comment gérer des courts-circuits qui pourraient être introduits entre différentes activités. En effet, la pratique ayant été insérée sans tenir compte des frontières entre les activités ni même de la structuration de ces activités entre elles, l'algorithme d'acquisition incrémentale de connaissances peut passer ces frontières. Le raisonnement contextuel n'est alors plus local, car des choix faits dans le cadre d'une activité influe sur des actions entreprises dans le cadre d'une autre. Le premier problème peut être rapidement traité en demandant à l'utilisateur de préciser l'appartenance des nouvelles actions insérées aux différentes activités (toujours au sens d'objet du graphe contextuel) concernées. Le second est plus problématique, mais n'est pas insurmontable. Par exemple, on peut décider que tout court-circuit doit se terminer avant la limite des activités, et que dans le cas contraire un nœud de recombinaison et un nœud contextuel sont ajoutés à la frontière, le premier terminant la première activité et le second commençant la deuxième. Le nœud contextuel ajouté peut, par exemple être une copie du nœud contextuel ayant introduit la divergence ou provenir d'une requête auprès de l'utilisateur. L'ajout de paires de nœuds de recombinaison et de nœuds contextuels est répété ainsi jusqu'à élimination du problème. Cette opération consiste à

modifier le graphe contextuel, quitte à ajouter des nœuds contextuels portant sur des connaissances contextuelles identiques, afin de rendre de nouveau local le raisonnement lié à chaque activité. Ceci est en fait un moyen de retrouver la structure en fuseau des différents sous-graphes contextuels concernés.

Le second principe s'applique de la manière suivante : lors de la saisie de la pratique, l'utilisateur doit structurer les actions en fonction de leur appartenance aux différentes activités, puis l'algorithme d'acquisition incrémentale de connaissances est appliqué indépendamment à chaque descripteur d'activité. Cette méthode a l'avantage de conserver la structuration des activités. Malheureusement la décomposition de la pratique en ses composantes correspondant aux différentes activités est une opération délicate : l'opérateur n'est pas toujours conscient des activités mises en jeu. Cette difficulté peut être contournée par une interaction entre le système et l'utilisateur. Il suffit en effet de guider l'opérateur lors de la saisie de la pratique. Dans notre cas, par exemple, le système d'aide doit connaître les caractéristiques principales de l'incident pour savoir, avant même la saisie de la pratique, quelle activité principale est concernée. Puis il organise la saisie de la pratique correspondant à l'activité principale en parallèle du parcours du graphe contextuel en conservant les activités secondaires comme simples références (comme si elles n'étaient que des actions). Le système d'aide reproduit ensuite cette phase de saisie des composantes de la pratique pour chaque activité rencontrée. L'utilisateur étant guidé par le système d'aide prend ainsi conscience des activités mises en jeu et des composantes de la pratique globale correspondant à chaque activité. Une fois que toutes les composantes de la pratique sont déterminées, les détails de la pratique sont demandés en représentant la structure des activités emboîtées.

Nous préférons de loin cette deuxième méthode qui permet de conserver la structure des activités et, de plus, fait prendre conscience à l'opérateur des activités inconsciemment mises en jeu.

Section 5 :

Conclusion

Le modèle des graphes contextuels est issu de la nécessité d'introduire le contexte dans une structure représentant les différentes méthodes connues pour mener à bien une tâche. Les graphes contextuels sont des graphes orientés, acycliques ayant une source et un puits uniques composés de cinq types de sommets : les actions, les activités, les nœuds contextuels, les nœuds de recombinaison et les branchements temporels. Dans un graphe contextuel, des instances de ces cinq types de sommets sont reliées par des arcs indiquant la précedence

temporelle directe. Les graphes contextuels représentent les diverses situations possibles, l'évolution d'une situation initiale à une situation finale en détaillant divers éléments de la situation et les différents chemins d'action permettant de passer de la situation initiale à la situation finale.

Les graphes contextuels sont des représentations de raisonnements locaux permettant d'établir une stratégie en fonction de la situation. Ils représentent des structures de diagnostic/action, le diagnostic étant représenté par les nœuds contextuels et l'action étant représentée par différents chemins d'action composés d'une séquence d'actions et d'activités secondaires. Avant tout diagnostic, la situation initiale est commune, car aucune connaissance contextuelle n'est mise en avant pour différencier les cas (ceci est opéré par le diagnostic). Les graphes contextuels commencent donc par un même « bout de chemin » représentant les actions devant être entreprise avant tout diagnostic.

La réalisation d'actions intermédiaire permet de réduire l'écart entre des situations précédemment différentes et autorise l'application de séquences terminales identiques. Dès qu'un tel cas se présente, le graphes contextuel opère une recombinaison des chemins d'action par le biais d'un nœud de recombinaison. Chaque chemin d'action représente les différentes méthodes connues pour la réalisation d'une tâche. Ainsi la situation finale est commune à tous les chemins d'action. Les graphes contextuels se terminent donc tous par un unique « bout de chemin » représentant les actions à réaliser pour atteindre le but souhaité, quelle que fut le résultat du diagnostic et les actions entreprises pour revenir à la situation commune.

Tous les chemins d'action contenus dans un graphe contextuel partent ainsi d'une situation initiale commune, divergent petit à petit en fonction du diagnostic établi et convergent quand les actions entreprises réduisent l'écart entre les situations. Ils se terminent tous par l'atteinte du but souhaité. Les graphes contextuels ont ainsi une forme globale de fuseau ou de série de fuseaux, dus à la divergence des chemins d'action créée par le diagnostic et la convergence obtenue par la réduction d'écart de situation grâce aux actions et aux activités entreprises.

Les graphes contextuels partagent parfois des sous-graphes contextuels représentant une structure de diagnostic/action ayant les mêmes propriétés que les graphes contextuels eux-mêmes. Ces sous-graphes contextuels représentent en réalité des raisonnements locaux associés à des buts intermédiaires. Ces structures de diagnostic/action sont justement ce qui distingue une action d'une activité. Les actions et les activités partagent certaines caractéristiques (un nom, un but, des pré-conditions, des résultats), mais se distinguent sur le degré de détail accordé à la description de leur mise en œuvre : les actions ne sont pas détaillées, alors que les activités nécessite une part de planification. Nous proposons donc d'extraire ces sous-graphes contextuels et de les associer à un composant « activité » des graphes contextuels. Dans notre modèle, chaque activité est associée de manière bijective à un

graphe contextuel représentant la structure de diagnostic/action pouvant lui être associée. Les graphes contextuels font également référence aux activités, ce qui crée une réflexivité des définitions des deux structures (graphe contextuel et structure de l'activité). Cette association organise les activités à deux niveaux : premièrement au niveau de chaque activité, le graphe contextuel organise la planification dynamique induite par cette activité ; deuxièmement au niveau de l'ensemble des activités mêmes, nous obtenons un réseau d'activités établi suivant leur dépendance. Ce réseau est utile pour garantir la terminaison des algorithmes (il suffit de vérifier qu'il n'y a pas de circuit dans ce réseau) mais également pour structurer les raisonnements produits sur la base de ce modèle.

Nous proposons également un algorithme d'acquisition incrémentale de connaissances stratégiques pour l'enrichissement des méthodes associées aux différentes activités décrites, en fonction des pratiques réellement observées. Cet algorithme simple permet d'ajouter facilement des pratiques à un ensemble de graphes contextuels. Deux points cruciaux sont à retenir pour l'acquisition de l'expertise des opérateurs : premièrement l'identification des éléments de contexte ayant induit les adaptations proposées par la nouvelle pratique, et deuxièmement la répartition des composants des pratiques entre les différentes activités. Cet algorithme repose sur une forte interactivité avec l'utilisateur afin d'acquérir une part de l'expertise des opérateurs. Cette expertise est en relation directe avec l'identification des connaissances contextuelles et l'organisation des activités. L'interaction système/utilisateur est d'autant plus forte que le système est capable de guider l'utilisateur dans la saisie des différentes composantes de la pratique concernant les activités impliquées, grâce à la structuration de l'ensemble des activités.

Ainsi parés, les graphes contextuels sont un modèle de représentation de structure de diagnostic/action et de l'évolution des pratiques associées, en fonction de la situation. L'enrichissement progressif de ces structures par l'acquisition des pratiques réelles des opérateurs permet d'affiner le diagnostic par l'intégration de nombreuses connaissances contextuelles. La structure des graphes contextuels repose sur l'évolution de la situation en cours de traitement et non de la situation initiale. Le dynamisme des raisonnements reposants sur ce modèle est réel et joue à deux niveaux : premièrement, les graphes contextuels permettent d'établir une stratégie en fonction du contexte et de son évolution en cours de traitement ; deuxièmement, les graphes contextuels permettent d'intégrer les nouvelles pratiques et de les organiser par rapport aux stratégies précédentes en fonction du contexte d'application de chacune. Les graphes contextuels sont ainsi des structures de diagnostic/action fortement liés à la notion de contexte et à son dynamisme et autorisant une prise en compte de l'évolution des pratiques.

Quatrième chapitre :

Représentations de raisonnements et contexte

Lors de travaux sur des systèmes inspirés de la cognition humaine, l'intelligence artificielle se heurte à une difficulté particulière : la représentation de connaissances sur les raisonnements. Cette représentation est complexe, principalement parce qu'il est difficile d'accéder aux connaissances stratégiques des individus. De nombreuses voies ont été explorées, et ont abouti à des modèles plus ou moins satisfaisant de raisonnements. Malheureusement, les modèles obtenus ont tendance à oublier un élément particulièrement important dans les situations réelles : le contexte.

L'introduction du contexte dans les raisonnements artificiels est une étape actuellement difficile. Plusieurs voies ont toutefois été étudiées et fournissent des résultats intéressants. Nous étudions dans ce chapitre les diverses voies déjà explorées pouvant être mises en relation avec notre modèle de graphe contextuel.

Nous présentons dans un premier temps les approches pluridisciplinaires structurant les connaissances de haut niveau relatives aux raisonnements et à la stratégie. C'est le cas des Schemas, des Frames, des Scripts, des MOPs et des schèmes. Nous ferons cas également d'un modèle de raisonnement contextuel peu structuré : le raisonnement à partir de cas. En ce qui concerne l'implémentation, notre modèle est plus proche de planifications plus ou moins dynamiques : diagramme d'activité d'UML, décomposition des tâches et planification dynamique. Nous étudions également le cas de quelques modèles de structure proche des graphes contextuels mais qui ne sont pas applicables dans notre cas. Ceci est le cas des diagrammes d'influence et des réseaux de Petri.

Section 1 :

Introduction

1.1 Motivations

Notre étude est réalisée dans le but de concevoir un système d'aide intelligent en contexte pour la gestion d'incident sur une ligne de métro. Les systèmes d'aide doivent être capables de comprendre les activités des opérateurs afin de pouvoir les aider efficacement. Pour cela il est nécessaire d'introduire des modèles de raisonnement des opérateurs dans ces systèmes (premier chapitre).

La gestion d'incidents sur une ligne de métro est une activité complexe faisant appel à différentes activités secondaires. Les opérateurs établissent des plans d'action en fonction de l'état actuel du processus à réguler, des règlements qu'ils doivent respecter, de procédures établies par l'entreprise, de leurs propres expériences et de celles de leurs collègues. Les raisonnements appliqués par les opérateurs, de type planification de diagnostic/action, sont hautement sensibles au contexte et à son évolution au cours du traitement. De plus, leurs pratiques évoluent rapidement en fonction des pratiques précédemment réalisées soit par eux, soit par leurs collègues (deuxième chapitre).

Nous proposons le modèle de graphes contextuels pouvant représenter des structures de diagnostic/action. Les graphes contextuels sont associés à divers algorithmes, dont le principal permet d'ajouter des nouvelles pratiques afin de maintenir à jour les connaissances stratégiques représentées par celui-ci (troisième chapitre). Le modèle de graphe contextuel est ainsi une structuration de connaissances sur des raisonnements fortement dépendants du contexte et évolutifs. La caractéristique principale du modèle est l'évolution des raisonnements qu'il représente en fonction des expériences passées. Les connaissances stratégiques organisées dans les graphes contextuels permettent ensuite d'établir des plans d'action de manière dynamique en fonction du contexte et de son évolution.

Dans ce chapitre, nous étudions les travaux apparentés à notre modélisation de raisonnements stratégiques fortement dépendant du contexte. Dans un premier temps, nous nous consacrons aux travaux réalisés sur la structuration de connaissances sur les raisonnements. Ces travaux sont principalement issus de la recherche en psychologie cognitive ou de propositions d'applications de ces résultats à l'intelligence artificielle. Nous étudions ensuite le raisonnement à partir de cas qui est un raisonnement par analogie fortement lié au contexte. Enfin, nous ferons cas de travaux en intelligence artificielle pouvant s'apparenter aux graphes contextuels. Cette dernière section présente dans un premier temps des structures proches des graphes contextuels mais qui ne sont pas applicables dans notre cas, puis nous faisons un point sur différents modèles plus proches de notre point de vue.

1.2 Liens entre les représentations et l'objet représenté

Etablissons d'abord un cadre à ces études. Le but de notre thèse est de proposer une représentation efficace de raisonnements stratégiques appliqués par des opérateurs chargés d'une tâche complexe. Les travaux de Davis, Shrobe et Szolovits [1993] sur les liens entre les représentations et les objets représentés nous semblent intéressants dans cette perspective. Ces travaux partent du principe qu'une représentation est un substitut de l'objet dont le rôle est de pouvoir inférer des conséquences par des raisonnements plutôt que par des actions directes sur l'objet. Ce point de départ met en évidence que toute représentation, par définition, n'est

pas l'objet représenté. Ceci implique qu'il y a des différences plus ou moins importantes entre une représentation et son objet, et que le choix d'une représentation n'est pas anodin.

De plus, en ce qui concerne la représentation de raisonnements intelligents, les représentations possibles sont très différentes suivant l'approche suivie (mathématique, psychologique, biologique, statistique ou économique). Le choix d'une approche limite le type des représentations admissibles et impose une conception particulière quant à la nature fondamentale du raisonnement représenté. Le choix de l'approche limite d'hors et déjà le choix des représentations possibles.

Toutefois, la représentation est le support des raisonnements qui seront appliqués par la suite pour inférer des conséquences sur l'objet. Les représentations et les raisonnements sont fortement liés : la définition de l'un se fait sur celle de l'autre. Les deux ne peuvent être définis ou traités séparément. Ainsi il est capital de bien choisir la représentation de l'objet pour avoir une chance d'obtenir un raisonnement efficace. De plus les représentations sont les seuls supports d'échanges de connaissances entre les différents acteurs, que ceux-ci soit humains ou informatiques : il n'est pas possible d'échanger des connaissances quand aucune représentation n'est commune à l'émetteur et au récepteur du message contenant les connaissances.

Enfin, les représentations sont des projections de l'objet réel et omettent inévitablement une partie des caractéristiques de celui-ci. La combinaison de représentation est une manière efficace d'augmenter les capacités de raisonnement sur les objets représentés et proposant plusieurs points de vue. Ceci est efficace uniquement si la combinaison ne réduit pas les représentations, mais les traite plus ou moins en parallèle.

Section 2 :

Structuration des connaissances sur les raisonnements

Cette section étudie les modèles issus ou inspirés de travaux en psychologie cognitive sur la modélisation de raisonnements humains. Le choix de l'approche psychologique permet de mettre l'accent sur les structures mentales des acteurs humains. Ceci est donc un point de départ d'un modèle facilitant les échanges entre un utilisateur et une machine.

Nous étudions d'abord les schémas, qui sont une première structure proposée par la psychologie cognitive pour représenter des connaissances de niveau stratégique. Nous avons choisis de commencer par cette structure, car elle a donné naissance à deux branches applicatives en intelligence artificielle : premièrement les Frames, deuxièmement les

Scripts puis les MOPS. Nous étudions ensuite les schèmes, qui sont une structure de connaissances stratégiques très proche des schémas, mais qui s'en distinguent par le rôle central que joue leur évolution dans le temps.

2.1 Schéma

Barlett [1932 ; 1958] introduit la notion de schéma dans l'étude de l'organisation de la mémoire. Selon lui, la mémoire prend la forme de schémas qui fournissent un cadre pour comprendre et retenir les informations. Les schémas sont des structures abstraites mettant en relation différents objets, ce qui permet, au fur et à mesure du remplissage de ce cadre abstrait par des connaissances spécifiques à une situation, d'en appréhender le sens. L'instanciation progressive du schéma par les connaissances sur la situation permet aussi de concentrer son attention sur des objets importants pour la compréhension de la situation. Une fois la situation analysée, cette expérience est retenue en enregistrant les instanciations des différents objets du schéma.

Selon d'Andrade [1995], les schémas sont des ensembles de slots apparentés qui peuvent être remplis par des informations supplémentaires ou des éléments du contexte. L'auteur précise que la modification de la valeur d'un slot modifie souvent les valeurs admissibles pour des slots en relation avec celui-ci. Ces relations de limitation des valeurs entre slots sont importantes, car il est ainsi possible de comprendre complètement une situation sans pour autant avoir d'information directe pour chacun des slots composant le schéma correspondant. En effet, certaines informations capitales suffisent pour appréhender totalement la situation.

Langacker [1987] précise que quand aucune information spécifique n'est connue pour un slot, celui-ci peut être rempli avec des valeurs par défaut. Ces valeurs par défaut sont des attentes normales quant à la situation représentée. Elles permettent de saisir plus rapidement une situation, mais peuvent induire des erreurs ou des malentendus. Un schéma rempli uniquement par des valeurs par défaut est appelé prototype et représente l'archétype d'une situation. Un prototype est une instance hautement spécifique d'un schéma.

L'un des aspects importants des schémas est qu'un schéma simple peut être inclus dans un schéma plus complexe. Ainsi les schémas peuvent être organisés hiérarchiquement en fonction de leur relation d'appartenance. Cette décomposition modulaire de la compréhension de situation est un point important à nos yeux, car il facilite la compréhension en l'organisant à un niveau supérieur.

L'étude des schémas dans diverses situations courantes fait ressortir leur caractère général. Ainsi les schémas sont considérés comme des composantes importantes des différences culturelles par Quinn et Holland [1987]. Cette remarque nous semble d'autant plus importante

qu'elle peut s'appliquer autant à des cultures liées aux populations qu'à des cultures liées à des professions, et donc aux communautés de pratiques, telles que celle des chef de régulation du métro parisien. La recherche sur les performances comparées des experts et des novices menée par Chi, Glaser et Farr [1988] montre que la nature de l'expertise est largement due à la possession de schémas qui guident la perception et la résolution de problèmes. Cette remarque nous concerne directement, puisque nous cherchons à représenter l'expertise des opérateurs en vue de l'exploiter dans un système d'aide.

2.2 Frames

La première application informatique des schémas étudiés en psychologie cognitive correspond aux Frames proposés par Minsky [1975]. Un Frame est une structure de données qui représente une situation stéréotypée. Chaque Frame est associé à différentes sortes d'informations : certaines à propos de l'utilisation du Frame, d'autres à propos des attentes possibles, et d'autres à propos de ce qui peut être fait lorsque ces attentes sont infirmées. Les Frames sont des réseaux de nœuds et de relations. Les plus hauts niveaux du Frame sont fixés et représentent des choses qui sont toujours vraies à propos de la situation supposée. Les plus bas niveaux sont composés de nombreux slots terminaux qui doivent être remplis par des instances ou des données spécifiques. Chaque slot terminal peut spécifier les conditions devant être respectées par son assignation. Les assignations sont souvent de petits sous-Frames.

Les Frames en rapport les uns avec les autres sont rassemblés dans des systèmes de Frames. Les différents Frames d'un système de Frames partagent les mêmes slots terminaux, afin de coordonner les informations accumulées par différents points de vue. La plus grande partie du pouvoir de cette théorie dépend de l'inclusion des attentes et d'autres formes de présomptions. Les slots terminaux d'un Frame sont initialement remplis avec des assignations par défaut. Ces assignations par défaut sont faibles et peuvent être remplacées par des items rendant mieux compte de la situation.

Les systèmes de Frames sont de plus liés entre-eux par un réseau d'extraction d'informations. Quand un Frame proposé ne peut pas être instancié de manière à représenter la réalité, le réseau propose un Frame de remplacement. Quand un Frame est proposé, un processus d'assortiment tente d'assigner des valeurs à chaque slot terminal, consistantes avec les marqueurs déjà instanciés. Cette instanciation est guidée à la fois par les informations associées au Frame et par les buts du système.

Selon Minsky, l'idée à la base des Frame n'est pas particulièrement originale (elle découle des schémas) ; son originalité réside dans les systèmes de Frames. Ce que nous retiendrons de

cette première application informatique des schémas est l'organisation en systèmes de Frames, qui permettent de représenter différents points de vues d'une même situation. Malheureusement, ces structures ne permettent que d'analyser une situation en organisant les différentes informations que l'on peut obtenir sur celle-ci. Les Frames ne peuvent pas être utilisés seuls dans notre application car ils ne proposent pas de plan d'action en fonction de la situation et ne se chargent que du diagnostic. De plus, les Frames sont définis *a priori* et ne sont pas conçus pour évoluer avec l'expérience.

2.3 Scripts

Le second modèle applicatif des schémas correspond aux Scripts de Schank et Abelson [1977]. Les Scripts sont composés d'acteurs, d'une scène, de supports (des actions), et d'une suite d'événements attendus. Les actions contenues dans un script peuvent faire appel à un autre script. Un script est une structure qui décrit des séquences appropriées d'événements dans un contexte particulier. Comme les Frames, il est composé de slots et de contraintes sur le remplissage de ces slots ; il forme un tout interconnecté et ce qui est dans un slot peut affecter ce qui peut être mis dans un autre. Une grande différence avec les Frames est que les Scripts contiennent non seulement des liens entre les divers objets composant la scène, mais aussi les séquences d'actions prévues pour la scène en fonction des événements.

Comme les Frames, les Scripts sont instanciés lors de l'appel et leurs slots remplis au fur et à mesure de l'arrivée de nouvelles informations sur la situation. Les Scripts sont initialement remplis par des valeurs par défaut. Les liens entre les différents slots permet alors de supposer des faits et gestes à partir de quelques informations seulement. Dans la compréhension d'histoires (application étudiée par Schank et Abelson), un script permet d'inférer des événements probables à partir des informations fournies et de compléter l'histoire. Les non-dits sont remplacés, en supposant que ceux-ci correspondent au cas le plus courant.

Les Scripts autorisent la référence directe à des acteurs, des faits et des événements alors qu'ils n'ont pas été explicitement introduits, grâce à ces instanciations partielles par défaut. L'application des Scripts à la compréhension d'histoires de la vie courante autorise de nouvelles références à des objets contenus par les Scripts comme si ces objets avaient été mentionnés auparavant : les objets contenus dans un Script peuvent être déterminés par des articles définis sans introduction explicite car le script lui-même les introduit.

En fonction des différents rôles des acteurs, l'histoire n'est pas perçue de la même manière. Considérant une scène au restaurant, celle-ci n'est pas interprétée par un client comme elle l'est par la serveuse ou le cuisinier. Un script doit donc être décrit en prenant le point de vue d'un rôle particulier. Les scripts provenant des différentes perspectives sont combinés et

forment ce qui peut être considéré comme une vue d'ensemble de la scène. Une telle vue d'ensemble est rarement utile pour la compréhension usuelle, bien qu'elle puisse constituer ce qui peut être considéré comme un concept. Elle est donc utile pour analyser une scène dans sa globalité en recherchant les interactions entre les différents acteurs et pour comparer leurs points de vue.

Les Scripts reposent fortement sur des suppositions *a priori* sur les événements à venir. Des événements inattendus peuvent interférer avec le déroulement du script. Ces événements peuvent soit contredire les attentes du script et modifier les conclusions de celui-ci, soit être indépendants. Pour savoir comment interpréter ces événements inattendus, il est nécessaire d'attendre des informations supplémentaires sur ces événements.

Les Scripts constituent un modèle d'interprétation de situation complexes intégrant des éléments de contexte et des actions. Ils permettent de comprendre une histoire (ou une série d'événements) grâce à l'établissement de liens entre les informations perçues et des événements attendus. Ils sont également organisés hiérarchiquement, ce qui facilite la compréhension de scènes complexes par la décomposition en petites scènes. Pour notre application, les Scripts semblent intéressants par leur structure associant des acteurs à une scène et à des événements attendus. De plus ils présentent un plan d'action à réaliser en fonction des événements. Un autre point fort des Scripts est leur effet structurant descendant : un Script est d'abord identifié, puis instancié et donne enfin accès à des événements attendus. Ainsi la structuration proposée par les Scripts va de l'aspect le plus général aux détails de la situation.

Malheureusement les scripts ne sont pas sujets à des évolutions importantes et ne sont pas adaptés au maniement de situations totalement nouvelles, ni aux situations aux variantes infinies telles que les pratiques que nous observons. Un script est une séquence, prédéterminée et stéréotypée, d'actions qui définissent une situation bien connue, si bien qu'ils ne sont pas directement applicables pour modéliser des raisonnements en évolution permanente tels que ceux rencontrés lors de la gestion d'incident dans le métro. Selon Schank et Abelson [1977] eux-mêmes, « *l'acceptation d'une théorie de l'entendement basée sur les scripts passe par l'acceptation de certaines restrictions fortes : afin de comprendre les actions se déroulant dans une situation, la personne doit avoir déjà rencontré cette situation. La compréhension est basée sur la connaissance. L'être humain est vu comme un processeur capable de comprendre seulement ce qu'il a déjà compris* ». Les Scripts ne prévoient pas d'évolution simple et rapide de leurs structures en fonction des situations nouvelles rencontrées. L'opération d'adaptation de Scripts à de nouvelles situations est coûteuse.

2.4 MOPs

Schank [1982] complète sa vision des structurations de la mémoire humaine par l'introduction de la notion de *Memory Organization Packet* (MOP). Ceci s'inscrit dans une conception plus dynamique de la mémoire : chaque être humain peut modifier son système de classification interne facilement quand ses intérêts changent ou quand ses connaissances sont modifiées. Ce n'est pas actuellement le cas pour de nombreux systèmes de classification tels que les bibliothèques, par exemple. Ces systèmes de classification nécessitent une intervention extérieure pour être modifiés et cette opération est longue et fastidieuse. De plus un être humain est conscient de lui-même et de ce qu'il sait. La mémoire humaine est dynamique. De nouvelles expériences peuvent infirmer des classifications issues de conceptions initiales. Nous ne pouvons pas toujours savoir comment des informations reçues pourront être utilisées plus tard. Nous apprenons à partir d'expériences multiples. Nos mémoires changent dynamiquement dans le sens où elles enregistrent des informations par abstraction et généralisation significative de nos expériences et enregistrent les exceptions de ces généralisations. Tout ceci est réalisé sans intervention extérieure. Les êtres humains ne sont pas conscients des schèmes de catalogage mais sont juste capables de les utiliser.

Les Scripts proposés par Schank et Abelson [1977] ne permettent que de comprendre efficacement des situations. Schank [1982] cherche à ce que ses systèmes apprennent à partir de leurs expériences. La distinction entre les Scripts proposés en 1977 et d'autres structures de connaissances de haut niveau tenait aux notions d'abstraction et de généralisation : les Scripts étaient des informations spécifiques associées à une situation spécifique. Par ailleurs, les êtres humains ont des informations générales associées à des situations générales. L'acquisition de ces informations générales se fait d'une manière complexe (par abstraction et généralisation d'expériences multiples et par les expériences des autres. La différence entre une mémoire basée sur des Scripts et une mémoire basée sur des structures plus générales est que, dans le premier cas, ce que nous savons dans une situation provient de ce que nous avons déjà vécu dans des situations plus ou moins identiques. L'utilisation de structures plus générales nous autorise à utiliser des informations engrangées dans une situation pour nous aider dans une situation totalement différente. L'un des inconvénients des Scripts est leur manque d'adaptabilité d'une situation à une situation similaire mais non identique. D'un autre côté, les Scripts ont un avantage: nous avons des connaissances spécifiques à des situations particulières. Les informations généralisées, qui éliminent les différences particulières, perdent de l'information, et cette information peut se révéler fort utile.

Schank [1982] propose les MOPs pour représenter une mémoire dynamique. Un MOP est un organisateur de scènes, une scène étant une structure de mémoire qui regroupe des actions avec un but commun qui ont eu lieu en même temps. Chaque scène fournit une séquence

d'actions générales ; des souvenirs spécifiques sont enregistrés dans les scènes, indexées suivant leurs différences par rapport aux actions générales de la scène. Les MOPs ne contiennent pas explicitement de souvenir. Ils organisent des scènes qui contiennent des souvenirs. Les Scripts représentent, suivant le modèle de 1982, des instanciations usuelles de scènes. Ainsi, une scène est composée d'une séquence générale d'actions, alors qu'un Script (1982) représente des réalisations particulières des généralisations d'une scène.

Les MOPs et les Scripts (1977) sont tous deux des ensembles organisés de scènes. La différence réside dans la définition d'une scène. Pour les MOPs, une scène est une structure qui peut être partagée par de nombreux autres MOPs. Pour les Scripts (1977), une scène était particulière à un script donné et n'était pas accessible sans passer par le script. Les MOPs permettent ainsi une généralisation de scènes communes à plusieurs MOPs totalement différents. Par exemple la scène *d'acheter un billet* est commune aux MOPs de *prendre l'avion*, *prendre le bus* ou *aller à un concert*.

Cette vision beaucoup plus modulaire que celle des Scripts (1977) nous intéresse particulièrement. En effet, nous avons observé que de nombreuses activités des opérateurs ont des activités secondaires communes. Les MOPs permettent de bien représenter les différentes activités et leurs relations et organisent ces activités de manière hiérarchique. Les activités ainsi représentées peuvent être partagées par des activités plus générales. Chaque activité est associée à un plan d'actions à réaliser en fonction de la situation. Toutefois, les MOPs restent difficiles à modifier. La différence entre les Scripts (1977) et les MOPs réside dans leur relation avec les scènes, mais rien n'est changé quand à la structure de ces scènes. Ainsi celles-ci sont toujours récalcitrantes à une modification structurelle opérée en fonction des expériences. Les MOPs sont des structures plus efficaces que les Script de 1977 pour organiser les souvenirs, mais n'offrent pas encore le dynamisme suffisant pour obtenir une réelle mémoire dynamique.

2.5 Schèmes

Les schèmes ont été définis par Kant vers 1800 [Eco, 1997]. Ces structures mentales sont des enregistrements de pensées, d'actions et de résolutions de problèmes passés réutilisés par les êtres humains pour interagir avec le monde et résoudre de nouveaux problèmes. Ils permettent d'organiser nos différentes activités en s'appuyant sur nos activités passées.

Selon Pastré [1994], « *les schèmes sont à l'action ce que les concepts sont à la cognition* ». Cette remarque souligne deux points importants sur les schèmes. Premièrement les schèmes sont les supports de nos actions et de leur organisation. Il n'y a pas d'actions ni de planification de ces actions sans l'utilisation de schèmes. Deuxièmement, les schèmes font

partie de notre quotidien, même si nous n'avons pas conscience de leur existence ni de l'utilisation que nous en faisons. Cette dernière remarque est la principale cause de leur mise en évidence difficile dans les activités de chacun d'entre nous (au même titre que les concepts dans nos réflexions, d'ailleurs).

La théorie instrumentale de Rabardel [1995] précise que les instruments ne sont pas donnés aux opérateurs, mais que l'instrument est créé par celui-ci à partir de l'artefact (l'objet physique) et de schèmes d'utilisation. Les schèmes d'utilisation construits autour de l'artefact correspondent plus ou moins à l'utilisation prévue par le concepteur de l'artefact. Pour chaque type d'utilisation d'un artefact (c'est-à-dire l'association d'un schème d'utilisation à l'artefact), nous obtenons un instrument spécifique. L'utilisation d'un artefact *tourne-vis* peut ainsi correspondre à deux instruments différents : premièrement, en utilisant un schème classique, celui-ci peut servir à visser (ou dévisser) ; deuxièmement, en adaptant un schème moins classique pour cet artefact, il peut servir pour faire levier pour ouvrir un pot de peinture. L'*artefact* n'est pas l'*instrument* et n'est pas toujours utilisé comme le concepteur l'avait prévu.

Les schèmes d'action sont des unités mentales qui organisent l'activité de la même manière que les schèmes d'utilisation organisent l'usage d'artefact. Les schèmes d'action ont été proposés par Piaget [1936] pour expliquer l'apprentissage chez les jeunes enfants. Les enfants apprennent par essais-erreur et construisent petit à petit une structure organisant leurs actions en fonction d'éléments de la situation. Ces schèmes ont été par la suite identifiés chez l'adulte en situation de travail dans divers domaines et pour diverses activités : dessinateurs [Béguin, 1994], camionneurs [Galinié, 1996], opérateurs sur machine-outils [Duvenci-Langa, 1997]... Les travaux de Zanarelli [1998 ; 2002] identifient également des schèmes d'actions chez les opérateurs qui nous concernent (voir deuxième chapitre, section 4.3, page 82).

Comme nous l'avons esquissé dans le deuxième chapitre, un schème d'action est composé d'une structure des actions permettant d'atteindre un but, et des moyens utilisés pour réaliser ces actions. La structure d'actions peut également faire appel à d'autres schèmes pour réaliser des sous-buts. Vergnaud [1985] décrit les schèmes d'action comme étant un tout organisé de manière dynamique. Selon lui les schèmes d'action sont composés de quatre éléments :

1. Des invariants opératoires : objets, propriétés, relations et processus.
2. Des règles d'action, qui créent et guide l'action.
3. Des inférences et des calculs.
4. Des prédictions, qui sont des buts ou des étapes intermédiaires attendues.

La composition de ces éléments permet de représenter les différentes stratégies possibles pour atteindre un but en fonction des éléments de la situation. Cette notion rappelle celles des Schémas, Frames, Scripts et MOPs qui sont utilisés en intelligence artificielle, mais s'en distingue principalement par sa dimension dynamique. En effet, le dynamisme des schèmes ne se limite pas à une gestion dynamique de l'activité, mais s'étend également au schème lui-même, le principe même des schèmes étant la prise en compte des expériences du passé pour guider les activités futures.

Les schèmes peuvent évoluer de trois manières : par adaptation à la situation, par adition de nouvelles stratégies et par différenciation. L'**adaptation** d'un schème d'action à une situation se fait par « essais-erreurs » ; elle fait intervenir les règles d'action et les adaptes à la situation réelle courante. L'**addition de nouvelles stratégies** enrichit un schème « après coup » en modifiant l'organisation des actions dans l'activité globale. Elle permet d'augmenter les capacités du schème en diversifiant les réponses possibles aux situations futures. Enfin la **différenciation** est la copie d'un schème existant et son adaptation à la réalisation d'une activité apparentée à l'activité du schème initial. C'est sans doute l'évolution la plus originale décrite par les schèmes.

Nous retrouvons dans les schèmes une structuration à deux niveaux des activités, comme celle proposée par les Schémas et ses diverses applications en intelligence artificielle. Le premier niveau correspond à la structure de chaque activité, le second à la structuration des activités entre elles. Tout comme les structures issues des Schémas, les schèmes doivent être instanciés pour obtenir une opérationnalisation de ceux-ci. Cette instanciation correspond à l'adaptation du schème à la situation. Une fois adapté, le schème propose une planification dynamique d'actions pour atteindre un but fixé, et ce en fonction du contexte. Comme pour les structures précédemment étudiées, cette structuration à deux niveaux de planifications dynamiques adaptées à la situation nous intéresse grandement pour notre application.

Les schèmes sont, de plus, construits autour de leurs évolutions. Les deux modes d'évolution des structures des schèmes sont complémentaires. Le premier (l'ajout de nouvelles stratégies) permet d'enregistrer des nouvelles pratiques pour un schème connu et de réorganiser la structure du schème en conséquence. La seconde (la différenciation) permet de créer un nouveau schème pour la réalisation d'une nouvelle activité proche d'une activité déjà connu sans partir d'*ex-nihilo*. Ces deux modes d'évolution des schèmes sont particulièrement intéressant pour notre application. Comme précisé ci-dessus, des schèmes d'action ont été identifiés dans la communauté de pratiques qui nous intéresse. Malheureusement aucune structure informatique n'est actuellement développée pour représenter ces structures mentales. Nous soulignons ici que le développement de *scheme* [Dyer, 1990 ; Stacey, Sharp et Petre, 1996] ne reprend pas l'aspect principal des schèmes, à savoir leurs mécanismes d'évolution.

Section 3 :

Raisonnement à Partir de Cas

Le Raisonnement à Partir de Cas (RàPC) est un raisonnement de type « analogie » [Burstein, 1989 ; Kolodner, 1993 ; Jurisica, 1993 ; Aamodt et Plaza, 1996 ; Mille et Napoli, 1997], où des expériences passées sont mises à contribution pour définir une solution à un problème courant. Ce raisonnement artificiel a pour but de proposer des solutions à un problème faisant partie d'une classe de problèmes couramment rencontrés par le décideur.

3.1 Principes de fonctionnement

Le RàPC repose sur une base de cas où sont enregistrées les situations déjà rencontrées et sur une mesure de similarité entre les situations, qui permet, pour une situation donnée, de retrouver les cas les plus proches dans la base de cas. Chaque cas enregistré est composé des descripteurs du problème et de la solution adoptée. La mesure de similarité est une somme pondérée de calculs locaux de similarité pour chacun des descripteurs possibles des cas. Cette mesure de similarité est ajustable à deux niveaux : premièrement, pour chaque descripteur on doit définir la similarité entre les différentes valeurs possibles (par exemple, par des matrices quand les descripteurs prennent des valeurs discrètes ou par une fonction quand ils prennent des valeurs numériques) ; deuxièmement, on doit définir les poids relatifs des différents descripteurs les uns par rapport aux autres. Cette mesure de similarité doit être définie de manière assez fine pour que les cas retrouvés aient des solutions assez proches d'une solution possible au problème à résoudre.

De nombreuses études ont été réalisées sur le RàPC et de nombreuses applications opérationnelles reposent sur ce principe [Watson, 1995]. L'avantage de ce raisonnement face aux premiers raisonnements utilisés dans les systèmes experts est la gestion simplifiée des bases de connaissances nécessaires au raisonnement. En effet, cette base est constituée des expériences précédentes et à chaque nouvelle utilisation, la mise à jour s'effectue par l'ajout du nouveau cas.

La Figure 4-1 représente le modèle fonctionnel du RàPC donné par Aamodt [1995]. Le principe du raisonnement est le suivant. Un cas courant, composé d'un problème à résoudre et de sa description, est donné par l'utilisateur. Ce cas est comparé aux cas précédemment enregistrés dans la base de cas, chacun d'eux étant composé d'un problème et de sa solution. La comparaison est effectuée par le calcul d'une mesure de similarité, qui est une somme pondérée des comparaisons locales des différents descripteurs d'un problème (ceci correspond à l'étape « retrouver » de la figure). Les cas ayant obtenu les meilleurs scores par cette mesure

sont déclarés plus proches voisins et sont sélectionnés pour construire une solution au problème courant (étape « réutiliser »). Cette solution est alors « révisée » par l'utilisateur, puis introduite dans la base de cas pour être réutilisable pour une prochaine (étape « retenir »). Ces quatre étapes sont les « quatre R » du RàPC [Aamodt, 1995 ; Mille & Napoli 1997].

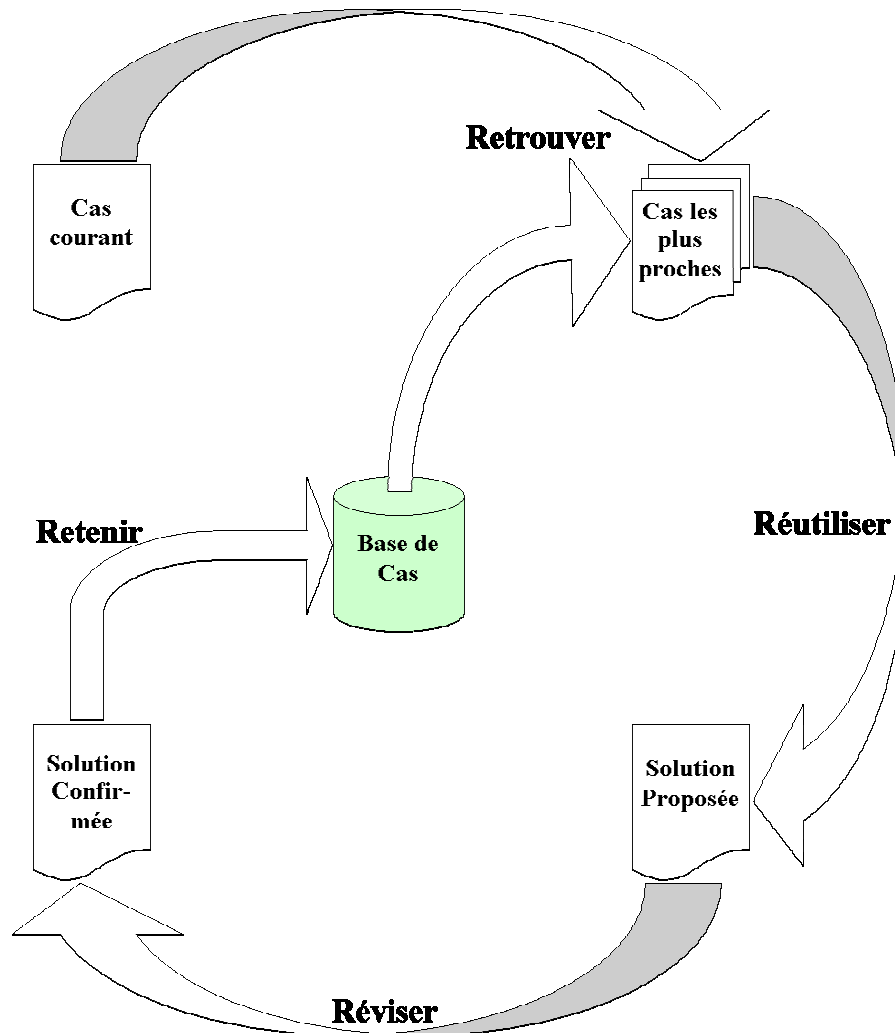


Figure 4-1: Cycle du Raisonnement à Partir de Cas

L'exécution de ce cycle permet de créer, cas après cas, une base de connaissance sur des résolutions de problèmes réels et de réutiliser ces connaissances pour résoudre des problèmes futurs. Les connaissances ainsi réutilisées sont soit directement les solutions des problèmes rencontrés, soit une trace de la construction de ma solution, tel que le propose Leake [1996]. Ce mode de raisonnement est très utilisé par les êtres humains dans de nombreux environnements et joue un rôle capital dans l'expertise. Il est l'archétype du raisonnement par analogie.

Les autres principaux avantages de ce raisonnement sont son pouvoir de généralisation et sa capacité à s'adapter à des données manquantes. La généralisation s'opère par la mesure de

similarité. Deux cas peuvent être jugés semblables par le système alors que nous n'aurions même pas eu l'idée de les rapprocher par ailleurs. La capacité à traiter des données manquantes dépend beaucoup de la définition de la mesure de similarité et de ce qu'elle doit faire dans de tels cas. Toutefois il est possible soit de considérer que ces données sont peu importantes (en associant un poids nul à des données manquantes) soit à considérer qu'elles sont au contraire capitales en augmentant leur poids et attribuant une valeur faible (voire négative) à l'absence d'un tel descripteur dans l'un des cas comparés. Toutes les latitudes sont autorisées entre ces deux extrêmes.

3.2 RàPC et contexte

Selon Kolodner [1993], ce mode de raisonnement prend en compte le contexte de manière explicite, du moment que celui-ci est un élément de description des problèmes. En effet, dans de telles conditions les éléments de contexte participent directement au calcul de la similarité entre le cas courant et les cas enregistrés. Le contexte peut également faire partie intégrante du système pour contraindre la mesure de similarité [Bilgic et Fox, 1996].

Toutefois, ceci n'est que partiellement vrai, car les cas n'enregistrent, dans l'acceptation initiale du RàPC, que des éléments prédéfinis. Ceci ne fait donc qu'au plus une projection du contexte sur un espace de dimension fini préfixé par le concepteur du système, alors que le contexte est *a priori* infini et que l'ensemble des descripteurs de contexte connus est en constante progression. Ceci peut être réglé en proposant une gestion conjointe des éléments manquants et de la mesure de similarité. En adoptant une vision non alarmiste des données manquantes (en considérant que ces données importent peu sur la similarité) et en étendant la mesure de similarité à chaque déclaration d'un nouveau descripteur lié au contexte, il est possible de tenir compte d'un contexte *a priori* illimité.

3.3 Mesure de similarité dépendante du contexte

Mignot [1997] a étudié le rôle du contexte dans le calcul de la mesure de similarité. Il intervient effectivement dans la mesure où deux cas peuvent être similaires dans un contexte et totalement différents dans un autre contexte. Par exemple (emprunté à Mignot) la Chine et la Corée du Sud sont similaire relativement à la « *localisation géographique* », mais dissemblables par rapport à la propriété « *forme de gouvernement* ».

C'est ce que nous avons également observé lors de la résolution d'incidents sur une ligne de métro, où deux incidents très différents sont résolus de la même manière en heure de pointe alors qu'en heure creuse, chacun d'eux sont résolus suivant des stratégies spécifiques. Par exemple, en cas d'incident traction ne pouvant pas être réparé sur place, mais autorisant la

rame à repartir sans voyageurs, quelque soit le type d'heure (de pointe ou creuse), la stratégie sera de partir « *haut le pied* ». Pour un incident de porte ne s'ouvrant plus, en heure creuse la stratégie sera de condamner la porte, alors qu'en heure de pointe, le train partira également « *haut le pied* ». Ces deux incidents, par nature très différents, ont des stratégies spécifique applicables en heure creuse, mais sont résolus de la même manière en heure de pointe.

3.4 Limites du modèle

Nous voyons deux faiblesses à ce mode de raisonnement. Premièrement la plus grande partie de l'expertise se situe dans la mesure de similarité, ce qui, d'une part, rend la mise au point de ce mode de raisonnement très délicate et, d'autre part, empêche toute explication du raisonnement suivi.

Deuxièmement, dans son idée fondatrice, le RàPC prévoit d'intégrer tous les nouveaux cas dans la base de cas. Dans notre domaine d'application, nous avons affaire à 2500 incidents par an sur la seule ligne 4, dont la plupart ne diffèrent que par des détails, autant sur le plan des descripteurs que sur les stratégies appliquées pour les résoudre.

L'adaptation du RàPC dans notre domaine d'application peut être réalisée en adoptant le point de vue de Mignot [1997] sur la relation entre le contexte et les mesures de similarité et de Bisson [1995] sur les mesures de similarité définies sur des objets. Toutefois ce mode de raisonnement empêche toute explication des résultats obtenus par le système et ajoute un côté obscur pouvant tourner au désavantage du système.

Section 4 :

Formalismes de représentation par modèles informatiques

Dans cette section, nous étudions différentes structures informatiques apparentées à notre approche. Dans un premier temps, nous analysons des modèles dont la structure est proche, mais qui ne peuvent pas être directement mis en application dans notre cas. C'est le cas notamment des diagrammes d'influence et des réseaux de Petri. Nous étudions ensuite des modèles dont la structure est très différente, mais qui sont plus proches de notre modèle. Nous nous intéressons ainsi aux diagrammes d'activités d'UML, à divers mécanismes de décomposition des tâches en sous-tâches et à la planification dynamique.

4.1 Modèles informatiques de structure similaire

4.1.1 Diagrammes d'influence

Les diagrammes d'influence [Oliver & Smith, 1990 ; Neapolitan, 1990] résultent de l'introduction d'actions dans les réseaux Bayésiens [Jensen, 1995 ; Jensen, 1996]. Ce sont ainsi des structures permettant de prendre des décisions en fonction de probabilités de réalisation d'événements et d'espérances d'utilités des décisions prises. Ils sont constitués de deux types de sommets. Les premiers sommets représentent des décisions. Ces sommets proposent plusieurs alternatives au décideur. Les seconds sommets sont des nœuds d'événement sur lesquels le décideur n'a aucune emprise. Ces nœuds événements sont associés à une table de probabilité d'occurrence de chaque réalisation possible de l'événement. Les sommets sont reliés par des arcs représentant la dépendance du second nœud par rapport au premier. Le graphe a une unique entrée.

Les graphes ainsi construits permettent à un décideur d'asseoir sa décision. En partant de l'unique entrée, le décideur parcourt le graphe et cherche à optimiser son espérance de gain en opérant des choix judicieux. Quand il croise un nœud événement il calcule l'espérance de gain par l'intégration des espérances des différentes possibilités qui s'offrent à lui par la suite ; quand il croise un nœud de décision, il suit la branche qui maximise son espérance de gain. Les diagrammes d'influence peuvent être utilisés pour compacter les arbres de décision.

Ce modèle de représentation de décision est issu des arbres de décision classiques et des réseaux Bayésiens. Il repose donc sur les probabilités d'occurrence des différents événements et sur un calcul d'utilité des différents choix possibles [Pearl, 1988]. La simplification des arbres correspondants en un graphe d'influence suit le même principe que celui que nous avons opéré pour le passage des arbres aux graphes dans notre modèle (voir les motivations dans le troisième chapitre, page 87). Il s'agit de rassembler les sous-structures communes à différentes branches. Cette simplification n'est toutefois possible que lorsque les variables sont statistiquement indépendantes [Howard, 1990], ce qui n'est pas toujours le cas, surtout dans les diagnostics d'incidents.

Notre modèle de graphes contextuels est issu de la même opération sur des arbres proches des arbres de décision classiques. Il est donc logique que nous obtenions une structure proche. Les différences entre les arbres de départ se retrouvent toutefois dans les graphes à l'arrivée. Ainsi nous n'utilisons aucune probabilité et ce pour deux raisons : premièrement nous ne pouvons pas avoir accès à des probabilités d'occurrence des événements, deuxièmement, le raisonnement des opérateurs n'est pas basé sur des probabilités mais plutôt sur des possibilités. Le but n'est pas de savoir quelle est la situation la plus probable, mais de déterminer quelles sont celles qui sont possible et idéalement de spécifier exactement celle-ci

pour prendre des mesures parfaitement adaptées. De plus, le but des opérateurs n'est pas d'optimiser une utilité en fonction d'événements possibles, mais de prendre les mesures nécessaires pour se ramener à une situation convenable en fonction de la situation réelle et de son évolution. La grande différence se situe sur le fait que les graphes d'influence sont des supports de décision en fonction d'événements possibles (donc pas encore réalisés ou connus) alors que les chefs de régulation basent leurs décisions sur ce qu'ils savent de la situation réelle.

4.1.2 Réseaux de Petri

Les réseaux de Petri sont des graphes bipartites composés d'états et de transitions [Brams, 1983]. Les sommets du graphe sont reliés par des arcs représentant soit des pré-conditions des transitions (quand ils vont d'un état à une transition), soit des post-conditions des transitions (quand ils vont d'une transition vers un état). Ils permettent ainsi de représenter les différents états d'un système et les transitions possibles d'un état à un autre. Les arcs peuvent être valués et permettre ainsi de représenter non plus de simples états du système mais également de prendre en compte des ressources disponibles.

Les réseaux de Petri peuvent être représentés par des matrices représentant les évolutions des états du système en fonction des transitions possibles. Le formalisme des réseaux de Petri repose donc sur des bases solides d'algèbre linéaire. Les réseaux de Petri sont toutefois clairement représentables et peuvent être utilisés directement sous forme de graphes. Les réseaux de Petri représentent explicitement les dépendances et indépendances d'ensembles d'événements au sein du système modélisé. Les événements dépendants sont liés à des états communs, alors que des événements indépendants n'ont pas d'état en commun.

Les réseaux de Petri ont un pouvoir représentatif supérieur aux diagrammes d'états et ont notamment l'avantage de pouvoir simuler les transitions des états du système et de tester son bon fonctionnement [Humphreys et Berkeley, 1992]. Si la simulation par le réseau de Petri révèle un dysfonctionnement du système, elle indique par la même occasion les transitions qui ne s'opèrent pas et permet donc de remonter aux causes du dysfonctionnement. La simulation, dans la version graphique des réseaux de Petri, repose sur le déplacement de jetons dans le réseau. Une transition est tirable si il y a au moins autant de jetons dans chaque état précédent la transition que le nombre portée par l'arc de pré-condition les reliant. Quand une transition est « tirable » (c'est-à-dire « peut être exécutée », voir Figure 4-2), son application élimine autant de jetons que précisé par les arcs de pré-condition dans chacun des états concernés et ajoute le nombre de jetons correspondant à la valeur portée par les arcs de post-condition dans les états qui le suivent.

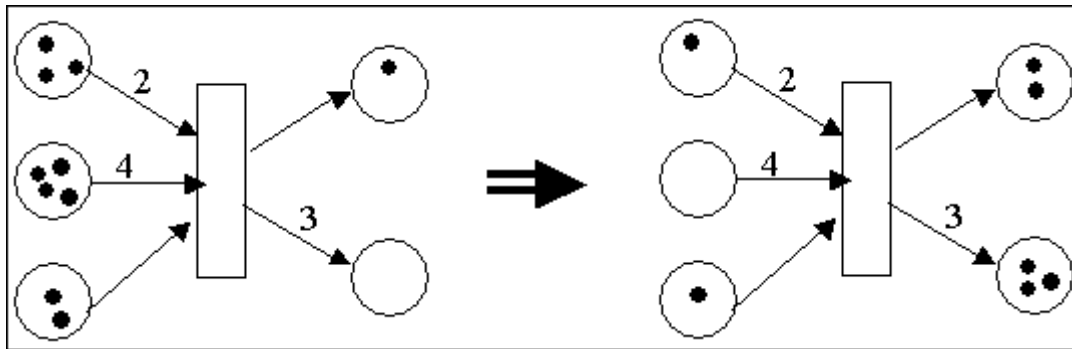


Figure 4-2: Tir d'une transition dans un réseau de Petri

Les réseaux de Petri permettent également une représentation à différents niveaux d'abstraction. Des transitions peuvent être détaillées par un réseau de Petri en dehors du réseau de Petri dont elle font partie. Les réseaux de Petri sont ainsi un outil puissant de modélisation et d'analyse de systèmes dynamiques à états discrets, basés sur les états du système ou de ses sous-systèmes et des transitions conditionnées par ces états.

Ce modèle peut être mis en parallèle avec notre approche : les transitions correspondent à des actions ou des activités de l'opérateur, et les états correspondent à l'état du processus régulé (la ligne dans son ensemble). Les éléments sur lesquels les opérateurs fondent leurs choix sont alors représentés par les états du système. L'affinement de transitions correspond au détail d'une activité. Malheureusement une telle modélisation nécessiterait une analyse complète du système (la ligne, les rames, les voyageurs...), des différents états possibles et des transitions admises. Vu la complexité d'une ligne de métro, ceci n'est pas réalisable. De plus, les pratiques des opérateurs évoluent rapidement et les réseaux de Petri devraient en tenir compte par l'adaptation de leurs structures à ces nouvelles pratiques.

4.2 Diagramme d'activité dans UML

Les diagrammes d'activité d'UML [OMG, 1999] sont également des extensions des diagrammes d'état traditionnels, dont le but est de représenter des activités partagées par plusieurs classificateurs. Leur rôle initial concerne plus les séquences et conditions d'exécution des actions que de déterminer quels classificateurs va réaliser ces actions. La plupart des états correspondent à des actions élémentaires et les transitions sont opérées par des événements qui sont : des événements de fin de réalisation d'une action précédente, des événements signalant la disponibilité d'un objet dans un état particulier, des signaux ou la satisfaction de certaines conditions.

Les diagrammes d'activité étendent notamment le modèle de diagrammes d'état par l'introduction de sous-activités qui correspondent à des actions complexes ayant une certaine

durée et dont l'exécution fait appel à des structurations d'actions et d'autres sous-activités. Les diagrammes d'activité conservent la structure des diagrammes d'état traditionnels et peuvent donc représenter des décisions grâce à l'utilisation de fourches, de jonctions et de synchronisations.

Les modèles obtenus par l'utilisation de diagrammes d'activité d'UML sont ainsi très proches de notre modèle à la fois par leur structure et par leur capacité représentative. La structure des diagrammes d'activités permet de représenter des successions de décisions et d'actions, les décisions portant sur des événements internes au système (signaux de fins d'actions et d'activités, signaux de disponibilité de ressources, autres signaux), soit sur la réalisation de conditions nécessaires à la réalisation des actions qui suivent. Nous avons une correspondance forte entre les quatre éléments constitutifs des diagrammes d'activité et de nos graphes contextuels que sont les actions, les activités, les fourches et les jonctions et respectivement les actions, les activités, les nœuds contextuels et les nœuds de recombinaison. De plus ces différents éléments constitutifs sont organisés de la même manière dans les deux cas. Une différence structurelle peut toutefois être soulignée : les graphes contextuels sont beaucoup plus riches en nœuds contextuels et nœuds de recombinaison que les diagrammes d'activité ne le sont en fourches et jonctions. Les deux modèles partagent également la capacité de représentation modulaire par sous-activités. UML ne précise pas le lien entre les activités et les sous-activités : ces dernières sont-elles indépendantes des premières et accessibles en dehors de celles-ci ou ne sont-elles que des simplifications graphiques ? Les synchronisations des diagrammes d'activité n'ont pas de correspondance dans les graphes contextuels. En effet, ces derniers ne représentent l'activité que d'une seule personne (dans notre application, le régulateur) et n'ont donc pas besoin de synchroniser les activités de divers acteurs : au pire certaines actions des graphes contextuels correspondent à l'attente d'informations ou de la réalisation d'une action par une tierce personne. Réciproquement, les branchements temporels ne sont pas représentés dans les diagrammes d'activités car un classificateur ne réalise pas plusieurs activités en parallèle. Des activités peuvent être réalisées en parallèle par différents classificateurs, mais pas par un seul.

Au niveau des capacités de représentation, nous retrouvons également des similitudes. Bien que les diagrammes d'activité concernent des activités de classificateurs, ce modèle pourrait être appliqué à des activités humaines. D'un point de vue statique, ceci est tout à fait faisable : les structures sont très proches (jusque dans les noms des éléments constitutifs). Toutefois dès que nous voulons utiliser les diagrammes d'activités sur une plus longue durée, nous retrouvons le problème de l'évolution des structures représentatives. UML ne prévoit pas l'évolution de ses diagrammes d'activité, car les systèmes informatiques ne modifient pas leurs propres structures de contrôle. Les structures des deux types de graphes étant très

proches, il doit sans doute être faisable de définir un algorithme permettant de faire évoluer les diagrammes d'activité, mais il ne correspondrait pas à l'esprit de ces diagrammes.

Les diagrammes d'activité d'UML sont ainsi proches des graphes contextuels à la fois d'un point de vue structurel et d'un point de vue représentatif. Toutefois les points de départ des deux représentations sont diamétralement opposés : les premiers représentent des activités de machines programmées et donc non-évolutifs et peu sensibles au contexte ; les seconds représentent des activités humaines en constante évolution et hautement sensibles au contexte.

4.3 Décomposition des tâches et tâches génériques

Les diagrammes d'activité d'UML permettent de représenter une certaine décomposition des tâches de différents classificateurs automatiques et leurs interactions. Ce diagramme sert principalement lors de la conception de systèmes automatiques complexes pour organiser les différents modules et leur communication. Nous abordons dans cette sous-section une décomposition des tâches à un niveau différent. Ce niveau concerne l'organisation interne de chaque programme.

La décomposition de tâches en sous-tâches est une opération classique en informatique et particulièrement en intelligence artificielle. Elle est également référencée sous le nom de paradigme tâche/méthodes, car elle consiste, pour chaque tâche, à identifier les différentes méthodes permettant de mener à bien cette tâche. Ces décompositions incluent également des références aux objets du domaine qui sont nécessaires pour l'application des différentes méthodes et permettent ainsi de gérer des moyens et les conséquences des réalisations de tâches sur ces objets.

Les tâches, dans cette acception, décrivent les différents buts à atteindre pour résoudre un problème donné. Par exemple, dans notre domaine d'application, la tâche principale des opérateurs est de *réguler le trafic et gérer les incidents*. Cette tâche définit les buts suivants : *conserver une certaine régularité entre les rames*, et en cas d'incident *identifier et éliminer les causes de l'incident*.

Les méthodes représentent comment atteindre les buts fixés par les tâches. Deux types de méthodes existent : premièrement certaines méthodes décrivent directement une marche à suivre pour mener à bien la tâche ; deuxièmement d'autres méthodes proposent des plans plus ou moins préétablis de tâches plus simples qui, une fois réalisées en respectant les contraintes imposées par le plan, aboutissent au résultat souhaité. Chaque tâche est associée à au moins une méthode et plusieurs méthodes peuvent être applicables pour mener à bien une même tâche.

Le principe de décomposition d'une tâche en sous-tâches repose sur le deuxième type de méthodes présenté ci-dessus. Quand une méthode propose de décomposer une tâche en sous-tâches, cette méthode identifie des étapes intermédiaires à atteindre. Chaque étape est associée à un but, lui-même faisant référence à une sous-tâche pour l'atteindre. Ce type de méthode peut alors être vu comme une succession de sous-tâches.

Quand plusieurs méthodes sont possibles pour la réalisation d'une même tâche, il faut préciser comment choisir la méthode. Traditionnellement chaque méthode est décrite indépendamment des autres et associée à des informations globales sur ses conditions de réalisation pour qu'une structure de contrôle du processus puisse choisir la méthode appropriée à la situation rencontrée. Ainsi chaque tâche est associée à une liste de méthodes, qui sont elles-mêmes soit des listes de sous-tâches, soit une description directe de la marche à suivre.

La représentation d'une tâche (et de ses décompositions éventuelles) peut ainsi être faite par des arbres ET/OU. Ces arbres sont composés de deux types de nœuds : les premiers (nœuds ET) indiquent que toutes les branches doivent être suivies pour atteindre le but souhaité, les seconds (nœuds OU) indiquent qu'une seule de leurs branches doit être suivie pour atteindre le but souhaité. Ainsi un nœud OU peut représenter une tâche, les différentes branches correspondent alors aux différentes méthodes possibles ; un nœud ET peut représenter une méthode, les différentes branches correspondent alors aux sous-tâches.

Chandrasekaran [1986] étudie la décomposition des tâches de haut niveau (diagnostic, supervision, ...) et définit les *tâches génériques*. Celles-ci sont des tâches d'utilité générale dont les méthodes sont simples. Les tâches génériques peuvent ensuite être utilisées par différentes méthodes de diverses tâches plus complexes. Elles servent ainsi de briques élémentaires à l'édifice des tâches. Ceci est une optique ascendante du paradigme tâches/méthodes. Tandis que la vision classique consiste à partir des tâches les plus hautes et de les décomposer en sous-tâches jusqu'à obtention d'un degré de détail opérationnel, la vision de Chandrasekaran propose de décomposer les tâches jusqu'aux tâches génériques.

Le paradigme tâches/méthodes, tel qu'expliqué ci-dessus et utilisé dans de nombreuses applications, est très intéressant, mais a des limites. Quand le nombre de méthodes possibles est conséquent, il devient important de définir plus précisément dans quelle condition appliquer telle ou telle méthode. De plus, quand les différentes méthodes partagent une grande part des sous-tâches qu'elles identifient, il est préférable de passer d'une structure arborescente à une structure de graphe. Cette opération est proposée par Chandrasekaran, Johnson et Smith [1992] et aboutit à des structures de tâches. Ces structures représentent les tâches et les différentes méthodes permettant de les réaliser. Les auteurs soulignent également que des connaissances sont nécessaires à quatre niveaux dans ces structures de tâches :

premièrement pour mener à bien chaque tâche ; deuxièmement pour indiquer quand une sous-tâche est nécessaire ; troisièmement, quand une méthode fait appel à plusieurs sous-tâches, pour ordonnancer celles-ci ; quatrièmement, quand plusieurs méthodes sont possibles, pour sélectionner une méthode. L'absence d'une connaissance indispensable à la réalisation d'une tâche (quelque soit le niveau auquel la connaissance intervient) peut être comblée par la réalisation d'une sous-tâche permettant de l'acquérir. Ainsi les connaissances permettant d'organiser la réalisation d'une tâche n'est pas un préalable à sa commande d'exécution : il reste possible de l'obtenir au moment de la réalisation même.

Les graphes contextuels peuvent être interprétés comme des structures organisant les différentes méthodes connues pour la réalisation d'une tâche. Chaque chemin autorisant la traversée d'un graphe contextuel correspond à une méthode possible. L'avantage des graphes contextuels par rapport à des structures ET/OU (arbres ou structures de tâches) est qu'ils organisent les différentes méthodes à deux niveaux. La première organisation des méthodes est l'ordonnancement des sous-tâches de chaque méthode. Cet ordonnancement est opéré par l'orientation des arcs indiquant une précédence temporelle. Deuxièmement les graphes contextuels contiennent les informations nécessaires au choix des méthodes à appliquer en fonction des conditions rencontrées. Ainsi les connaissances spécifiques à chaque tâche sur l'ordonnancement et sur les choix des méthodes sont intégrées dans la structure même des graphes contextuels. Le formalisme compact des graphes contextuels permet en outre de représenter un grand nombre de méthodes pour une même tâche. L'utilisation de graphes contextuels pour la décomposition de tâche simplifie la vision des différentes méthodes. Ils sont particulièrement intéressants quand ce nombre est grand. Dans notre domaine d'application, le nombre de méthodes connues pour chaque tâche est très variable, mais l'évolution des pratiques le fait croître expérience après expérience.

Le modèle des graphes contextuels distingue deux types de tâches : celles liées aux actions et celles liées aux activités. Les premières sont définies comme étant élémentaires du point de vue de l'opérateur et proposent une simple marche à suivre. En ce sens, elles peuvent être considérées comme associées à une unique méthode qui ne décompose pas cette tâche en sous-tâche. Elles sont identifiables aux tâches génériques de Chandrasekaran [1986]. Les secondes sont des tâches dont la réalisation met en œuvre une certaine planification et décompose la tâche en sous-tâches.

4.4 Planification et planification dynamique

La planification dynamique est une branche de recherche particulièrement active en intelligence artificielle, notamment en ce qui concerne les lois comportementales en robotique et les automates ou programmes devant évoluer dans des environnements dynamiques. Elle

consiste en la définition de plans à partir d'actions élémentaires en vue d'atteindre un but fixé. Chaque action est définie par des pré-conditions, une procédure de mise en œuvre et des résultats escomptés. La planification consiste alors à construire un plan d'action (une séquence ordonnée d'actions) pour atteindre le but en fonction de la connaissance que le système a sur son environnement et sur lui-même.

Dans STRIPS [Fikes et Nilsson, 1971], qui fait figure de référence parmi les planificateurs indépendant du domaine, chaque action **a** est représentée par un triplet d'ensemble de faits : **Pre(a)** est l'ensemble des pré-conditions, **Add(a)** et l'ensemble des faits à ajouter et **Del(a)** est l'ensemble des faits à supprimer. Nous avons, pour chaque action, la relation $\text{Del}(\mathbf{a}) \subseteq \text{Pre}(\mathbf{a})$, puisque tout fait éliminé par une action doit être vérifié avant l'exécution de l'action. Chaque état possible du système est alors représenté par un ensemble fini de faits. Une action est applicable dans un état du système si son ensemble de pré-conditions est inclus dans l'ensemble des faits décrivant cet état. L'application d'une action **a** modifie l'état du système de telle manière que le nouvel état est décrit par l'ensemble des faits décrivant l'état initial allégé des faits contenus dans **Del(a)** et additionné de ceux de **Add(a)**. Dans ce formalisme, un problème de planification est un triplet composé de l'ensemble des actions possibles, d'un état initial et d'un but. L'état initial et le but sont des états du système et sont donc deux ensembles de faits. La résolution de ce problème de planification consiste à trouver une séquence d'actions (un plan) permettant de passer de l'état initial à l'état but en vérifiant à chaque étape que l'état du système autorise la réalisation de l'action suivante. C'est une planification statique dans le sens où les états initial et final ne changent pas.

Deux grandes classes d'heuristiques interviennent alors : soit par chaînage avant, soit par chaînage arrière. Les heuristiques par chaînage avant partent de l'état initial et avancent vers l'état final ; Les heuristiques par chaînage arrière font le contraire. Les premières sont plus intuitives, mais leur efficacité est limitée par le fait que la distance entre un état et le but n'est pas connue. En effet, à chaque étape plusieurs actions peuvent être appliquées ce qui conduit à plusieurs états. La recherche se poursuit donc en suivant un arbre de recherche, mais la sélection de la direction de recherche ne peut pas être opérée sur des critères de proximité des états par rapport au but car, en cours de construction par chaînage avant, cette proximité n'est pas connue. Ainsi les planificateurs ne savent pas *a priori* dans quelle direction chercher. Le chaînage arrière permet de savoir dans quelle direction chercher, mais se heurte à des non-dits sur la situation finale. Par exemple dans un problème de logistique dont le but est d'établir un plan de transport de différents colis en différents lieux, le but est exprimé en termes de localisation finale des colis, mais pas des éléments de transport. Ainsi il est nécessaire d'étendre la description du but pour intégrer tous les non-dits.

L'approche de Refanidis et Vlahavas [2001] propose une heuristique intermédiaire. Dans une phase de pré-traitement, leur heuristique établit, par chaînage arrière, la distance entre chaque état accessible et le but. Une fois cette mesure effectuée, l'heuristique reprend un chaînage avant pour la construction du plan en étant guidé par la distance entre les états et le but. Le planificateur ainsi construit est très performant.

Une autre innovation consiste en la hiérarchisation de plans et la réutilisation de plans intermédiaires déjà construits [Kambhampati et Hendler, 1992 ; Britanik et Marefat, 1999], afin de ne pas refaire ce qui a déjà été fait. Veloso [1994 ; 1997] propose une intégration du raisonnement à partir de cas dans des planificateurs afin de rendre le planificateur plus efficace. Les cas enregistrés consistent en une trace de la planification réalisée : l'état initial et le but, le plan obtenu et les choix réalisés lors de la planification. Ces cas sont ensuite réutilisés pour des planifications ultérieures. Le planificateur ne se contente plus alors de réutiliser des plans intermédiaires, mais également des raisonnements ayant conduit à ces plans.

Quelle que soit la méthode utilisée, le planificateur peut proposer plusieurs plans alternatifs. Ceci est particulièrement intéressant dans des environnements dynamiques ou dans le cas où les actions peuvent échouer. Dans de tels cas il est en effet important de pouvoir réagir à toute modification de l'environnement et de proposer des planifications réactives [Kott, Saks et Mercer, 1999]. Travaillant sur la planification dynamique, Drummond [1989] propose d'établir des plans partiels et partiellement ordonnés. Ainsi les systèmes reposant sur ces plans sont plus réactifs à leur environnement. Rocha, Ramos et Vale [1998] construisent des plans à partir de graphes de précedence. Ces graphes intègrent les contraintes sur les différentes actions possibles en ordonnant partiellement les actions en fonction des contraintes sur leur réalisation. La particularité des plans ainsi obtenus est la définition de plans intermédiaires alternatifs. Les différentes alternatives permettent d'atteindre un même but, mais de manières différentes et en utilisant différentes ressources.

Nous ne faisons qu'esquisser ici le domaine de recherche lié à la planification dynamique, car il n'est que faiblement lié à notre approche. Toutefois il nous intéresse dans le sens où les futurs utilisateurs de notre système d'assistance opèrent une telle planification pour chaque incident. Ainsi la planification dynamique et les recherches dans ce domaine nous sont utiles pour appréhender les problèmes spécifiques rencontrés par les futurs utilisateurs. Les solutions apportées par ces recherches sont utiles pour intégrer un modèle de l'utilisateur dans notre système. De plus, les travaux sur la réutilisation de plans partiels [Hayes-Roth et Hayes-Roth, 1979] sont en concordance avec notre problématique : notre système modélise en effet ces différents plans partiels construits par les opérateurs. Les graphes contextuels peuvent ainsi être utilisés pour l'organisation des plans déjà établis pour la réalisation d'un but et

faciliter ainsi l'édification de nouveaux plans. De plus nous retrouvons également la notion de plans intermédiaires alternatifs qui sont au cœur de notre modèle.

Section 5 :

Conclusion

Dans le cadre de la modélisation de raisonnements à un niveau stratégique en vue de l'application à la gestion d'incident, nous nous sommes intéressés à divers modèles de structuration de la mémoire et de l'activité. Les Schémas, et leurs applications informatiques (Frames, Scripts et MOPs), sont des structures permettant de représenter une situation en mémoire. Cette structure est un support à la compréhension de la situation représentée. Les structures ainsi construites sont des relations entre différents objets qui sont instanciés en fonction de la situation, un objet pouvant être lui-même un Schéma plus simple. Ainsi un même Schéma peut être appliqué à la compréhension de deux situations proches. Le point fort des Schémas est leur structuration à deux niveaux : premièrement au niveau de chaque Schéma, deuxièmement au niveau de l'ensemble des Schémas.

Les Frames sont des implémentations de Schémas partiellement pré-instanciés représentant ainsi des situations stéréotypées. Les instanciations sont faibles et peuvent être adaptées à une situation particulière, rendant ainsi compte de spécificités. L'originalité des Frames réside dans leur organisation en systèmes de Frames, qui regroupent les Frames pouvant être appliqués à une même situation. Ces systèmes de Frames offrent ainsi différents points de vue sur la situation.

La deuxième implémentation des Schémas correspond aux Scripts. Ceux-ci contiennent de plus des actions attendues dans les situations représentées. Ils organisent ainsi des scènes avec leurs acteurs, les événements attendus et les faits devant être réalisés par chaque acteur. Ces structures permettent de comprendre des situations plus complexes et de diriger éventuellement l'activité d'un acteur. Les Scripts sont définis suivant le point de vue d'un des acteurs et un ensemble de Scripts permettent d'appréhender une situation à un niveau supérieur. Les MOPs sont une évolution des Scripts dissociant les scènes des Scripts afin de permettre un partage des scènes par plusieurs MOPs. Ils opèrent ainsi une certaine généralisation de scènes.

Tous les modèles issus des Schémas offrent les deux niveaux de structuration. Ces structures permettent non seulement la compréhension d'une situation particulière, mais aussi une vision plus globale des situations traitées. Ce point est capital car il permet de représenter le degré de généralité associé à chaque situation. Nous retenons cette structuration à différents

niveaux, que nous retrouvons également dans des modèles informatiques tels que la décomposition des tâches, pour notre application. Malheureusement les structures obtenues n'intègrent pas de mécanisme d'évolution en fonction des expériences passées et se révèlent ainsi difficilement applicables dans notre cas.

Les schèmes sont des structures cognitives proches des Schémas, mais qui sont axées sur l'évolution de leurs structures en fonction des expériences passées. Ainsi ils offrent plus de flexibilité quand à la compréhension d'une situation. Les schèmes d'action sont de schèmes qui organisent les activités. Ils comportent des invariants opératoires (qui sont en fait les composants des Schémas), des règles d'action, des inférences et des prédictions. Ils étendent ainsi la notion de Schéma en proposant de plus des guides permettant de construire un plan d'action en fonction de la compréhension que nous avons de la situation. Ils sont de plus fortement orientés vers l'évolution des leurs structures. Cette évolution opère à trois niveaux. Le premier niveau est l'adaptation du schème à la situation, permettant d'aménager le schème au moment même de son utilisation. C'est une sorte de boucle de retour rapide sur l'organisation de l'activité : quand un plan d'action est inefficace, la structure du schème ayant produit ce plan d'action est immédiatement modifiée pour proposer un nouveau plan. Le deuxième niveau est l'acquisition de nouvelles stratégies qui s'effectue après coup. Enfin, le troisième niveau est la différenciation, qui consiste à copier puis adapter le schème associé à une activité pour créer un schème pour une activité proche.

Les schèmes partagent une part de leur structure avec les Schémas (la part statique sur la compréhension de la situation). Les schèmes d'action proposent de plus une organisation des actions en vue de la réalisation d'une activité complexe en fonction de la situation. Ils entrent ainsi dans le cadre de notre étude sur les activités des chefs de régulation. Les différentes perspectives d'évolution qu'ils offrent sont également très prometteuses et permettent de rendre compte des évolutions des pratiques des opérateurs. Ainsi ce modèle cognitif nous intéresse particulièrement, mais malheureusement aucune implémentation informatique de ces structures n'est proposée actuellement.

Les raisonnements sensibles au contexte actuellement proposés sont peu nombreux. Dans le premier chapitre, nous avons présenté le système ORCA [Turner, 1989 ; Turner, 1995 ; Turner, 1998] (section 4.2, page 45), basé sur des schémas contextuels et le Context-Based Reasoning [Gonzalez et Ahlers, 1993 ; Gonzalez et Ahlers, 1995] (section 4.3, page 47), basé sur des Scripts et proposant trois niveaux de contexte délimitant respectivement la mission (ou tâche globale), la stratégie et la tactique. Dans ce chapitre, nous avons mis en avant un troisième raisonnement artificiel tenant compte du contexte : le Raisonnement à Partir de Cas, qui est un raisonnement de type analogie. Ce raisonnement a la particularité de prendre en compte le contexte de manière naturelle. De plus une adaptation de celui-ci, par la définition

de différentes mesures de similarité, est particulièrement sensible au contexte et permet de rendre compte de la différence de similarité des cas en fonction de la situation. Ce mode de raisonnement n'est pas en liaison directe avec notre recherche d'un modèle de raisonnement basé sur les activités tel que nous l'avons présenté dans le troisième chapitre, mais nous intéresse dans la perspective de l'implémentation d'un Système d'Assistance Intelligent en Contexte. En effet, comme souligné dans l'introduction de ce chapitre, l'utilisation de différents modèles de raisonnements permet d'accroître sensiblement l'efficacité de la représentation. C'est dans cette optique que nous intégrerons ce mode de raisonnement à côté du mode de raisonnement développé dans cette thèse, dans le système d'aide proposé (voir sixième chapitre).

D'autres modèles informatiques de raisonnements ont été proposés. Les diagrammes d'influence sont intéressants, mais sont basés sur des notions statistiques qui sont inaccessibles pour notre application. Le point de vue est également très différent : les diagrammes d'influence sont prévus pour établir une décision sur des faits à venir à partir des évolutions possibles de la situation, alors que nous cherchons à établir une décision sur des faits connus sur la situation. La différence se ressent sur les structures proposées par les deux modèles, bien qu'une parenté puisse-t-elle être établie sur l'obtention des modèles à partir des arbres de décision. Les réseaux de Petri sont également de structure proche des graphes contextuels que nous proposons. Leur fonctionnement impose toutefois une connaissance précise des différents états du système modélisé, ce qui nous obligerait à prendre en compte la ligne de métro dans son ensemble. De plus les structures ainsi construites ne peuvent pas évoluer, et ne peuvent donc pas prendre en compte les évolutions des pratiques des opérateurs.

Les diagrammes d'activité d'UML permettent de représenter le partage d'activité entre différents classificateurs. Ils ont une structure proche des graphes contextuels et partagent leurs principaux éléments constitutifs. Toutefois il n'est pas prévu d'évolution de ces structures et l'esprit de ces diagrammes les rend statiques. En faisant abstraction de l'aspect dynamique des activités de nos opérateurs, ces diagrammes seraient parfaitement adaptés. L'esprit est toutefois orthogonal et les structures, bien que très proches d'un point de vue théorique, sont en pratique l'une plus orientée vers les processus et l'autre vers la sélection d'un plan d'actions suite à une analyse très fine de la situation.

Le paradigme tâche/méthode est ainsi plus proche de notre application. Nous ne cherchons pas à avoir une seule possibilité, mais à organiser les différents plans d'actions possibles. Ce paradigme est lié à la décomposition des tâches en sous-tâches et traite de l'organisation des différentes méthodes possibles pour la réalisation d'une tâche. Actuellement rien n'est fait en ce qui concerne l'organisation des différentes méthodes entre elles : les tâches sont associées

à une liste de méthodes parmi lesquelles nous devons choisir au moment de l'application grâce à une description des conditions d'application des différentes méthodes. Les graphes contextuels offrent ainsi une structuration possible des différentes méthodes en structurant l'ensemble des méthodes en fonction de ces conditions d'application. Ils proposent ainsi une simplification de l'utilisation de ce paradigme en intégrant une grande part des connaissances nécessaires aux choix et aux ordonnancements des actions, grâce à leur structure.

Cette structure est identifiable à un plan d'actions organisé par les contraintes contextuelles sur la réalisation de celles-ci. Les graphes contextuels peuvent ainsi être également des structures guidant la planification dynamique des actions et des activités des opérateurs en fonction de l'évolution de la situation.

Cinquième chapitre :

Modélisation de raisonnements tenus en contexte

Les raisonnements humains sont fortement dépendants du contexte et sont réactifs à toute évolution de l'environnement. Cette capacité d'innovation fait défaut dans les systèmes automatiques. Le développement de SAIC est l'occasion de se pencher sur la modélisation des activités des opérateurs plutôt que sur celle de raisonnements artificiels productifs : un tel système ne cherche pas à prendre des décisions, mais à aider un opérateur à en prendre. Ainsi il n'est pas utile pour le SAIC d'être capable d'innovations, mais plutôt de comprendre l'opérateur et d'intégrer l'évolution de ses pratiques.

Les graphes contextuels, présentés dans le troisième chapitre, permettent une prise en compte très fine du contexte dans le modèle des raisonnements suivis par les opérateurs et sont associés à un algorithme permettant l'ajout de nouvelles pratiques. L'utilisation de tels graphes pour la représentation des activités des opérateurs propose un modèle implémentable et simple de structures cognitives organisant ces activités, tels que les schèmes d'action.

Nous présentons dans ce chapitre comment les graphes contextuels peuvent être utilisés pour modéliser les activités et les raisonnements suivis par les opérateurs, notamment les pratiques de gestion des incidents, mais aussi les procédures officielles. Nous présentons également comment l'algorithme d'acquisition incrémentale de connaissances des graphes contextuels permet de compléter un raisonnement par l'introduction de nouvelles pratiques. Nous étudions particulièrement les généralisations induites par cet apprentissage. Nous faisons ensuite un parallèle avec les schèmes d'action identifiés lors de l'analyse de l'activité des opérateurs. Enfin, nous résumons les apports de telles structures pour l'implémentation d'un SAIC.

Section 1 :

Raisonnements tenus en contexte

Une grande différence entre l'être humain et les machines est sa capacité à traiter des situations imprévues. L'intelligence humaine prend en compte de nombreux détails de la situation pour adapter ses raisonnements. Cette adaptation des raisonnements est en grande partie possible grâce à l'expérience que nous acquérons pas à pas, par essai-erreur et par reproduction de schémas antérieurs ayant porté leurs fruits. Ainsi les raisonnements que nous

produisons sont des adaptations incessantes de raisonnements déjà produits, ce qui permet, petit à petit, de construire un ensemble de raisonnements appliqués à différentes situations.

Les machines, et particulièrement les systèmes informatiques, ne sont pas capables de telles évolutions : les algorithmes n'intègrent pas leurs réussites (et échecs) passées. Certains raisonnements artificiels gardent une trace de leurs exécutions précédentes en vue d'adapter les raisonnements produits par la suite. Ceci est une première étape, mais n'est pas suffisant pour atteindre les capacités humaines. La manière dont ces expériences sont réutilisées est statique.

La décomposition des tâches est ainsi une solution proposée par l'intelligence artificielle pour représenter un certain degré de latitude dans les raisonnements automatisés. Toutefois, comme souligné dans le chapitre précédent, les décompositions actuellement réalisées ne proposent pas de prendre en compte la richesse des situations précises, et ne sont pas dotées de système d'évolution des méthodes proposées en fonction des expériences passées. Les tâches sont définies, dans ce cadre, comme étant des buts à atteindre, dont la réalisation se manifeste par une activité. Pour un système informatique, l'activité se traduit par l'exécution d'un algorithme prédéfini, alors que pour l'être humain l'activité est plus complexe et met en œuvre des raisonnements.

Nous revenons, dans cette section, sur les notions de tâches, d'activité et de raisonnement et étudions leurs liens et interactions. Nous précisons ensuite ce que nous entendons par raisonnement stratégique et par raisonnement en contexte et développons la notions de raisonnement stratégique en contexte. Enfin nous analysons les éléments à retenir en vue de la modélisation de telles activités pour une assistance informatique adaptée à la situation.

1.1 Tâches et activités

Les tâches, qu'elles soient définies pour un système automatique ou dans le cadre de l'analyse des activités humaines, sont assimilables à des buts à atteindre pour la résolution d'un problème donné. Selon Richard [1983] (comme discuté par Tijus [2001a]), les tâches sont caractérisées par un but, des contraintes et un domaine de connaissances. La ou les méthode(s) permettant d'atteindre le but souhaité (et donc de réaliser la tâche) n'est pas contenue dans la tâche.

Pour réaliser une tâche, un acteur (personne ou programme) doit entrer en activité [Leplat et Hoc, 1983]. L'activité a des composantes observables et d'autres inobservables [Tijus, 2001a]. Les premières correspondent aux actions mises en œuvre pour la réalisation de la tâche et aux verbalisations parfois faites par l'actant. Les composantes inobservables de l'activité correspondent à l'activité interne, qu'elle soit consciente ou non. L'activité regroupe

ainsi les actions mises en œuvre, leur organisation suivant un plan d'action et les raisonnements appliqués pour obtenir ce plan d'action en fonction de la situation pour la réalisation de la tâche souhaitée. L'activité englobe divers raisonnements pour organiser les actions et opérer des choix stratégiques tels que ceux liés à des solutions alternatives.

La confusion entre les notions de tâche et d'activité résulte du fait que, dans le cadre d'applications artificielles traditionnelles (automates, informatique...), nous observons une bijection tâche/activité. Par exemple, dans le cadre d'applications informatiques classiques, les tâches sont directement associées à une méthode décrite par un algorithme et mise en œuvre par un programme. Ce programme peut être mis en application, ce qui conduit à une activité de l'ordinateur.

Avec le développement de l'informatique, nous observons également la multiplication des applications possibles pour la réalisation d'une même tâche. Ainsi plusieurs programmes ont les mêmes fonctionnalités. De plus, certaines tâches nécessitent une chaîne d'applications (par exemple pour le montage vidéo, il faut acquérir les différentes séquences, les numériser, établir le plan de montage puis combiner les séquences). La solution proposée consiste en la décomposition d'une tâche en sous-tâches et de la définition de différentes méthodes pour la réalisation de chacune des tâches (voir quatrième chapitre). Le modèle obtenu est ainsi plus proche, structurellement, des activités humaines, mais pas d'un point de vue fonctionnel.

Les activités humaines sont beaucoup plus complexes. Elles intègrent de nombreux éléments extérieurs à la tâche pour établir un plan d'actions le mieux adapté. Toutefois ce plan est dynamique, dans le sens où à chaque instant il peut être remis en cause par l'occurrence d'un événement inattendu. Cette réactivité est liée à notre expérience et à notre gestion de plans alternatifs en parallèle du plan mis en œuvre. Plus nous avons eu à faire à une situation, mieux nous la traitons. Les plans précédents, qu'ils aient réussi ou échoué, forment un guide à notre activité courante. Les erreurs sont ainsi formatrices car elles sont enregistrées avec leur contexte d'occurrence et quand une situation similaire se présente à nouveau, nous faisons beaucoup plus attention. Nos activités s'enrichissent ainsi, expérience après expérience, des plans ayant réussi et des éléments sensibles pouvant mener à un échec. Chaque enrichissement affine l'activité par rapport aux contextes d'application et amène son lot de connaissances contextuelles associées à cette activité.

La différence entre les activités humaines et des activités des automates réside dans cette capacité d'intégrer les expériences et leur contexte pour augmenter notre réactivité.

La décomposition des tâches en sous-tâches opérée pour des systèmes informatiques s'applique aussi aux activités humaines. Le modèle de schèmes d'action proposé par Piaget en 1936 pour l'analyse de l'apprentissage chez les enfants (voir le quatrième chapitre, section 2)

fait également référence à un tel mécanisme. Les activités humaines peuvent faire référence à des activités secondaires (par exemple *conduire une voiture* fait référence à *regarder la route et les autres usagers, maintenir une vitesse adaptée, calculer et maintenir une trajectoire...*). L'intégration du contexte dans chaque activité développe, toutefois, considérablement le nombre de méthodes pour chacune d'elles. L'organisation de l'activité humaine est ainsi fortement associée au contexte.

1.2 Raisonnements stratégiques contextuels

Pour chaque activité, la planification des actions est établie par l'application de divers raisonnements de type stratégique. Les raisonnements stratégiques sont des calculs sur les buts finaux et intermédiaires ainsi que sur l'ordonnancement d'actions et d'activités secondaires en vue de la réalisation de la tâche associée. Le résultat obtenu est un ensemble de méthodes possibles pour passer d'un état initial à un état final, chaque méthode correspondant à une séquence ordonnée d'actions.

Dans le paradigme tâche/méthode proposé par l'intelligence artificielle, les raisonnements stratégiques sont réduits. Chaque méthode est déjà construite (le plan est pré-établi) et la sélection de la méthode s'effectue suivant des critères donnés à l'avance (la sélection est fortement dirigée par les conditions d'application de chaque méthode). Ainsi les étapes de diagnostic et celles de planification des actions sont fixées lors de la conception du système. La pré-définition des méthodes et des conditions d'application de celles-ci est la cause de la faible acuité contextuelle de la décomposition obtenue.

Au contraire, les activités humaines sont fortement liées à la situation. Dans de nombreux cas, nous modifions nos plans en fonction du contexte. Les raisonnements stratégiques que nous appliquons alors ne se contentent pas de choisir une méthode parmi les méthodes conservées en parallèle les unes des autres, mais s'appuient sur une généralisation et une structuration des différentes méthodes en fonction du contexte. Sauf dans le cas de problèmes pour lesquels aucune innovation n'est nécessaire, les raisonnements par analogie s'appuient souvent sur plusieurs cas précédemment rencontrés pour concevoir une solution originale en prévoyant les événements spécifiques de la situation. Les raisonnements plus structurés, quant à eux, produisent régulièrement des solutions différentes des solutions précédemment proposées, grâce à une certaine généralité des règles mises en œuvre.

L'expérience joue un rôle particulièrement important dans nos capacités. A chaque expérience productive (c'est-à-dire ayant produit des connaissances soit par une innovation réussie, soit par un échec) nos raisonnements sont modifiés. La fréquente répétition de

situations proches permet ainsi petit à petit d'améliorer nos raisonnements et de les enrichir en contexte.

C'est ce que nous avons observé notamment pour les chefs de régulation des lignes du métro parisien. Les incidents les plus courants, quand ils surviennent dans des situations classiques, sont gérés suivant des principes bien établis. Les incidents plus rares ou dans des situations exceptionnelles sont gérés cas par cas et amènent, souvent, une adaptation des principes jusqu'alors utilisés : le processus de construction des raisonnements stratégiques est encore en cours pour ces incidents. Les raisonnements appliqués évoluent en fonction des expériences vécues et des expériences relatées, c'est à dire de l'expérience de l'opérateur. L'opérateur inexpérimenté ne peut s'appuyer que sur ses connaissances théoriques, qui sont très techniques et liées aux procédures. Incident après incident, il forge son expertise en intégrant de plus en plus de détails contextuels adaptant ses raisonnements stratégiques. L'expertise se constitue ainsi par la modification des schémas conceptuels et leur adaptation graduelle au contexte. Dans notre cas, cette expertise se traduit par des pratiques spécifiques, qui sont des instanciations des plans construits par leurs raisonnements. Ces pratiques permettent de remonter à la stratégie (le plan prévu) qui lui est porteur des adaptations des raisonnements au contexte. C'est cette stratégie qu'il est important de mettre en évidence et d'intégrer dans les modèles de raisonnement stratégiques contextuels.

1.3 Conséquences pour la conception d'un SAIC

Un SAIC (voir premier chapitre, section 5) est un système à base de connaissances dont le but n'est pas d'anticiper les innovations de l'opérateur, mais de savoir en tenir compte pour la compréhension de ses activités en vue d'une aide efficace. Un système classique de décomposition des tâches permet ainsi de structurer les activités de l'opérateur, et de comprendre une activité en cours de réalisation. Toutefois un tel système ne suffit pas, car la décomposition est peu sensible à la situation précise et les innovations de l'opérateur ne sont pas intégrées.

L'innovation étant la clé de l'expertise, un système d'aide se doit d'être capable de prendre en compte ces innovations, même (et surtout) à un niveau stratégique. Ceci conduit à une multiplication des méthodes associées à chaque tâche. Il devient alors nécessaire d'organiser toutes ces méthodes et donc l'activité qui résulte de cette tâche (l'activité est, rappelons le, la conceptualisation de la planification des actions en vue de la réalisation de la tâche). Ainsi chaque expérience productive modifie les activités des opérateurs.

Un SAIC doit être doté d'un modèle de représentation des activités autorisant la prise en considération des expériences, afin d'intégrer les innovations mises en oeuvre par les

opérateurs pour une réutilisation lors de situations semblables. Un tel système construirait alors un modèle de l'expertise des opérateurs et de son évolution expérience après expérience.

Section 2 :

Modélisation d'activités par les graphes contextuels

Ce que nous proposons ici est de baser la modélisation des activités sur les graphes contextuels, que nous avons présenté dans le troisième chapitre (page 87). Ceux-ci ont en effet une structure permettant l'organisation de différentes méthodes pour une tâche donnée et sont dotés des capacités d'évolution nécessaires à l'intégration des innovations proposées par l'opérateur. Nous présentons, dans cette section, comment utiliser les graphes contextuels pour représenter une activité (nous détaillons ici les idées esquissées dans le troisième chapitre, section 4, page 107). Puis nous étudions comment le modèle obtenu structure les activités. Enfin nous faisons un point sur une adaptation du modèle de graphe contextuel pour la mise en œuvre pratique du modèle d'activité.

2.1 Organisation d'une activité par un graphe contextuel

L'association d'un graphe contextuel à une tâche permet de représenter une organisation de l'activité mise en œuvre pour réaliser cette tâche : chaque structure ainsi formée contient des informations sur la tâche à réaliser (nom, but, pré-conditions, post-conditions, ressources...) et sur les différentes manières de les réaliser (les méthodes). Nous obtenons ainsi un système original de décomposition classique de tâches en sous-tâches (voir le quatrième chapitre, section 4.3, page 133), qui prend en compte la richesse de la situation pour encadrer l'activité et qui autorise une prise en compte des évolutions des méthodes au cours du temps, représentant ainsi l'évolution des pratiques.

Lors de l'implantation d'un système basé sur les concepts de tâches et sous-tâches, les méthodes sont habituellement choisies *a priori* et fixées une fois pour toutes. Dans le meilleur des cas, les méthodes possibles sont organisées dans des arbres ET/OU, associant à chaque nœud OU des éléments permettant de choisir parmi les différentes options. L'utilisation de graphes contextuels pour organiser les différentes méthodes permettant la réalisation d'une tâche facilite le choix d'une méthode appropriée. La richesse du contexte pouvant être pris en compte augmente le nombre de méthodes pouvant être représentées. La Figure 5-1 représente ainsi l'exemple que nous avons suivi dans le troisième chapitre sous la forme d'arbre ET/OU. Nous retrouvons les tâches et les différentes méthodes proposées : chaque tâche est associée à une ou plusieurs méthodes (branchement OU, représenté par de simples flèches) ; chaque

méthode étant constituée d'une ou plusieurs actions ou sous-tâches devant toutes être effectuées (branchement ET, représenté par des flèches reliées par un arc).

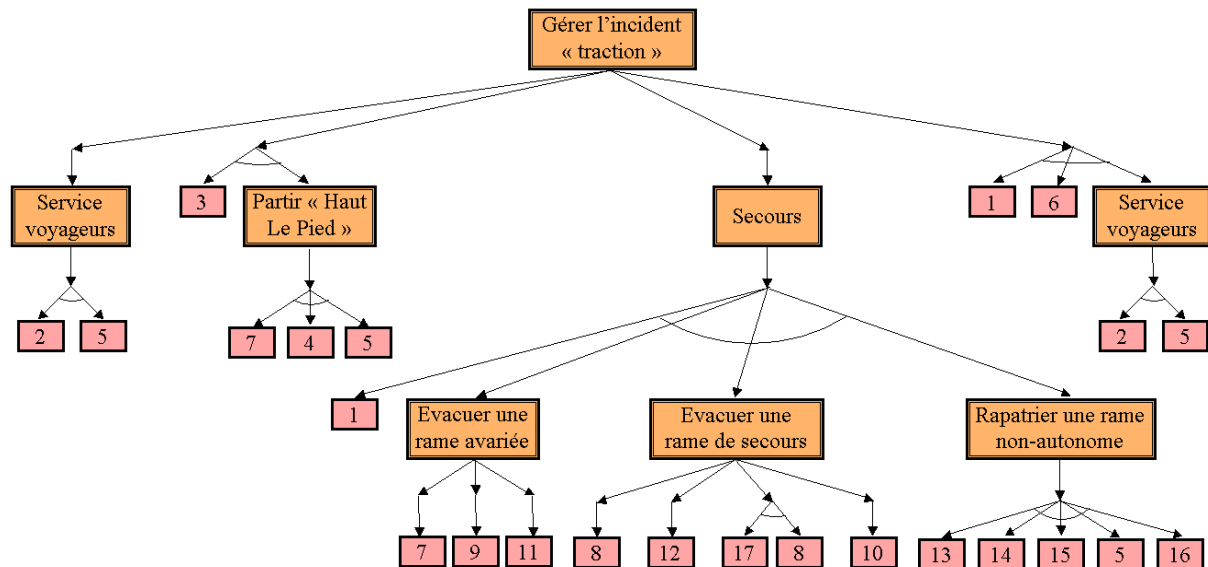


Figure 5-1: Décomposition des tâches liées à la gestion d'un incident traction suivant le formalisme de représentation des arbres ET/OU

Il ne faut pas oublier que l'exemple que nous présentons n'est que la procédure de gestion d'un incident de type traction et que le contexte n'intervient que très peu dans cet exemple. Ainsi, même si cet exemple semble manipulable, les pratiques réelles des opérateurs ajoutent de nouvelles méthodes pour chaque tâche ce qui accroît rapidement la structure. De plus, les structures de tâches (type arbre ET/OU ou celles de Chandrasekaran, Johnson et Smith [1992]) sont complexes et pas assez intuitives pour être directement employées par des utilisateurs peu réceptifs à ce genre de modèles. Elles ne représentent pas la dimension temporelle, qui est le principal guide de nos opérateurs. Les graphes contextuels sont orientés par la précédence temporelle et permettent aux opérateurs de connaître immédiatement l'ordonnancement des actions à entreprendre pour une tâche donnée en fonction de la situation précise. Les éléments contextuels du graphe dirigent également la phase de diagnostic en mettant en avant les éléments sur lesquels la décision est établie et en orientant donc les recherches d'éléments de contexte supplémentaires pour des situations incomplètement identifiées.

L'originalité de notre approche réside dans l'organisation des différentes méthodes dans un graphe contextuel. Celui-ci permet une sélection dynamique des méthodes en fonction de la situation réelle. Les graphes contextuels sont axés sur la succession temporelle des actions et sous-tâches à réaliser, tout en laissant possible la prise en compte de séquences parallèles. Chaque action ou activité modifie le contexte global de la tâche pour laquelle elle a été réalisée, par ses conséquences et les ressources disponibles, et influe donc sur les décisions

suivantes. L'aspect dynamique de la sélection de la méthode adoptée pour la réalisation est ainsi mise en avant, ainsi que la prise en compte de l'évolution du contexte de résolution du problème.

La structure obtenue représente ainsi le diagnostic et les actions intimement liés, ce qui permet une réactivité à l'évolution de la situation en cours de traitement.

La structure des graphes contextuels fait également apparaître les points communs des différentes méthodes possibles pour la réalisation d'une tâche.

2.2 Structuration des activités

Comme précisé dans le troisième chapitre (section 4), l'utilisation de graphes contextuels pour la représentation et l'organisation des activités permet de structurer à la fois les activités elles-mêmes, mais aussi les activités entre-elles.

Pour les opérateurs, chaque activité se trouve ainsi représentée dans un formalisme simple montrant les liens entre les différentes méthodes et leurs contextes d'application respectifs. Les stratégies sont clairement établies et leurs domaines d'application sont bien délimités. La structuration de chaque activité est importante pour représenter à la fois les procédures officielles et les différentes pratiques dérivées de celles-ci. La structuration au sein d'un même graphe contextuel fait ressortir les liens entre les procédures et les pratiques, ainsi que les adaptations de ces dernières au contexte de l'incident et à son évolution au cours de la gestion de celui-ci. L'expertise des opérateurs est ainsi représentée explicitement par la structure du graphe contextuel.

La structuration de l'ensemble des activités est un cadre simplifiant la compréhension des activités des opérateurs. Chaque activité est parfaitement définie (par la tâche correspondante) et son champ d'action est spécifié. L'opérateur est capable, sauf pour quelques rares situations exceptionnelles, de savoir si une action spécifique entre dans le cadre d'une activité ou d'une autre. Ceci est particulièrement important pour l'intégration d'une pratique dans le graphe contextuel d'une activité quand celle-ci fait appel à une ou plusieurs activités secondaires. La structuration de l'ensemble des activités permet également de tenir compte de l'expertise des opérateurs en analysant quelles activités sont parfaitement connues de celui-ci ou non. Les activités sont organisées dans un réseau de dépendance (voir Figure 3-11, page 109). Ce réseau permet d'identifier différents niveaux d'activité.

Dans notre application, le premier niveau correspond à la gestion d'événements incidentels. Ce sont les seuls points d'entrée dans le réseau. Ces activités sont peu nombreuses (10 dans le cas du PCC de la ligne 4, voir le deuxième chapitre, section 2.4.2, page 72) et sont

donc régulièrement observées. Ces activités organisent la première phase de diagnostic/action dans l'incertain (voir le deuxième chapitre, section 2.4.3, page 74) et font référence aux activités de traitement d'un incident dès que la cause de celui-ci est connue (actuellement SAGIM distingue 61 causes d'incident et organise autant d'activités, qui sont plus ou moins couramment mises en application). Ce deuxième niveau d'activité est plus technique et fait référence à des activités secondaires correspondant à des sous-tâches définissant des buts intermédiaires. Ensuite, plus nous avançons dans le réseau, plus les activités sont partagées par des activités de niveau supérieur. Ainsi elles sont plus souvent exploitées et établissent rapidement une structuration des méthodes connues. Les activités les plus étrennées correspondent ainsi aux activités courantes de plus bas niveau et aux activités de gestion des événements incidentels. Plus l'expertise de l'opérateur est grande, moins celui-ci a besoin d'être guidé dans les activités courantes. Ceci est également un point important pour la mise en œuvre du SAIC.

2.3 Adaptation du formalisme des graphes contextuels

Dans la définition des graphes contextuels donnée dans le troisième chapitre (section 3, page 98), nous ne limitons la structure des graphes qu'à des graphes orientés acycliques ayant une seule source et un seul puits. Du point de vue de la représentation théorique des activités, ceci est suffisant, mais certains graphes ainsi obtenus pourraient ne pas être planaires. La Figure 5-2 montre le graphe que nous devrions obtenir en théorie pour l'activité *évacuation d'une rame de secours*, en appliquant à la lettre les simplifications de l'arbre de la Figure 3-1 (page 90) proposées dans le troisième chapitre.

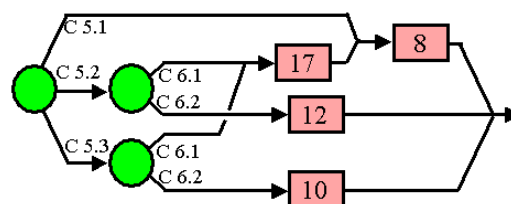


Figure 5-2: Graphe contextuel non-planaire de l'activité *évacuation d'une rame de secours*.

Ceci n'est pas gênant pour l'application du modèle dans un système informatique, mais la présentation visuelle de tels graphes à l'utilisateur peut introduire des erreurs de compréhension des graphes dues aux croisements des arcs. Le but de ce modèle étant de représenter de manière lisible par des experts du domaine (et donc pas forcément experts en informatique ou en raisonnements abstraits), nous limitons la structure des graphes contextuels à des graphes *planaires*.

Ceci implique une modification des graphes non-planaires obtenus. Cette opération peut être réalisée de différentes manières. La première, et la plus simple d'un point de vue

technique, consiste en la multiplication de parties de graphe et l'inversion des branches de certains nœuds. Le but recherché est alors de rapprocher les chemins qui partagent une grande part de leurs structures de diagnostic/action et, quand le rapprochement est impossible, de relâcher la contrainte de compacité du graphe en autorisant la copie des zones ne pouvant pas être mises en commun pour supprimer les croisements (visuels) restants. La Figure 5–3 montre ce qui peut ainsi être obtenu à partir de la figure précédente en appliquant le principe décrit ci-dessus : les branches du nœud contextuel **C6** placé sur la branche **C5.2** sont inversées pour permettre le partage de la séquence d'actions **17-8** entre les deux chemins **C5.2/C6.1** et **C5.3/C6.1** et l'action **8** est copiée sur le chemin **C5.1**.

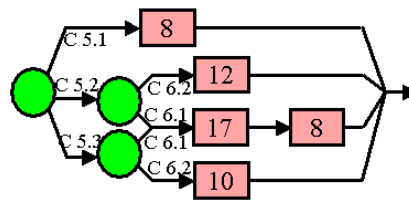


Figure 5–3: Graphe planaire obtenu par échange de branches des nœuds contextuels et copie de parties du graphe

La seconde méthode implique une intervention d'experts pour modifier les connaissances contextuelles impliquées dans le diagnostic. Dans l'exemple présenté ici, la connaissance contextuelle **C5** correspond à la **position de la rame par rapport à la station** et peut prendre trois valeurs : **à quai (C5.1)**, **partiellement à quai (C5.2)** ou **en interstation (C5.3)** ; la connaissance contextuelle **C6** correspond à **l'existence d'une station libre devant la rame** et peut prendre deux valeurs : **oui (C6.1)** ou **non (C6.2)**. Un expert peut alors proposer l'utilisation de trois connaissances contextuelles pour établir la stratégie plutôt que les deux précédentes, en remplaçant la connaissance contextuelle **C5** par les connaissances contextuelles **C5'** (la rame est-elle à quai : oui / non) et **C5''** (la rame est-elle partiellement à quai : oui / non). Nous obtenons ainsi une structure de diagnostic/action différente (Figure 5–4), représentant le même raisonnement que précédemment et les mêmes solutions appliquées, mais en basant les décisions sur des connaissances légèrement différentes.

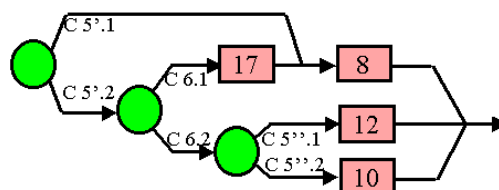


Figure 5–4: Graphe planaire obtenu par la modification par un expert des connaissances contextuelles impliquées dans le diagnostic

Cette seconde solution peut être assistée par le système d'aide qui peut détecter les connaissances contextuelles introduisant une non-planarité. Tout un panel de méthodes est ainsi possible entre ces deux extrêmes, laissant plus ou moins de travail à l'expert. La seconde méthode est plus lourde à mettre en œuvre car elle implique un travail de réflexion important sur les raisonnements et s'apparente ainsi plus au travail des éditeurs de procédures qu'à celui des opérateurs visés. La première méthode est préférable, bien qu'elle réduise légèrement la généralité du graphe contextuel obtenu. En effet, les chemins qui se trouvent ainsi séparés ne pourront plus mettre en commun les modifications de leurs structures par l'acquisition à venir de nouvelles pratiques. La perte de généralité du modèle est toutefois très faible. Par contre l'illisibilité est un facteur limitant pouvant faire rejeter le système par les opérateurs concernés. Par la suite, nous considérons que la planarité des graphes contextuels utilisés pour la représentation des activités des opérateurs est une propriété intrinsèque liée au modèle, qu'il faudra garantir lors de l'évolution des graphes contextuels.

Section 3 :

Acquisition de connaissances stratégiques

Comme nous l'avons souligné à maintes reprises dans le cours de cette thèse, le principal objectif est la modélisation de raisonnements stratégiques contextuels hautement évolutifs en fonction des expériences. Nous proposons d'utiliser les graphes contextuels pour représenter les différentes méthodes connues pour la réalisation d'une tâche. Le modèle ainsi construit est un support aux raisonnements stratégiques des opérateurs concernés. Dans le troisième chapitre (section 3.4), nous proposons un algorithme d'acquisition incrémentale de connaissances pour les graphes contextuels. Cet algorithme permet d'introduire une nouvelle séquence dans un graphe contextuel et de la positionner par rapport aux autres séquences déjà contenues dans le graphe en fonction des éléments contextuels supportant les décisions. Nous analysons, dans cette section, l'application de cet algorithme à la modélisation d'activités et de l'évolution de celles-ci. Dans un premier temps, nous rappelons le fonctionnement de l'algorithme et présentons les adaptations causées par le choix de restriction du modèle à des graphes planaires. Nous analysons ensuite le rôle de l'acquisition incrémentale de connaissances stratégiques dans notre modèle d'activités. Enfin nous faisons un point sur la généralisation opérée par un tel apprentissage et le comparons aux systèmes d'apprentissage automatiques classiques d'intelligence artificielle.

3.1 Algorithme

L'algorithme proposé dans le troisième chapitre (sections 3.3, page 100 et 4.3, page 110) consiste en l'appariement des actions et des activités communes à la pratique à introduire et à la stratégie attendue par le système dans une telle situation. Dans le cadre de l'étude de l'introduction d'une définition circulaire entre les activités et les graphes contextuels (voir le troisième chapitre, section 4), nous préconisons l'application de l'algorithme à chaque activité plutôt qu'à l'ensemble des activités mises en œuvre par l'opérateur. Ceci permet en effet à l'opérateur de bien saisir, dès le début, quelles actions correspondent à quelles activités et de mieux structurer l'acquisition de l'expertise. Ce choix simplifie également l'algorithme en décomposant l'acquisition par étape bien définie par les activités.

Il convient également de maintenir la planarité de chaque graphe contextuel modifié. Même si cette propriété n'est pas théoriquement utile, elle l'est en pratique pour des raisons de simplicité de compréhension des graphes présentés. Des entorses à la planarité sont possibles lors de l'acquisition de nouvelles pratiques. La Figure 5-5 montre l'acquisition d'une pratique conduisant à une violation de la planarité du graphe contextuel, suite à l'application brute de l'algorithme d'acquisition incrémentale de connaissances (tel qu'il est décrit dans le troisième chapitre, section 3.3, page 100).

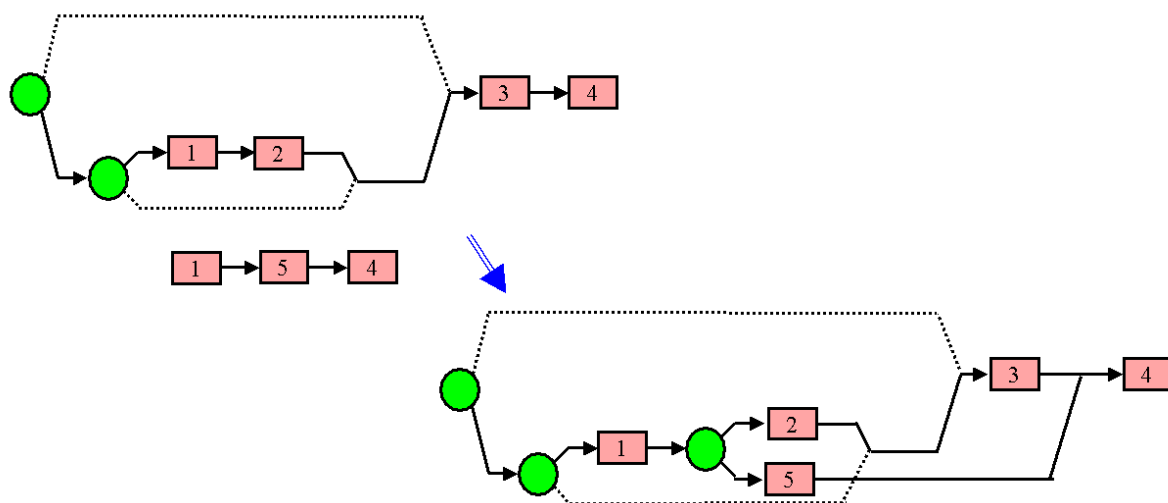


Figure 5-5: Non-planarité lors de l'acquisition d'une pratique

Des algorithmes, tels celui de Fary, proposent l'inversion de nœuds dans de telles conditions afin de rendre le graphe de nouveau planaire. Dans l'exemple ci-dessus, l'inversion du deuxième nœud contextuel permet de retrouver un graphe planaire (Figure 5-6). Ces algorithmes sont coûteux (proportionnels au carré du nombre de nœuds du graphe).

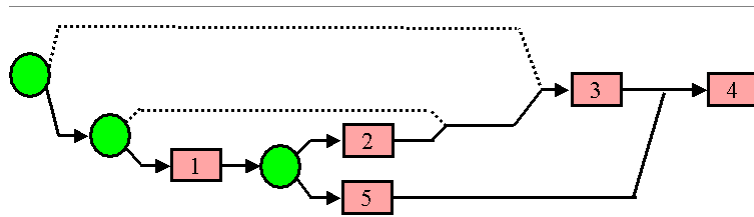


Figure 5-6: Graphe rendu planaire par inversion de noeuds contextuels

L'algorithme d'acquisition incrémentale de connaissances proposé est simple et permet d'ajouter facilement des pratiques à un ensemble d'activités concernées. Deux points cruciaux sont à retenir pour l'acquisition de l'expertise des opérateurs et son intégration dans les structures d'activité : premièrement l'identification des éléments de contexte ayant induit les adaptations proposées par la nouvelle pratique, et deuxièmement la localisation des frontières entre les composantes de la pratique correspondant aux différentes activités lorsque des actions ont été ajoutées dans ces zones frontières. Sur ce dernier point la décomposition préalable de la pratique est une aide précieuse.

3.2 Rôle de l'acquisition incrémentale de connaissances stratégiques

L'acquisition incrémentale de connaissances est un moyen simple de maintenir les connaissances à jour dans un système opérationnel fonctionnant dans un milieu à évolution rapide. L'acquisition incrémentale de connaissances ne modifie pas le type de connaissances intégrées, ni leur organisation, mais autorise seulement l'ajout de connaissances en suivant un modèle prédéfini. Dans le cas des graphes contextuels, le modèle permet d'ajouter des connaissances de niveau élevé (stratégiques) et de les insérer parmi les connaissances déjà acquises en tenant compte du contexte dans lesquelles ces connaissances peuvent s'appliquer.

Les principaux avantages de l'algorithme d'acquisition incrémentale de connaissances stratégiques proposé ici sont en relation directe avec l'interactivité nécessaire entre l'utilisateur et le système lors de cette étape. En effet cette interaction a trois grandes conséquences. Premièrement, cette phase d'acquisition de connaissances guide l'utilisateur dans une introspection de ses activités et est ainsi un excellent outil de formation continue. L'opérateur, avec l'aide du système, doit décomposer la pratique qu'il vient d'appliquer en les composantes correspondant à chaque activité. Le système joue un rôle important dans cette étape en présentant à l'opérateur les activités et les stratégies attendues pour chacune d'elles, fournissant ainsi un guide pour la décomposition de la pratique. Ensuite, pour chaque activité, l'insertion de la composante de la pratique fait ressortir les éléments du contexte ayant supporté les décisions de l'opérateur. Finalement la structure présentée suite à l'acquisition de

la pratique montre à l'opérateur comment celle-ci s'intègre parmi les autres pratiques et la procédure officielle.

Deuxièmement, l'interaction permet de mettre en évidence les explications en même temps que les connaissances elles-mêmes. Quand un élément contextuel est ajouté, l'opérateur peut préciser les raisons pour lesquelles cet élément du contexte intervient dans la planification. Il peut ainsi expliquer son propre raisonnement, ce qui permettra, ensuite, de justifier les stratégies proposées par le système pour un autre incident et à un autre opérateur et de guider les opérateurs dans les saisies des pratiques à venir.

Troisièmement, le système acquiert par ce biais des connaissances structurées compréhensibles par l'utilisateur (puisque c'est lui qui les communique au système). Ces deux derniers points sont particulièrement importants pour le développement de SAIC : un tel système doit être capable d'expliquer les raisonnements appliqués. Les explications sont ainsi possibles à deux niveaux : premièrement, la planification et les choix conduisant cette planification sont explicables par la structure de l'activité organisée par le graphe contextuel correspondant ; deuxièmement chaque choix peut être expliqué par le commentaire de l'opérateur l'ayant introduit.

L'acquisition incrémentale de connaissances stratégiques est un point très important quand on le replace dans le contexte d'utilisation de ce modèle. En effet, à la RATP, les procédures ont été établies à partir des données techniques sur le réseau mais également des expériences passées. Elles sont en évolution constante, bien que le processus de spécification de ces procédures soit très lent. L'acquisition incrémentale des pratiques réelles permet d'organiser celles-ci en identifiant les éléments du contexte à l'origine de ces nouvelles pratiques. La structure des graphes contextuels et leur organisation hiérarchique est proche de celle des procédures, ainsi les graphes obtenus suite à l'acquisition incrémentale de stratégies est un support idéal pour l'évolution des procédures. Il suffit de vérifier la validité des pratiques introduites et de les officialiser.

3.3 Généralisations opérées

L'acquisition des connaissances dans les graphes contextuels, avec l'aide de la construction dynamique du plan d'action, permet d'obtenir, dans une certaine mesure, une généralisation des stratégies. Cette généralisation joue à deux niveaux : celui des tâches et sous-tâches et celui des plans de diagnostic/action (définis par les activités). Au niveau des tâches et sous-tâches, une nouvelle méthode acquise suite à une nouvelle pratique proposée dans un certain contexte peut être de nouveau appliquée pour atteindre ce même but intermédiaire dans un contexte légèrement différent lors de la réalisation d'une tâche

différente. Par exemple (voir la structure de l'ensemble des activités, Figure 3–11, page 109, pour un support), si une nouvelle méthode a été proposée (et intégrée dans le graphe contextuel) pour la tâche *Evacuer une rame à quai* dans le cadre de la tâche *Evacuer une rame de secours* au cours de l'application d'une procédure de *Secours* lors de la *Gestion d'un incident « freinage »*, cette nouvelle méthode peut être appliquée également pour l'application de la procédure *Partir « Haut Le Pied »* dans le cadre de la *Gestion d'un incident « traction »*.

Au niveau du plan d'action, la généralisation est visible dans le cas où des connaissances ont été ajoutées dans un graphe contextuel après une convergence de stratégies suite à une nouvelle pratique copiant partiellement l'une des stratégies concernées par la convergence. Ainsi ces nouvelles connaissances sont accessibles également lors de la construction d'un plan d'actions calqué sur l'une des autres stratégies concernées par la convergence. Dans l'exemple donné Figure 3–3 (page 98), suite à l'apprentissage d'une pratique *2-18* dans le contexte *C1.2 ; C2.1 ; C3.1 ; C7.2* dans le graphe, nous obtenons le graphe de la Figure 5–7. Dès lors, la pratique *1-6-2-18* est une solution possible dans le contexte *C1.1 ; C6.2*. Cette stratégie n'est issue ni d'une procédure, ni d'une pratique, mais simplement obtenue par la généralisation de la modification induite par la pratique précédemment apprise.

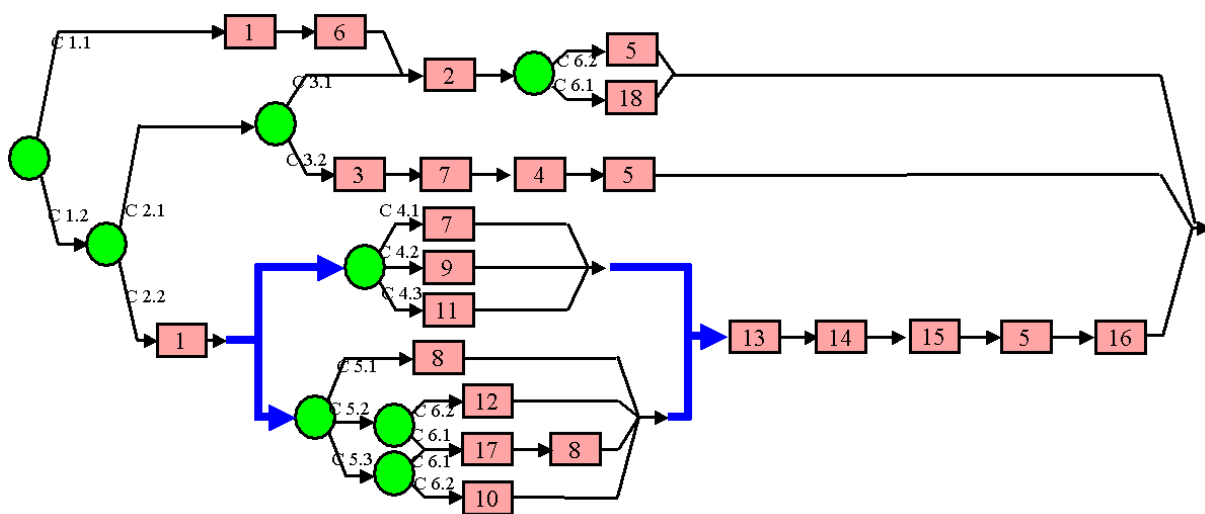


Figure 5–7: Généralisation au niveau de l'élaboration du plan d'actions

L'acquisition incrémentale de connaissances stratégiques introduit ainsi deux niveaux de généralisation des pratiques. Le premier niveau correspond à une généralisation d'une pratique déjà éprouvée dans un contexte différent, le second généralise une modification stratégique à une pratique différente. Ces deux types de généralisation des pratiques sont complémentaires et nous intéressent particulièrement car ils sont à la base de l'édition des procédures à partir des pratiques réelles des opérateurs.

3.4 différences avec l'apprentissage automatique

L'apprentissage automatique « classique » est basé sur des généralisations. Toutefois ces généralisations concernent des concepts et leurs relations. Elles permettent une structuration de la mémoire par un réseau de généralisation/spécialisation des connaissances et des concepts et modifie ces structures. L'introduction de nouvelles connaissances consiste alors à positionner ces connaissances dans le réseau construit et, éventuellement, d'ajouter des concepts supplémentaires permettant une généralisation des connaissances intégrées et de connaissances déjà apprises. Ainsi l'introduction de connaissances dans le modèle courant modifie le modèle. L'utilisation d'arbres pour la classification (type ID3) permet d'apprendre par réorganisation des connaissances et de faire émerger des concepts intermédiaires à partir des concepts connus. Cette classification est basée sur l'identification des propriétés générales partagées par les concepts apparentés et permet d'organiser les connaissances dans une structure montrant leur proximité.

Au contraire, l'acquisition incrémentale de connaissances ne joue que sur le contenu du modèle et pas sur le modèle lui-même. Dans le cas de notre modélisation des activités basée sur des graphes contextuels, seule la structure des graphes concernés est modifiée, et non pas le principe de l'utilisation de ces graphes. La généralisation que nous obtenons ne modifie pas le modèle mais est liée au formalisme des graphes contextuels et à leur structure. Elle n'est qu'un effet de bord de l'acquisition de nouvelles pratiques. Ainsi l'acquisition incrémentale n'est pas construite autour de la généralisation. Seuls certains modèles de connaissances présentent un certain degré de généralisation lors d'acquisition incrémentales de connaissances. Dans notre cas, la généralisation obtenue ne concerne pas les connaissances et les raisonnements eux-mêmes, mais uniquement leurs conditions d'application. En effet, que la généralisation opère sur une nouvelle méthode pour une sous-tâche ou sur un raisonnement local commun à plusieurs plans de diagnostic/action, les raisonnements et les connaissances contextuelles impliquées ajoutés lors de l'acquisition d'une pratique sont applicables tels quels dans les autres situations (appel à la même sous-tâche à partir d'une tâche différente ou application d'un raisonnement local commun dans un autre plan de diagnostic/action) sans qu'une seule opération n'ait été nécessaire pour rendre cette généralisation explicite.

Une autre différence capitale avec l'apprentissage automatique est que le mécanisme d'acquisition incrémentale de connaissances que nous proposons a l'utilisateur pour principal moteur. Les connaissances ne sont pas organisées et stockées par le système de manière automatique, mais grâce à une interaction avec l'utilisateur. Le système guide l'utilisateur dans la classification de ses pratiques puis organise celles-ci dans un ensemble structuré. Le rôle premier de cette acquisition de connaissances est d'accumuler les pratiques, puis de détecter les points communs entre elles. Cette détection est effectivement liée à

l'apprentissage, puisqu'elle cherche à classifier les pratiques et à faire ressortir les connaissances contextuelles permettant de les trier. Cet apprentissage n'est toutefois pas automatique puisque les connaissances exhibées ne sont pas extraites par le système, mais par l'utilisateur.

Section 4 :

Graphes Contextuels et schèmes d'action

4.1 Modélisation d'activités et schèmes d'action

Résumons la situation : le modèle de graphe contextuel permet de structurer l'activité des opérateurs au niveau des connaissances stratégiques. La définition de ce modèle est récursive dans le sens où certains graphes contextuels font appel, dans leur structure, à des activités, qui contiennent elles-même d'autres graphes contextuels. Ainsi la structuration d'activités par les graphes contextuels permet une structuration à la fois des activités elles-mêmes, mais aussi des activités entre elles. Les graphes contextuels sont en outre capables d'évoluer par l'intégration de nouvelles pratiques. Ces caractéristiques rappellent celles des schèmes d'action, structures cognitives organisant l'activité, comme nous les avons présentés dans le quatrième chapitre (section 2.5, page 122). Etudions de plus près les relations entre ces deux notions.

Les structures décrites ci-dessus, associant les graphes contextuels à un mécanisme de décomposition des tâches offrent un cadre de représentation des activités, de leur organisation hiérarchique et de l'élaboration dynamique d'un plan d'actions en fonction de la situation. Elles sont de plus capables d'évolution en fonction des expériences rencontrées par les opérateurs. Ces structures ont des propriétés pouvant être rapprochées des caractéristiques intrinsèques des schèmes d'action décrites dans le quatrième chapitre (section 2.5, page 122). Le Tableau 5-1 présente un parallèle entre les principales caractéristiques des schèmes d'action et les structures proposées.

Schèmes d'actions	Modélisation d'activités en utilisant les graphes contextuels
<u>Composition :</u> <ul style="list-style-type: none"> - But à atteindre - Structure des actions - Moyens nécessaires 	<u>Composition :</u> <ul style="list-style-type: none"> - But associé à la tâche - Graphe contextuel - Ressources
Appel d'autres schèmes pour atteindre des buts intermédiaires	Décomposition des tâches en sous-tâches et délégation à des activités secondaires.

Schèmes d'actions	Modélisation d'activités en utilisant les graphes contextuels
Prise en compte de la situation pour établir un plan d'action (accommodation)	Stratégie construite à partir du contexte
<u>Evolution :</u> <ul style="list-style-type: none"> - Adaptation - Addition de nouvelles stratégies - Différentiation 	<u>Evolution :</u> <ul style="list-style-type: none"> - Acquisition incrémentale de stratégies - Différentiation possible

Tableau 5-1: Comparatif des schèmes d'action et de structures à base de graphes contextuels

Les schèmes, structures mentales organisatrices de l'activité, partagent leurs principales caractéristiques avec la structure obtenue par l'utilisation de graphes contextuels pour représenter et organiser les différentes méthodes actuellement connues pour la réalisation d'une tâche. Ces deux structures sont orientées principalement vers l'évolution pour permettre de tirer le meilleur des expériences passées lors de la gestion de situations futures. Pour cela elles sont composées d'éléments permettant la construction dynamique de plans d'actions en fonction de la situation rencontrée (buts, ressources, structure guidant l'action) et sont douées de capacité d'évolution pour intégrer les solutions originales (acquisition incrémentales de stratégies) ou de création de nouvelles structures pour la réalisation de tâches légèrement différentes en adaptant une structure connue (différentiation). Enfin ces deux types de structures partagent également leur capacité à déléguer à des structures spécialisées pour l'accomplissement de buts intermédiaires.

4.2 Pouvoir représentatif du modèle

Ce qui est intéressant ici est le parallèle pouvant être tracé entre les schèmes d'action, qui sont des structures cognitives, et l'organisation d'activités par les graphes contextuels qui forme un modèle directement implémentable dans un système informatique. Dans le deuxième chapitre, nous avons précisé que des schèmes d'actions ont été identifiés lors de l'analyse de l'activité des chefs de régulation. Dans le quatrième chapitre, nous avons souligné que ces schèmes d'action n'ont pas encore été implémentés dans un système informatique. Ainsi notre proposition de modélisation d'activités par les graphes contextuels est une première tentative de l'implémentation de ces structures cognitives guidant l'activité des opérateurs.

La représentation, dans un système informatique, des activités de l'utilisateur par le biais de graphes contextuels donne au système un pouvoir déductif sur les activités et actions attendues de la part de l'utilisateur en fonction de la situation. En effet, connaissant l'activité principale de l'utilisateur, le contexte et les actions réalisées, le système est capable de savoir

où l'utilisateur en est et de prédire quelles actions vont être réalisées. Cette prédiction est soit validée par les actions réalisées par l'utilisateur (ceci signifie que l'utilisateur n'introduit pas de nouveauté dans cette partie du plan de diagnostic/action), soit invalidée (ce qui est caractéristique d'une nouvelle pratique). Dans ce dernier cas, l'acquisition incrémentale de connaissances stratégiques complète le plan de diagnostic/action concerné. Le système est ainsi capable de simuler l'activité de l'utilisateur et de réagir aux nouvelles pratiques de celui-ci. La représentation des activités par le biais de graphes contextuels est donc un modèle des activités pouvant être mis en œuvre dans un système informatique.

Notre modèle n'est toutefois pas construit en vue d'un fonctionnement autonome. Il se contente de représenter les activités des opérateurs et l'organisation de celles-ci en schèmes d'action, qui eux sont réellement impliqués dans les décisions prises. Un système utilisant le modèle d'activité que nous avons décrit ne crée pas de nouvelles stratégies (mis à part, par effet de bord, les généralisations discutées dans les deux sous-sections précédentes) mais organise, compile et structure l'ensemble des innovations proposées par les opérateurs. L'invention de nouvelles pratiques reste encore l'apanage des êtres humains. Le système proposé est par contre une avancée importante dans le cadre de systèmes d'aide adaptant leurs connaissances à celles de leurs utilisateurs, comme nous le discutons dans la section suivante.

Section 5 :

Graphes contextuels et développement de SAIC

Le développement d'un SAIC est fortement calqué sur celui des SBC traditionnels. Il s'en distingue principalement par l'intervention du contexte à tous les niveaux de connaissances impliqués : des objets du domaine et de leurs interactions, jusqu'aux niveaux stratégiques de construction du raisonnement et de la génération d'explications. Nous allons maintenant développer les conséquences de l'utilisation de graphes contextuels pour la modélisation des raisonnements stratégiques mis en œuvre dans un SAIC. Nous analyserons d'abord ces conséquences pour les connaissances générales

5.1 Connaissances générales

En ce qui concerne la définition des objets du domaine et des règles d'interaction entre ceux-ci, les graphes contextuels ne changent rien. Les méthodes classiques sont bien adaptées à ces modélisations, puisque ces éléments de connaissances n'évoluent pas ou (malheureusement) très lentement. Toutefois, il ne faut pas perdre de vue que le système doit

être orienté vers la notion de contexte : contexte d'utilisation normale des infrastructures, contexte de défaillance...

Le premier niveau d'intervention des graphes contextuels se situe lors de la spécification des actions élémentaires pouvant être réalisées par les différents acteurs. En effet, comme souligné précédemment, la limite entre les actions élémentaires et les activités (qui sont détaillées et associées à un graphe contextuel) est très ténue. Non seulement chaque action élémentaire pourrait être détaillée et rejoindre ainsi l'ensemble des activités, mais aussi certains objets du domaine pourrait être utilisés pour la réalisation d'actions par des méthodes inattendues et modifier ainsi la nature des actions impliquées. Nous ne rentrerons pas dans les détails de ce genre (qui soit dit en passant risqueraient de coûter cher à l'actant non-précautionneux). Il faut retenir de ceci que la liste des actions, qui sont des connaissances générales, peut évoluer au cours de l'utilisation du système et qu'il est donc important d'en tenir compte dès le départ.

5.2 Connaissances stratégiques

Les connaissances stratégiques sont celles qui organisent le raisonnement au niveau stratégique. Elles sont directement impliquées par l'utilisation de graphes contextuels pour le développement de SAIC. Ces connaissances sont de deux types : les procédures et les pratiques. La différence entre ces deux types de connaissances stratégiques se situe au même niveau que la différence entre le prescrit et le réel, ou que la logique de fonctionnement et la logique d'efficacité [De Terssac, 1992].

Les procédures sont des formes compilées de connaissances stratégiques établies à partir des éléments techniques régissant le domaine et de leurs interactions. Le formalisme employé pour la description des procédures mêle un diagnostic et des actions. Le diagnostic est dirigé par des questions sur la situation. Les procédures sont décrites par fiches, chaque fiche étant consacrée à un but précis. Une fiche peut faire appel à une suivante et ainsi de suite. La structure des graphes contextuels est issue de celle des procédures ; la représentation de celles-ci par des graphes contextuels est très aisée. Les procédures se distinguent à ce moment là des activités correspondantes par la faible attention portée sur les éléments du contexte. Les éléments supportant la décision font référence surtout à des éléments techniques sur la situation.

Les pratiques sont plus complexes à appréhender. Celles-ci correspondent aux applications réelles des procédures et peuvent en dévier légèrement pour tenir compte de particularités de la situation rencontrée. Une pratique se présente sous la forme d'une séquence d'actions réalisées dans un contexte donné pour atteindre un but précis. Elles sont à la base du principe

d'évolution des activités. Chacune d'elles peut être intégrée dans le graphe contextuel associé au but réalisé par celle-ci et être réutilisable dans une situation similaire à venir.

Les graphes contextuels permettent de modéliser les procédures de façon naturelle, mais sont également capables de prendre en considération les nouvelles pratiques des opérateurs. Les connaissances stratégiques peuvent ainsi évoluer au sein du SAIC : le SAIC peut ainsi être initialisé avec une description purement procédurale des activités, et utilisation après utilisation ses structures d'activité s'enrichissent des pratiques mises en œuvre par les opérateurs. Cette évolution est un point capital de ce genre de systèmes, car elle permet au système de rester à l'écoute des activités réelles de l'opérateur qu'il doit aider. Elle s'inscrit pleinement dans l'idée des SAIC car elle est réalisée grâce à une interaction constante avec l'utilisateur. Les connaissances acquises ne sont pas natives chez l'utilisateur mais issues de l'interaction constructive entre l'utilisateur et le système.

De plus, les connaissances ainsi acquises sont immédiatement disponibles pour l'établissement de nouveaux plans d'actions, que ce soit pour une tâche identique ou pour une tâche différente faisant appel à des sous-tâches communes ayant évolué. En effet, la généralisation de méthodes induite à deux niveaux par les graphes contextuels peut être exploitée pour la généralisation de stratégies par le système.

5.3 Génération d'explication

Un dernier point important dans l'esprit des SAIC est la génération d'explications par le système. L'utilisation de graphes contextuels dans un tel système pour la modélisation des connaissances stratégiques introduit des modifications dans la génération d'explication concernant les choix stratégiques et l'établissement des plans d'action.

L'aspect le plus important des graphes contextuels mis en jeu lors des explications est sans doute la hiérarchisation des tâches impliquées. Cette hiérarchisation permet d'organiser les explications pouvant être fournies par le système et de les adapter au niveau d'expertise de l'opérateur. Ainsi, lors de la présentation d'une stratégie particulière ou d'un graphe contextuel dans son ensemble, le système peut commencer par le plus haut niveau (ou un niveau défini par l'utilisateur lui-même) sans détailler les sous-tâches impliquées, supposant celles-ci connues de l'utilisateur. Si tel n'est pas le cas, l'utilisateur peut demander le développement des graphes contextuels de certaines sous-tâches. Les graphes conceptuels de Sowa [1984 ; 1992] offrent un mécanisme similaire de développement « à la carte ». Dans un premier temps les concepts principaux et leurs relations sont présentés à l'utilisateur. Ce dernier peut développer ensuite le graphe conceptuel autour des notions qui l'intéressent. Le

mécanisme de développement des tâches tel que nous le proposons est proche de celui des concepts proposé par Sowa.

Dans notre exemple récurrent, l'utilisateur peut par exemple demander le développement des actions composées *secours* puis *évacuer le train avarié* et *évacuer le train de secours* (Figure 5–8). Il peut également développer, si besoin est, les actions composées restantes (*service voyageur* et *partir haut le pied* dans *gestion de l'incident « traction »*, *rapatrier un train non-autonome* dans *secours* et *évacuer un train à quai* dans *évacuer le train de secours*).

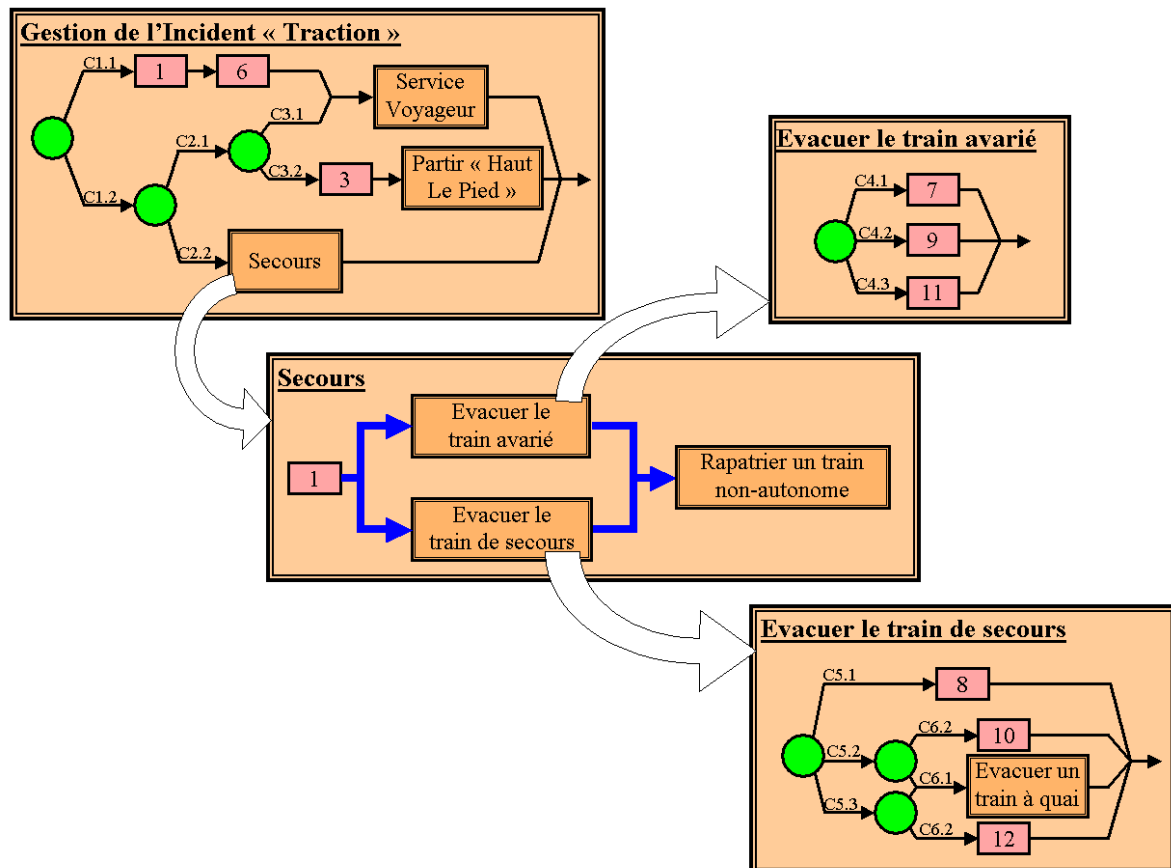


Figure 5–8: Différents niveaux d'explications possibles

On remarque ici la lisibilité particulière des graphes contextuels. Ce mécanisme de présentation des modèles de raisonnements est en effet très rapidement maîtrisé par les opérateurs. Ceci tient sans doute au fait que le modèle est proche du mode de raisonnement des opérateurs, qu'il est axé sur l'aspect temporel de la planification des actions et qu'il met en évidence les éléments du contexte permettant de construire ces plans d'actions.

Un deuxième niveau d'explications a déjà été mis en avant dans la section 3.2 : le cas échéant, l'utilisateur peut demander au système d'expliquer pourquoi un élément contextuel particulier implique telle ou telle décision. Le système peut alors répondre grâce aux explications fournies par les opérateurs lors de la saisie des pratiques. Ce mécanisme

d'explications permet de faire circuler, au sein de la communauté des utilisateurs, des éléments de raisonnements produits par un opérateur et de faire ainsi progresser l'expertise de chacun d'entre eux.

Section 6 :

Conclusion

Les graphes contextuels peuvent représenter les différentes méthodes possibles pour la réalisation d'une tâche, dont ils organisent les différentes méthodes autour de la notion de contexte qui permet de choisir quelle méthode appliquer en fonction de la situation courante. L'association d'un graphe contextuel à une tâche et l'utilisation des structures ainsi obtenues dans le cadre d'un mécanisme de décomposition des tâches permet d'obtenir un ensemble organisé de structures représentant les diverses activités concernées.

Chaque graphe contextuel précise les différentes méthodes actuellement connues pour la réalisation de l'activité correspondante. Les méthodes sont organisées dans une structure montrant l'ordonnement des actions et des activités secondaires, mais aussi (et surtout) les éléments contextuels sur lesquels se basent les choix stratégiques. Les raisonnements des opérateurs sont ainsi stockés dans la structure même des graphes contextuels.

Les graphes contextuels ont été spécifiés en vue d'intégrer les évolutions futures des méthodes possibles pour la réalisation d'une tâche. Ainsi ces structures sont particulièrement intéressante pour la modélisation d'activités fortement évolutives d'opérateurs. L'algorithme d'acquisition incrémentale de connaissances associé aux graphes contextuels est au cœur du système. Le système est capable de faire évoluer ses connaissances sur les pratiques des opérateurs.

Les nouvelles pratiques sont immédiatement disponibles pour la génération de stratégies. La structure des graphes contextuels et l'organisation hiérarchique de ceux-ci permet d'obtenir une généralisation de stratégies à deux niveaux. Premièrement, une nouvelle méthode intégrée pour la réalisation d'une tâche peut modifier les autres méthodes de réalisation de cette même tâche dès qu'elles partagent une partie du graphe. Deuxièmement, quand une nouvelle pratique introduit une nouvelle méthode pour la réalisation d'une sous-tâche dans le cadre d'une tâche plus importante, cette méthode peut être appliquée pour atteindre ce même sous-but dans le cadre d'une tâche principale différente.

L'association des graphes contextuels à un mécanisme de décomposition classique des tâches permet d'obtenir un modèle de représentation des activités proche de ce qui peut être

observé en situation réelle. Les structures obtenues permettent en effet de représenter les schèmes d'actions identifiés par l'ergonomie cognitive dans diverses situations de travail. Ces représentations sont aussi bien traitées par un système informatique qu'elles sont comprises par les opérateurs. Cet avantage est non-négligeable quand on cherche à développer un Système d'Aide Intelligent en Contexte.

Ce modèle permet de représenter les diverses connaissances de niveau stratégiques impliquées dans des activités complexes telles que la gestion d'incidents sur une ligne de métro. La structure des graphes contextuels est inspirée de celle des procédures d'abord et déjà en place, ce qui facilite grandement la représentation de celles-ci par un tel modèle. Les graphes contextuels ont été également conçus pour intégrer l'évolution des pratiques en les combinant aux procédures et pratiques déjà identifiées en explicitant les éléments du contexte permettant de sélectionner parmi toutes ces stratégies possibles. Le pouvoir de généralisation à deux niveaux est également un élément important quant à la généralisation des pratiques et donc de leur uniformisation.

La représentation des différentes méthodes de réalisation des tâches et sous-tâches par des graphes contextuels donne une dimension supplémentaire aux explications pouvant être fournies par le système à l'utilisateur. Les connaissances de niveau stratégiques peuvent ainsi être plus ou moins détaillées en fonction du niveau d'expertise de l'utilisateur. L'acquisition de connaissances stratégiques de la part d'un opérateur permet aussi au système de conserver une trace des raisonnements suivis par l'opérateur et peut ensuite utiliser ces connaissances pour expliquer ses choix futurs à d'autres utilisateurs.

Le modèle proposé, associant un graphe contextuel à chaque tâche en vue de représenter les activités, permet de bien représenter à la fois les procédures et les pratiques. Il permet d'accumuler et d'organiser les nouvelles pratiques en identifiant les différences du contexte dans lesquelles celles-ci s'appliquent. En intégrant un tel système dans le cycle de production des procédures à partir des pratiques facilite grandement la procéduralisation des pratiques grâce à l'identification du contexte procéduralisé.

La représentation des activités à l'aide de graphes contextuels permet de proposer une vision claire des activités et des deux niveaux organisationnels relatifs à ces activités. Ce modèle est non seulement très performant d'un point de vue conceptuel, mais également à la portée d'un grand nombre d'utilisateurs. La représentation en activités et leur organisation par des graphes contextuels est immédiatement assimilée par les opérateurs concernés. Cette assimilation rapide provient du fait que cette décomposition est calquée sur celle des procédures, dont l'aspect séquentiel montre bien la succession des diagnostics et des actions. Ce modèle permet également de prendre en compte de nombreux éléments de la situation, notamment des connaissances contextuelles et est évolutif. Son pouvoir représentatif et son

assimilation aisée sont des atouts quant à l'appropriation par les futurs utilisateurs d'un outil d'aide basé sur ce modèle.

Sixième chapitre :

Systeme d'Aide à la Gestion d'Incidents dans le Métro

La gestion des incidents sur une ligne de métro est une tâche difficile, fortement dépendante du contexte de l'incident et de son évolution. Dans le cadre du projet SART (Système d'Aide à la Régulation du Trafic), nous proposons d'aider les opérateurs dans leur activité par la conception et la réalisation d'un Système d'Aide à la Gestion d'Incidents dans le Métro (SAGIM). Ce système, conçu comme un SAIC, est un système hybride basé sur le modèle des *graphes contextuels* et sur un *raisonnement à base de cas*.

Dans ce chapitre, nous présentons le projet SART : son histoire, ses fonctionnalités et sa structure logicielle. Nous détaillons plus précisément l'agent d'aide à la gestion d'incident de SART, appelé SAGIM, qui est le cadre d'application des idées développées dans cette thèse, notamment le modèle de graphes contextuels. Nous présentons également les spécificités de notre application par rapport aux différents raisonnements implémentés, ainsi que les choix d'implémentation réalisés. Nous terminons par une analyse de l'utilisation du prototype dans des conditions réelles d'exploitation du métro.

Section 1 :

Le projet SART

Le projet SART (acronyme de Système d'Aide à la Régulation du Trafic) est un projet franco-brésilien¹⁴ dont le but est la conception et la réalisation d'un outil d'aide aux agents responsables de la régulation d'une ligne de métro [Brézillon *et al.*, 1997]. Nous présentons dans cette section le projet SART dans son ensemble : les raisons ayant conduit à l'établissement de ce projet, les fonctionnalités attendues du système, puis son architecture logicielle.

¹⁴ Ce projet est encadré par deux conventions nationales entre d'une part la RATP et l'Université Pierre et Marie Curie (UPMC, Paris 6) et d'autre part l'entreprise Metrô de Rio de Janeiro (Metrô-Rio) et l'Université Fédérale de Rio de Janeiro (UFRJ) et une convention internationale entre l'UPMC et l'UFRJ.

1.1 Genèse du projet

Le projet a été initialement proposé par deux responsables du poste de commande centralisé (PCC) des lignes du métro de Rio, afin de pallier les défauts d'organisation locale du PCC. En effet, les chefs de régulation brésiliens sont de jeunes cadres nouvellement embauchés par l'entreprise, sans expérience du terrain. Ceux-ci ont une bonne connaissance technique mais livresque. Ainsi il arrive que les actions ordonnées par les chefs de régulation conduisent à des inconsistances au niveau local. Metrô-Rio a donc décidé de concevoir un outil d'aide à la décision pour ses chefs de régulation, dont le but est, en cas d'incident, de proposer une stratégie de résolution adaptée à la situation.

Ceci n'est pas le cas dans le métro parisien, car les chefs de régulation de la RATP n'accèdent à ce poste qu'après une expérience réelle du terrain. Ainsi les responsables de la régulation d'une ligne de métro parisien sont capables d'estimer les conséquences de leurs décisions au niveau local. Toutefois un tel projet intéresse la RATP à l'heure où il est question de capitalisation des connaissances. Ces connaissances capitalisées pouvant alors mieux être exploitées pour améliorer la qualité de service à Paris et l'exportation du savoir-faire de l'entreprise.

Une coopération est alors mise en place entre les deux entreprises et les universités de Paris 6 (Université Pierre et Marie Curie, UPMC) et Fédérale de Rio de Janeiro (UFRJ). Les deux responsables du PCC de Metrô-Rio ont effectué une thèse de doctorat en informatique en cotutelle à l'UFRJ et à l'UPMC afin d'intégrer des paradigmes d'intelligence artificielle dans le projet SART. Ils ont passé une année (1997-98) en France pour initialiser le projet. La structure du système d'aide et les principales fonctionnalités ont alors été définies.

Les deux entreprises ont toutefois besoin, dans ces deux cas, d'un système de capitalisation de l'expérience sur la gestion des incidents. Cette activité étant fortement liée à la situation, des études sont nécessaires pour l'intégration du contexte dans un système de connaissance. Le contexte joue donc un rôle particulier dès la conception même du projet. Le rôle du contexte est de plus renforcé par les différences entre les deux réseaux et les spécificités (langue, culture d'entreprise, niveau et type d'expertise des opérateurs...).

1.2 Fonctionnalités

Les principes fondamentaux de SART sont les suivants :

1. Le système doit être paramétrable, tant en ce qui concerne l'interface (langue, notamment) qu'en ce qui concerne les connaissances nécessaires à la gestion d'incidents sur une ligne de métro particulière. Outre les interfaces et les paramètres

d'une ligne, les éléments de connaissance du domaine ne sont pas forcément identiques d'une application à l'autre, car les organisations internes et le fonctionnement des différents réseaux de métro sont distincts. Aussi est-il nécessaire de considérer ces connaissances du domaine en tant que paramètres du système. Ceci est vrai à plus forte raison pour les connaissances sur la gestion même des incidents, puisque celles-ci dépendent à la fois de l'organisation des éléments du domaine, de politiques internes et de la culture d'entreprise. Tous ces éléments peuvent être introduits dans le système de manière déclarative (c'est à dire qu'ils ne sont pas inscrits dans le code du programme, mais qu'ils sont lus par le système lors de son initialisation).

2. Pour une aide efficace sur la gestion des incidents sur une ligne de métro, il est nécessaire de tenir compte de la situation, afin d'adapter au mieux les stratégies de résolution au contexte de l'incident. Ainsi, les connaissances de SART doivent être adaptées à la prise en compte du contexte. C'est pour cette raison que nous avons développé des structures de connaissances telles que les graphes contextuels (voir les troisième et le cinquième chapitres).
3. Pour suivre l'évolution des connaissances, des techniques et des stratégies, SART devra être capable d'assimiler des enseignements, notamment de la part des chefs de régulation. Pour cela, les connaissances du domaine et les bases de connaissances sur la résolution d'incidents pourront être complétées grâce à une discussion entre le système et l'utilisateur. La phase d'apprentissage des connaissances sur les stratégies sera plus amplement détaillée dans la section
4. Ces principes sont la base du système SART. D'autres fonctionnalités, secondaires par rapport à celles-ci, complètent cette spécification fonctionnelle. Le principe du système multi-expert SART est d'ajouter des fonctionnalités en intégrant un nouvel agent spécialisé pour chacune d'elles. Ainsi d'autres fonctionnalités sont d'ores et déjà envisagées, telles que des outils statistiques sur les fréquences d'incidents, d'aide à la maintenance préventive des matériels roulants et des installations fixes, de formation des futurs chefs de régulation, d'aide à l'élaboration de procédures spécialisées... Celles-ci seront prises en compte dans une deuxième phase de développement de SART, une fois que les fonctionnalités de base et les premières fonctions d'aide auront été développées et testées en condition réelle.

1.3 Architecture logicielle

Il a été décidé d'implémenter SART sous la forme d'un système multi-agent de type multi-expert. Ceci permet, sur une même plate-forme de communication, d'ajouter des fonctionnalités par l'insertion d'agents spécialisés dans une tâche. A l'heure actuelle, 4 agents

sont prévus : un agent configurateur de ligne, un simulateur de trafic, un agent gestionnaire d'incidents et un agent de communication.

Le *configurateur de ligne* permet de définir et de modifier les paramètres d'une ligne de métro (tracé, courbure, déclivité, stations, sectionnements, signalisation...). Certains paramètres peuvent être modifiés temporairement de manière dynamique pour tester des hypothèses par le simulateur de trafic.

Le *simulateur de trafic* calcule l'évolution du trafic sur la ligne, en fonction d'un état initial (position des trains sur la ligne, état des signaux...), des paramètres actuels de la ligne (gérés par le configurateur) et du contexte. Ces différents éléments sont passés en paramètres au simulateur, ainsi le calcul peut être réalisé afin de tester des hypothèses, soit sur la structure de la ligne, soit sur une stratégie de résolution d'incident adaptée au contexte.

L'*agent gestionnaire d'incidents* a pour but la maintenance des bases de connaissances sur la gestion des incidents et la réutilisation directe de ces bases pour l'aide à la gestion d'incident en temps réel ou le retour d'expérience. L'agent gestionnaire d'incident est plus amplement détaillé dans la suite de ce chapitre, car ils est plus précisément le cadre d'application de nos travaux.

Ces trois agents communiqueront entre eux et avec l'utilisateur via l'*agent de communication*, qui jouera ainsi le double rôle de plate-forme de communication du système multi-agent et d'interface avec l'utilisateur. Il sera chargé notamment de décomposer les questions complexes de l'utilisateur en entités traitables par les différents agents et de combiner les réponses de ceux-ci en une unique réponse à l'utilisateur. L'utilisateur n'aura ainsi qu'un seul interlocuteur et n'aura pas à s'adapter à différents agents et à leurs langages respectifs. Les flux d'information entre l'utilisateur et les différents agents de SART sont résumés dans la Figure 6-1.

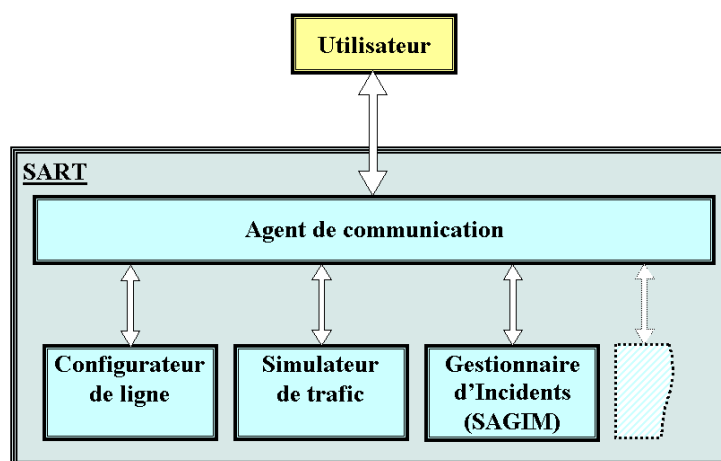


Figure 6-1 : Structure de communication de SART

Cette structure autorise l'ajout de nouvelles fonctionnalités à SART de manière quasi transparente aux yeux de l'utilisateur. Les nouveaux agents devront simplement respecter les protocoles de communication de SART et pourront partager les connaissances accumulées pour réaliser leurs buts. Toutefois, il faut noter que les différents agents peuvent être développés indépendamment les uns des autres.

Section 2 :

L'agent gestionnaire d'incidents

Nous détaillons dans cette section l'agent gestionnaire d'incidents de SART. Pour tester cet agent, l'agent de communication n'étant pas encore réalisé, nous avons développé un système indépendant sur la base de l'agent gestionnaire d'incidents : le Système d'Aide à la Gestion d'Incidents dans le Métro (SAGIM). Nous utiliserons désormais ce sigle pour désigner à la fois le système autonome développé et l'agent gestionnaire d'incident de SART. Nous précisons d'abord le rôle de SAGIM au sein de SART, puis nous présentons son architecture logicielle et enfin nous détaillons son fonctionnement suivant les trois modes d'utilisation possibles de SAGIM.

2.1 Rôle de SAGIM

L'agent gestionnaire d'incidents a trois buts principaux : premièrement il est chargé de la maintenance des bases de connaissances sur la gestion des incidents et deuxièmement il est le support de la réutilisation directe de ces bases pour l'aide à la gestion d'incident en temps réel et troisièmement il doit proposer des sessions de retour d'expérience. Pour cela il travaille suivant trois modes : archiviste, conseiller et courtier.

En mode *archiviste*, il travaille en interaction avec l'utilisateur afin d'acquérir des nouvelles connaissances sur la résolution des incidents et les stratégies appliquées et les organise dans des bases utilisables par les autres modes d'utilisation de SAGIM et les autres agents de SART.

En mode *conseiller*, il a pour but de proposer, en temps réel, des stratégies de résolution d'incidents adaptées au contexte. Il utilise les connaissances accumulées par le mode archiviste, des données sur la situation courante récupérées automatiquement et des données fournies par l'utilisateur pour retrouver et proposer des stratégies spécifiques applicables pour résoudre l'incident courant.

En mode *courtier*, il a pour but de proposer des stratégies de résolution d'incidents adaptées au contexte, soit juste après la saisie d'un incident, soit pendant des périodes calmes. Comme en mode conseiller, il exploite les connaissances accumulées par le mode archiviste, mais ses recherches ne sont basées que sur des éléments fournis par l'utilisateur. Celui-ci est libre en ce qui concerne ces recherches, et le mode courtier propose de comparer des stratégies diverses. Son rôle est d'amener le chef de régulation à étudier des cas d'incidents proches et de déterminer ce qui les distingue, afin d'affiner son expertise.

2.2 Structure logicielle de SAGIM

Pour faciliter la compréhensibilité du système, nous avons choisi de baser SAGIM sur des raisonnements artificiels proches de raisonnements réels des opérateurs. Ainsi les deux modes de réflexion que nous avons pu observer chez les chefs de régulation sont le raisonnement par analogie et surtout un raisonnement de type diagnostic/action impliquant diverses activités de différents niveaux. Nous nous proposons donc d'utiliser conjointement le Raisonnement à Partir de Cas (RàPC, voir le quatrième chapitre) et un raisonnement basé sur une structuration des activités par des graphes contextuels (voir les troisième et cinquième chapitres). Ces deux modes ont été choisis pour leurs avantages respectifs, leur intégration du contexte et leur complémentarité. Les graphes contextuels sont une représentation structurée de raisonnements à base de règles de production. Aussi le choix de l'hybridation de ces deux modes de raisonnements est conforté par l'étude de Sun [1995] qui indique que la combinaison de règles et de similarités rend les systèmes de connaissances robustes.

Ces deux raisonnements partagent une certaine vision du contexte : à chaque saisie d'un incident, les connaissances contextuelles impliquées dans la sélection de la stratégie dans les graphes contextuels sont ajoutées à la fiche descriptive de l'incident intégrée dans les bases de cas ; de plus, quand l'intégration d'une nouvelle pratique dans un graphe contextuel met en œuvre une connaissance contextuelle inconnue du système, cette connaissance est ajoutée à la liste des connaissances contextuelles déjà rencontrées et doit faire l'objet de la spécification des données relatives à sa mesure de similarité. Chaque mode de raisonnement gère ses propres bases de connaissances spécifiques, mais partagent de nombreuses connaissances générales, notamment sur le domaine, celles liées au contexte et aux connaissances contextuelles déjà rencontrées.

Comme souligné dans le premier chapitre (section 3.3, page 40), les opérateurs travaillent principalement sur la base d'adaptation de procédures à la réalité de la situation. Ainsi il nous semble important de conserver une trace des deux types de connaissances stratégiques que sont les procédures et les pratiques. SAGIM gère ainsi deux bases de type activité :

1. Une base contenant les procédures officielles (la base de procédures) ;

2. Une base contenant les stratégies appliquées (la base de stratégies).

Le nombre important d'incidents se produisant sur chaque ligne de métro et leur nature très diverse quant à leur originalité (voir le quatrième chapitre, section 3.4, page 128) fait que l'application du raisonnement à partir de cas est spécifique dans notre situation. Certains incidents sont très fréquents (plusieurs fois par jour) et leur gestion est parfaitement rodée ; d'autres sont extrêmement rares (voire exceptionnels). Aussi les incidents ne sont pas tous égaux face à leur originalité. Il est donc important de restreindre le nombre d'incidents sur lesquels le raisonnement à partir de cas sera effectué pour ne pas l'alourdir inutilement. Nous définissons ainsi deux bases d'incidents :

1. Une base contenant tous les incidents qui se sont produits sur la ligne (appelée base historique des incidents) ;
2. Une base contenant les incidents typiques rencontrés (la base des cas typiques) ;

Ces quatre bases se décomposent en deux groupes au regard de la structure des éléments qui y sont enregistrés. Premièrement les bases historique des incidents et de cas typiques sont composées d'éléments de type « rapport d'incident » (la forme de ces éléments sera spécifiée dans la section 3). Un incident y est enregistré avec ses descripteurs d'incident, de contexte, la stratégie appliquée pour le résoudre et des descripteurs de qualité de résolution. Les deux autres bases contiennent des éléments représentant les activités et leur organisation sous forme de graphes contextuels, qui sont des structures représentant des raisonnements.

L'existence de deux bases pour chacune des deux structures de mémorisation s'explique par la coexistence pour chacune de données officielles et de données directement utilisables par les chefs de régulation. Ainsi, les procédures sont les « organisateurs officiels » de l'activité, que les chefs de régulation adaptent à la situation réelle pour chaque incident (ils élaborent alors une stratégie pour résoudre l'incident courant en tenant compte du contexte). Les incidents typiques sont les incidents rencontrés qui se distinguent des autres incidents de même type, soit par leur contexte, soit par la stratégie appliquée pour les résoudre. Ils sont réutilisables par les chefs de régulation en adaptant la stratégie à un incident courant grâce à un raisonnement de type « analogie ».

SAGIM intègre donc deux modes de raisonnements assez différents. SAGIM est ainsi composé de deux modules de raisonnements (un à base de Cas, l'autre à base d'activités), de leurs bases de données spécifiques, de données communes (sur le contexte), d'une plateforme d'intégration des deux modes de raisonnement. Le système accède à des connaissances plus générales sur le domaine via SART (paramètres de la ligne, données sur le trafic). La Figure 6-2 montre la structure de SAGIM et son intégration dans SART. Nous détaillons ceci dans la section 3, consacrée à l'implémentation de SAGIM.

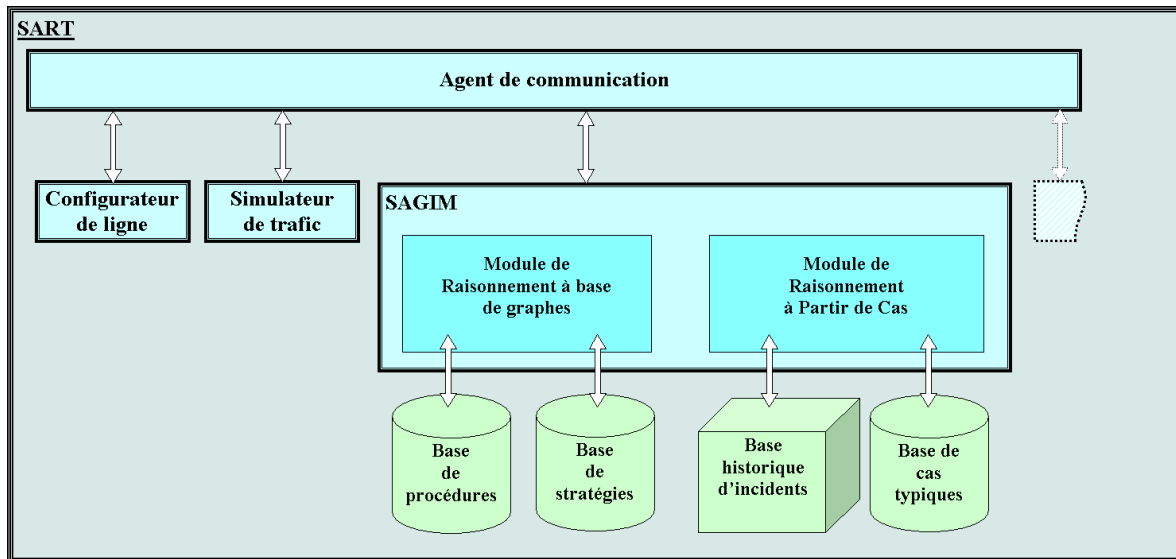


Figure 6-2: Architecture logicielle de SAGIM

2.3 Les trois modes d'utilisation de SAGIM

SAGIM a pour rôle d'accumuler les connaissances sur les stratégies de résolution des incidents et de proposer deux principaux mode d'investigation des bases de connaissances ainsi réalisées (en temps réel et en temps différé). Cet agent est utilisable suivant trois modes. Le mode archiviste est chargé de la maintenance des bases de connaissances, le mode conseiller propose des stratégies de résolution d'un incident en cours en tenant compte des éléments de contexte dont il dispose et le mode courtier propose à l'utilisateur d'étudier les différentes stratégies applicables pour un incident de son choix dans différents contextes. Nous allons maintenant détailler les fonctionnalités de ces trois modes d'utilisation.

2.3.1 L'archiviste

L'archiviste a en charge la maintenance des quatre principales bases de connaissances nécessaires aux deux modes de raisonnement utilisés dans SAGIM et l'intégration des nouvelles stratégies de résolution des incidents dans ces bases.

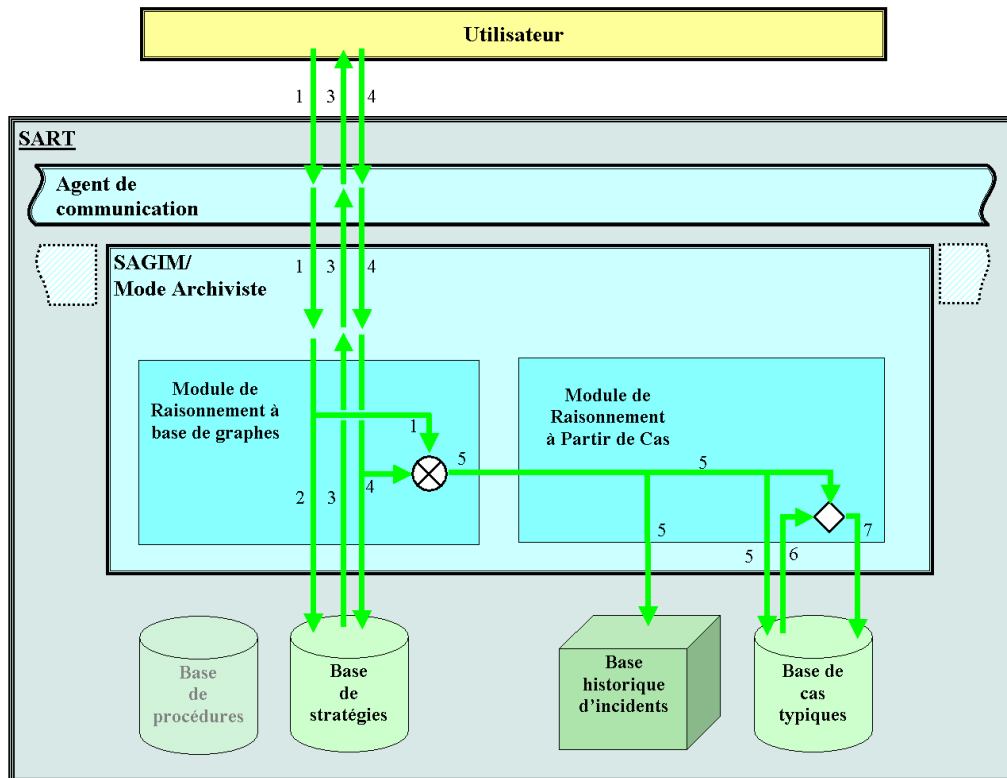


Figure 6-3 : Interaction utilisateur / Archiviste et flux de données

L'archiviste est appelé après chaque incident survenant sur la ligne concernée. L'utilisateur lui fournit les données décrivant l'incident, son contexte, la stratégie appliquée et les conséquences sur le trafic. Ces données (1) sont transmises par l'agent de communication à l'archiviste. Celui-ci intègre la stratégie appliquée à la base de stratégies (2). Cette étape peut engager une discussion (3 et 4) avec l'utilisateur afin de spécifier des éléments de contexte n'ayant pas été fournis, mais nécessaires pour distinguer la stratégie à intégrer de stratégies proches déjà employées. Ces descripteurs de contexte (4) sont ajoutés à la description de l'incident (1) pour l'enrichir. La description enrichie (5) est intégrée dans la base historique d'incidents. Elle est de plus transmise au module de RàPC pour étudier la typicité du cas présent. L'archiviste compare alors le cas courant au cas typique le plus proche (6) selon le module de RàPC. Si le cas courant est jugé suffisamment typique, il est intégré dans la base de cas typiques (7).

Les connaissances ainsi accumulées seront utilisées pour proposer des stratégies adaptées à un incident donné avec son contexte et pour expliquer les raisonnements ayant conduit le système à faire de telles propositions. Nous allons maintenant montrer le fonctionnement de ces deux modes d'aide.

2.3.2 Le conseiller

Le mode conseiller est activé lors d'un incident sur la ligne (utilisation « on-line »). Il aide alors l'utilisateur à définir une stratégie adaptée au contexte applicable pour résoudre cet incident. Son principe cherche à minimiser l'interaction avec l'utilisateur afin de laisser le plus de temps possible à l'utilisateur, et à se focaliser sur l'incident et le contexte courant, afin de proposer des solutions précises. Ceci est réalisé en deux ou trois temps. Premièrement le conseiller demande à l'utilisateur de spécifier les caractéristiques de l'incident et les éléments de contexte qui sont en sa possession (le conseiller est capable d'acquérir automatiquement certaines informations telles que la date, l'heure et ce qui s'en déduit, la position des trains sur la ligne, l'état d'alimentation des sections et sous-sections, etc., mais d'autres éléments sont hors de sa portée, comme le type d'incident ou les causes de celui-ci).

Deuxièmement, il recherche les stratégies applicables en utilisant les deux modes de raisonnement qu'il a à sa disposition. Le raisonnement à partir de cas lui fournit les stratégies appliquées pour résoudre les cas les plus proches contenus dans la base de cas typiques. Le raisonnement à base de règles fournit une ou plusieurs stratégies, ceci dépendant du degré de précision du contexte connu et de l'adéquation de celui-ci aux critères enregistrés dans les bases de procédures et de stratégies. A l'occasion le contexte peut être précisé afin de limiter le nombre de stratégies proposées par le système.

Toutes ces stratégies ne sont données à l'utilisateur qu'à titre indicatif. C'est à lui que revient la décision finale.

Une troisième phase peut être utile. Il s'agit d'une demande, de la part de l'utilisateur, d'explication des stratégies proposées par le raisonnement basé sur les activités. Celui-ci répond alors en traçant son cheminement (il s'agit en fait de montrer la succession des choix effectués par le moteur de raisonnement), en montrant comment l'activité en question est organisée (par le graphe contextuel) et en donnant les explications qu'il a enregistrées lors de l'acquisition de la pratique concernée (voir Troisième chapitre :3.3).

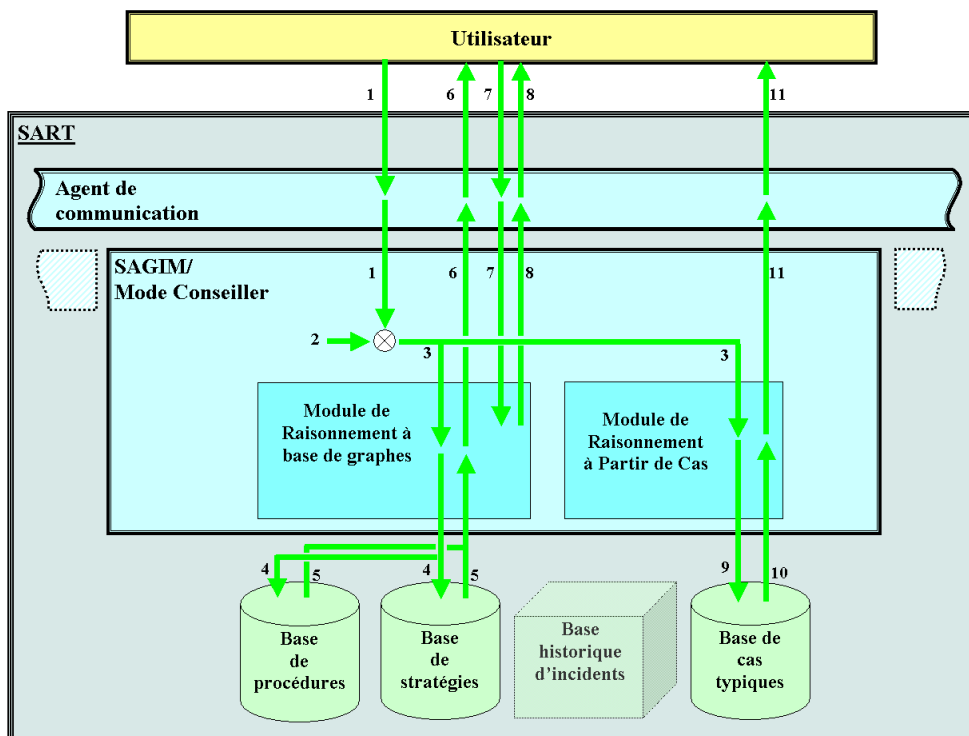


Figure 6-4 : Interaction utilisateur / Conseiller et flux d'informations

Pendant la première phase, le conseiller demande à l'utilisateur de spécifier les caractéristiques de l'incident et les éléments de contexte qui sont en sa possession (1). Le conseiller est capable d'acquérir automatiquement certaines informations, telles que la date, l'heure et ce qui s'en déduit, la position des trains sur la ligne, l'état d'alimentation, etc., (2). La description de l'incident et de son contexte fournie par l'utilisateur est complétée par les informations apportées par le conseiller et forment une description augmentée de l'incident (3) qui est transmise aux modules de raisonnement à Partir de cas et à base de graphes. Lors de la deuxième phase, le conseiller recherche les stratégies applicables en utilisant les deux modes de raisonnement qu'il a à sa disposition. Le raisonnement à base de graphes fournit une ou plusieurs stratégies (5). Le raisonnement à partir de cas fournit les stratégies appliquées pour résoudre les cas les plus proches contenus dans la base de cas typiques (10). Toutes ces stratégies ne sont données à l'utilisateur qu'à titre d'indication. C'est à lui que revient la décision finale. Une troisième phase peut être utile. Il s'agit d'une demande, de la part de l'utilisateur, d'explication des stratégies proposées par le raisonnement à base de règles (7). Celui-ci répond alors en traçant le cheminement de son raisonnement (8). Une fois l'incident résolu, les informations récoltées par le conseiller sont transmises à l'archiviste pour réaliser la mise à jour des bases de données.

2.3.3 Le courtier

Le mode courtier est activé automatiquement juste après la phase de saisie d'un incident ou à n'importe quel instant par l'utilisateur (utilisation « off-line »). Il permet d'interroger les bases de connaissances en spécifiant soit des caractéristiques d'incident, soit des éléments de contexte. Les requêtes peuvent être raffinées ou élargies à volonté. Ceci permet à l'utilisateur de comparer des stratégies appliquées pour résoudre un même incident mais dans des contextes différents, voire comparer des stratégies appliquées dans le même contexte et chercher à comprendre ce qui les distingue. Sur ce plan l'utilisateur est libre de faire ce qu'il veut, le mode courtier est un outil d'aide à la formation continue par retour d'expérience.

Le flux d'information entre l'utilisateur et le courtier (Figure 6-5) ressemble à celui précédemment montré pour le mode conseiller, car la différence principale entre eux se situe dans le mode d'investigation des bases de données et la quantité d'informations échangées avec l'utilisateur.

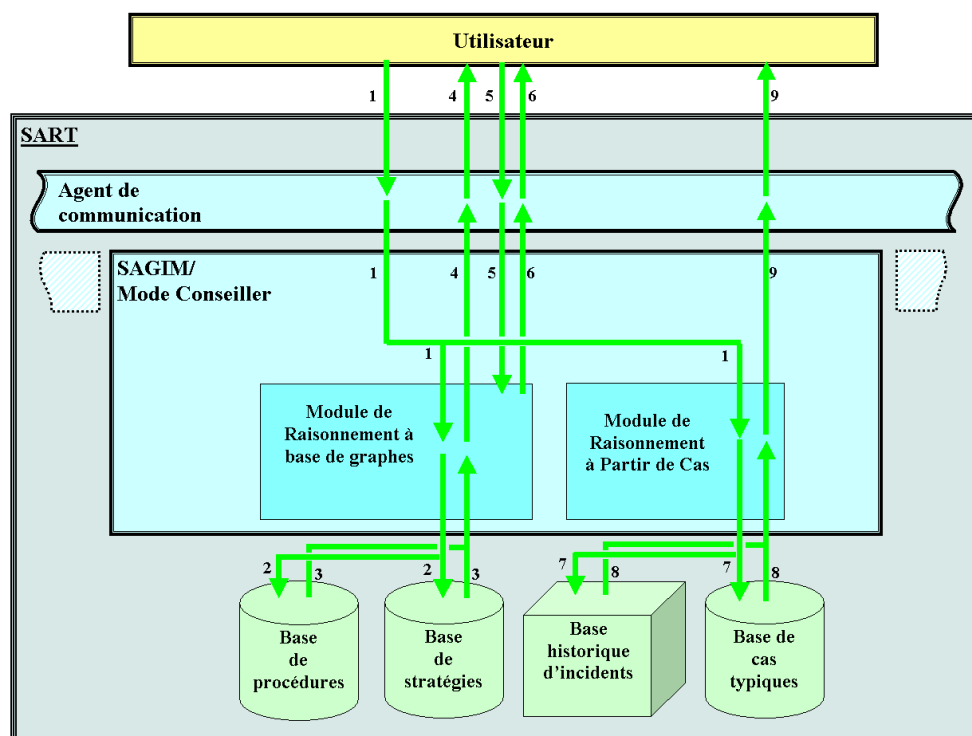


Figure 6-5 : Interaction utilisateur / Courtier et flux d'informations

Une fois les caractéristiques souhaitées rentrées (1), le courtier interroge les bases de données (2 et 7). La différence majeure avec l'investigation faite par le conseiller réside dans la quantité, la spécificité et la présentation des stratégies. En effet, le conseiller doit présenter une sélection limitée de stratégies ciblées sur l'incident courant et son contexte, alors que le courtier doit sélectionner un nombre plus important de stratégies (3, 4, 8 et 9) et diversifier celles-ci, afin d'élargir le champ de vision du chef de régulation. Comme pour le conseiller,

l'utilisateur peut demander des explications (5) qui lui seront fournies sous forme de traces du parcours des procédures et des stratégies (6).

L'utilisateur est libre de mener l'interrogation comme il l'entend, se concentrant ainsi sur les incidents et les contextes qui l'intéressent. C'est une formation continue par retour d'expérience. Ceci permet de partager des connaissances entre les chefs de régulation et peut pallier la suppression de la formation par friction actuellement rencontrée au PCC du métro parisien.

Section 3 :

Spécification fonctionnelle de SAGIM

Comme précisé dans la section précédente, nous avons choisi deux modes de raisonnements très différents pour SAGIM, afin de proposer deux vues complémentaires du problème et de ses solutions. Dans cette section, nous montrons comment ces deux modes de raisonnement sont appliqués effectivement dans SAGIM. Nous commençons par le raisonnement basé sur la structuration des activités de l'opérateur par des graphes contextuels. Ensuite nous étudions les spécificités de notre application concernant le raisonnement à partir de cas. Enfin, nous analysons quelles sont les données spécifiques de SAGIM qui sont partagées par les deux modes de raisonnement.

3.1 Raisonnement à base de Graphes Contextuels

Nous avons d'abord choisi un raisonnement fortement structuré afin de faciliter les explications pouvant être fournies par le système à l'utilisateur. Le modèle retenu est une modélisation des activités par le biais des graphes contextuels (voir le troisième et le cinquième chapitre). Nous analysons, dans cette sous-section, l'implémentation de ce modèle d'activité. Nous revenons rapidement sur les principes de ce raisonnement. Nous justifions ensuite les choix d'implémentation effectués. Nous présentons ensuite comment l'algorithme d'acquisition incrémentale de connaissances stratégiques opère en pratique dans SAGIM.

3.1.1 Principes du raisonnement

Le principe du raisonnement est le suivant : les activités des opérateurs sont en constante évolution et apportent régulièrement des innovations plus ou moins importantes quant aux raisonnements suivis et aux plans d'action appliqués à la gestion des situations incidentelles. Le système d'aide ne peut pas prévoir ces évolutions, mais doit être capable d'en tenir compte. La prise en compte de ces innovations passe par deux grandes idées : premièrement

le système ne peut que proposer une stratégie, libre à l'opérateur de la mettre en pratique ou non ; deuxièmement, le système doit modifier ses connaissances sur les pratiques réelles des opérateurs afin d'affiner ses propositions avec l'expertise qu'il accumule.

La solution proposée consiste en une modélisation des activités des opérateurs. Ceci permet, dans un mode d'interrogation de SAGIM, de proposer un plan d'action basé sur la décomposition des activités en activités secondaires. L'utilisation de graphes contextuels pour l'organisation des différentes méthodes actuellement connues pour la réalisation d'une tâche permettant ainsi pour chaque activité de définir le plan partiel adapté à la situation. La composition de ces plans partiels et hiérarchisés aboutit à l'établissement d'une stratégie complète. Les graphes contextuels permettent également une accumulation de l'expertise grâce à l'algorithme d'acquisition incrémentale de connaissances qui leur est associé.

Les activités peuvent faire référence à d'autres activités. Pour éviter toute référence croisée, nous organisons ces activités dans un réseau représentant les liens de dépendance des activités entre elles. Pour éviter tout bouclage, il suffit d'interdire à toute activité de faire référence à des activités situées à un niveau supérieur ou à elle-même. Nous obtenons ainsi un système de hiérarchisation des activités, qui permet également d'interpréter les activités des opérateurs à un niveau conceptuel supérieur.

3.1.2 Implémentation et choix d'implémentation

Les graphes contextuels sont des graphes orientés acycliques ayant une seule source et un seul puits. Dans le cinquième chapitre, nous avons de plus ajouté la condition de planarité, afin de faciliter leur assimilation par les opérateurs. Nous avons vu alors que la planarité devait être garantie par l'application d'algorithmes d'aplanissement coûteux.

Nous avons donc choisi, pour l'implémentation de SAGIM, de restreindre la structure des graphes contextuels à des graphes série/parallèle. Ce choix est fondé sur deux éléments : premièrement, de tels graphes sont toujours planaires et les algorithmes de représentation graphique sont simples et rapides ; deuxièmement, il existe une structure simple et intuitive pour l'enregistrement de tels graphes dans un fichier (que nous détaillons dans la section 4.2.2.3 de ce chapitre). Les graphes série/parallèle peuvent être expliqués de la manière suivante : tout nœud contextuel est associé de manière bijective à un nœud de recombinaison et chaque groupe nœud contextuel-nœud de recombinaison ayant été ouvert dans le cadre d'un autre doit être refermé avant celui l'encadrant. Ainsi la Figure 6-6 représente la version série/parallèle du graphe contextuel correspondant à la gestion d'un incident traction (voir Figure 3-10, page 108). Dans de tels graphes, toutes les branches issues d'un même nœud contextuel se rejoignent en un même nœud de recombinaison, et les branches se referment dans l'ordre inverse de leur ouverture. Ainsi dans la figure suivante, les

branches issues du nœud contextuel C3 se rejoignent avant celles issues du nœud C2, qui elles-mêmes se rejoignent avant celles issues du nœud C1.

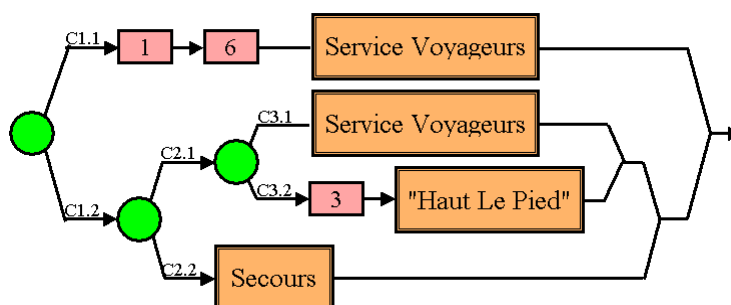


Figure 6-6: Graphe contextuel de la gestion de l'incident traction, version série/parallèle

Ce choix d'implémentation réalise un compromis entre la généralisation des nouvelles pratiques et la complexité de la structure. Les graphes série/parallèle sont très simples, mais les différentes stratégies représentées dans le graphe partagent une moindre part de leur structure. Ainsi certaines stratégies, bien qu'identiques sur la portion terminale de leur plan ne la partagent pas dans le graphe. Ceci peut amoindrir l'effet de généralisation de pratiques quand celles-ci modifient une partie qui aurait été partagée dans une structure moins contraignante de graphes contextuels. Ce choix n'a toutefois aucun effet sur le deuxième mécanisme de généralisation qui passe par la référence à des activités secondaires, d'où l'importance accrue de la hiérarchisation des activités. Ainsi si une pratique apprise modifie le second chemin (C1.2, C2.1, C3.1) et propose une alternative à l'activité secondaire *Service Voyageurs*, alors cette alternative ne sera pas disponible pour la stratégie correspondant au premier chemin. Par contre si une pratique issue du second chemin modifie l'activité *Service Voyageurs*, cette modification sera généralisée à la stratégie passant par la première chemin.

Nous avons toutefois vérifié que les graphes contextuels génériques sont bien utilisables, mais la génération automatique de leur représentation est plus complexe et peut troubler les opérateurs. Nous préférons donc perdre un peu en généralisation des pratiques et gagner largement en compréhension du système par les opérateurs, ce qui facilite l'appropriation de l'outil proposé.

3.1.3 L'algorithme d'acquisition incrémentale de connaissances

L'algorithme d'acquisition de connaissances est le cœur du module de raisonnement basé sur le modèle des activités des opérateurs. Le système d'aide guide, pendant cette phase d'acquisition, à la fois l'introspection de l'opérateur et l'introduction des innovations dans les structures maintenues.

Pour cela, le système opère activité par activité, en commençant par celle de plus haut niveau (l'activité de gestion de l'événement incidentel). Pour l'activité traitée, le système effectue d'abord la recherche de la stratégie qui aurait dû être appliquée en parcourant le graphe et demandant à l'opérateur de préciser la valeur des connaissances contextuelles rencontrées. Une fois la stratégie théorique obtenue, le système demande à l'opérateur quelles actions et activités ont été réalisées parmi celles attendues, puis si d'autres actions ou activités ont été réalisées. Cette étape guide l'opérateur dans la décomposition de la pratique globale en ses composantes correspondant aux différentes activités. Si la composante de la pratique correspondant à l'activité actuellement développée est différente de la stratégie attendue, alors le système demande à l'opérateur de préciser l'élément contextuel l'ayant conduit à adapter sa pratique, ainsi que des explications sur le raisonnement suivi à cet instant. Il introduit ensuite la nouvelle pratique pour cette activité et ajoute les activités secondaires de l'activité courante en tête de liste des activités à développer, puis passe au développement de l'activité suivante.

Une fois toutes les activités concernées développées, le système poursuit la saisie de la pratique. Il s'agit alors d'un « texte à trous » organisé hiérarchiquement et temporellement à partir des différentes composantes de la pratique identifiées lors de l'introduction des composantes dans les graphes contextuels correspondant aux activités concernées. La saisie de l'incident est ensuite terminée par celle de descripteurs généraux sur les résultats obtenus et le tout est transmis au module de raisonnement à partir de cas.

3.2 Raisonnement à Partir de Cas

Le Raisonnement à Partir de Cas (RàPC) a été choisi pour ses bonnes performances, notamment dans des situations inattendues, et son intégration native du contexte. Nous revenons rapidement sur les principes du RàPC (ce mode de raisonnement est présenté dans le quatrième chapitre, section 3). Puis nous exposons les spécificités de notre application par rapport au fonctionnement classique de ce raisonnement. Nous présentons ensuite les détails d'implémentation du RàPC pour SAGIM : au niveau de la structure des cas, de la mesure de similarité et enfin de la structure de la base de cas.

3.2.1 Principes

Le Raisonnement à Partir de Cas (RàPC) est un raisonnement par analogie : une solution à un problème est construite à partir des solutions de problèmes proches précédemment rencontrés. Les problèmes précédemment traités sont stockés, avec leur solution, dans une base de cas et sont comparés au problème courant grâce au calcul d'une somme pondérée de comparaison locale des différents descripteurs composant les cas. Cette somme est appelée mesure de similarité et contient l'expertise sur la classe de problèmes traités.

Ce mode de raisonnement prend en compte le contexte de manière explicite, du moment que celui-ci est un élément de description des problèmes. Les autres principaux avantages de ce raisonnement sont son pouvoir de généralisation et sa capacité à s'adapter à des données manquantes (ce qui est très important dans notre domaine d'application où les caractéristiques d'un incident sont connues au fur et à mesure de l'évolution de celui-ci).

Nous ne proposons aucune adaptation particulière des principes du RàPC dans cette thèse. Le RàPC est déjà fortement sensible au contexte et de nombreuses études déjà réalisées (notamment celle de Mignot [1997]) apportent des solutions pour l'application de ce mode de raisonnement à notre cas.

3.2.2 Spécificités de notre application

3.2.2.1 Mesure de similarité dépendante du contexte

Nous avons souligné (voir le quatrième chapitre, section 3.3) que, dans notre application, la mesure de similarité est très sensible au contexte. Les idées de Mignot [1997] sur l'adaptation de la mesure de similarité en fonction du problème courant nous intéressent particulièrement.

Ainsi nous utilisons différentes mesures de similarité en fonction d'éléments primordiaux du contexte. Les éléments primordiaux que nous avons identifiés, et qui orientent fortement la stratégie de gestion de l'incident et la similarité entre les cas correspondent à l'événement incidentel et au type d'exploitation (heure creuse, heure de pointe, flanc montant et flanc descendant¹⁵). Ces deux critères modifient d'emblée la vision que l'opérateur a de la situation et influent grandement sur la compréhension qu'il en a et sur la gestion de l'incident qui s'en suit.

Les mesures de similarités ainsi différenciées sont ensuite gérées de manière classique, et évoluent en fonction de l'ajout de nouveaux critères par des mécanismes de type ID3 (tel que cela est réalisé dans ReCall) pour adapter les poids de chaque descripteur.

¹⁵ Les flancs montants et descendants correspondent respectivement à des périodes d'augmentation de la charge (passage d'heures creuses à heures de pointe) et à des périodes de diminution de la charge (passage d'heures de pointe à heures creuses).

3.2.2.2 Gestion de grandes bases de cas redondantes

Comme indiqué dans le quatrième chapitre (section 3.4, page 128), nous rencontrons 2500 incidents par an sur la ligne 4. En considérant que certains incidents sont rares, il nous faudrait une base de cas recouvrant plusieurs dizaines d'années pour être précise. D'autres incidents sont à l'inverse très fréquents (plusieurs fois par jour, notamment les « objets tombés sur les voies », « les KSA »...) et sont peu discriminés par le contexte ou la résolution (qui est en générale rapide).

Ainsi nous observons plusieurs types de comportement des incidents. Le Raisonnement à Partir de Cas ne fait pas état de ces différences. Nous ne gardons dans la base de cas que les incidents *typiques*, qu'ils le soient pour la rareté de leur type d'incident ou de leur cause, une situation particulière ou leur résolution exceptionnelle. Ayant besoin toutefois d'une trace de tous les incidents, nous conservons un résumé de chaque cas rencontré dans une base historique d'incidents, à des fins statistiques et de maintenance préventive. Nous gérons ainsi deux bases de type cas : une base de cas typiques et une base historique des incidents.

3.2.3 Structure d'un cas

Chaque cas est enregistré dans un fichier informatique composé des principaux éléments suivants :

1. Les descripteurs de l'incident : le type, la cause, les événements qui l'ont annoncé au chef de régulation...
2. Les descripteurs du contexte. Ceux-ci sont de deux types : certains éléments sont communs à tous les incidents et sont donc toujours enregistrés (date, heure, type de jour, type d'exploitation, lieu de l'incident...), d'autres ne sont utiles que pour certains types d'incidents (par exemple le numéro de la rame n'est utile –et défini– que lorsqu'une rame est mise en cause ou touchée par l'incident) et ne sont donc enregistrés que lorsqu'ils ont un sens et un impact sur la résolution de l'incident.
3. La pratique mise en œuvre pour résoudre l'incident. Cette pratique est enregistrée sous la forme de deux séquences d'actions, une qui décrit les actions menées pour résoudre l'incident même (gestion locale) et l'autre qui décrit les actions menées pour réguler le trafic résiduel (gestion globale).
4. Des descripteurs permettant d'estimer la valeur de la stratégie. Ces critères sont notamment le retard, le retard cumulé...

Les causes d'incidents ont été étudiées : ce descripteur de l'incident est organisé suivant une hiérarchie donnée dans l'annexe 1. Les événements incidentels ont été proposés comme

descripteurs importants d'un incident, car, étant la première information obtenue par le chef de régulation sur l'incident, ils influencent grandement la stratégie appliquée. Les descripteurs d'un incident sont les éléments sur lesquels reposent le RàPC. Aussi est-il important de bien les étudier, premièrement pour les définir de manière efficace, deuxièmement pour bien comprendre leur rôle dans la prise de décision (ceci permet de faciliter la mise au point de la mesure de similarité).

La particularité de notre application, en ce qui concerne le RàPC, réside dans la gestion des éléments de contextes. Chaque cas ayant ses propres éléments de contexte, un mécanisme acceptant les éléments manquants doit être mis en place dans le cadre des mesures de similarité.

3.2.4 Mesures de similarité et contexte

La mesure de similarité est un autre élément sensible du RàPC. Sa définition est simple d'un point de vue formel, mais la mise au point des poids de la mesure est délicate et demande une expertise de haut niveau. Dans notre domaine, certains éléments contraignent si fortement la résolution d'un incident qu'il semble naturel de leur donner plus d'importance. Comme précisé ci-dessus, Nous avons défini différentes mesures de similarité, sélectionnées par ces descripteurs principaux du contexte. Chaque mesure de similarité se distingue des autres par les poids accordés à chaque descripteur.

Les descripteurs étant, dans notre cas, des éléments de contexte à valeurs discrètes, la comparaison locale de chaque descripteur est basée sur des matrices. Chaque fois qu'une nouvelle connaissance contextuelle est insérée dans le système, celui-ci demande à l'opérateur de remplir la matrice de comparaison locale des différentes valeurs connues pour ce nouveau descripteur (nous reviendrons sur l'évolution de ces connaissances contextuelles dans la section 3.3 en prenant un point de vue global sur SAGIM). Ceci est, de nouveau, une acquisition de l'expertise des opérateurs par le système.

3.2.5 Structure de la base de cas

Un autre élément important du RàPC pour notre application est le grand nombre d'exemples. Ceci nous a déjà poussé à gérer deux bases d'incidents, mais le nombre de cas typiques (ceux qui nous intéressent ici pour le RàPC) reste élevé. Aussi organisons-nous la base de cas suivant une arborescence de répertoires, chaque passage d'un répertoire à un sous-répertoire affinant la recherche.

Pour cela nous avons cherché les critères, communs à tous les enregistrements, qui créent des partitions de plus en plus fines de la base de cas. Pour des raisons techniques de

communication entre les systèmes, le premier critère de partition des cas sera la ligne. Les suivants sont respectivement : l'événement incidentel et le type d'exploitation (heure de pointe, heure creuse, flanc montant ou descendant). Cette partition de la base de cas correspond, en fait, aux différentes mesures de similarité gérées par SAGIM. Chaque répertoire terminal est associé à une mesure de similarité spécifique.

3.3 Données communes

Nous revenons dans cette partie sur les données communes aux deux modes de raisonnement employés dans SAGIM. Ces données partagées concernent d'abord les connaissances contextuelles déjà rencontrées par SAGIM. Ensuite nous faisons cas des connaissances liées aux incidents et aux événements incidentels.

3.3.1 Evolution des connaissances contextuelles

Les connaissances contextuelles sont des données communes aux deux modules de raisonnement. Quand une nouvelle pratique est insérée dans un graphe contextuel, elle induit la création d'un nœud contextuel. Ce nœud peut faire référence soit à une connaissance contextuelle déjà rencontrée par SAGIM (lors de l'introduction précédente d'une pratique), soit être totalement nouvelle pour le système. Dans ce dernier cas, SAGIM ajoute cette connaissance contextuelle et ses valeurs connues dans sa liste de connaissances contextuelles et demande à l'opérateur de remplir une table de comparaison des différentes valeurs pour le calcul de la mesure de similarité.

La liste des connaissances contextuelles est utile pour guider l'opérateur lors de la précision des éléments l'ayant poussé à suivre une nouvelle stratégie. En effet, quand une pratique diffère de la stratégie attendue, le système demande à l'opérateur de saisir l'élément contextuel ayant assis sa décision. SAGIM peut alors assister l'opérateur en proposant la liste des connaissances contextuelles déjà rencontrées (tout en laissant, bien sûr, la possibilité à l'opérateur d'en saisir une nouvelle). Ce mécanisme d'assistance à la sélection de connaissances contextuelles lors d'acquisition incrémentale de connaissances stratégiques est plus rapide, plus simple pour l'opérateur et permet, de plus, d'éviter la définition de deux connaissances contextuelles identiques sous des noms différents.

3.3.2 Evolution des connaissances sur les incidents et les événements incidentels

En théorie, les événements incidentels n'évoluent pas. Toutefois, quand de nouveaux systèmes d'alerte sont mis en place, de nouveaux événements incidentels apparaissent. Il en est ainsi, par exemple des alarmes *pneu porteur* et *frotteur* dans le nouveau PCC de la ligne 4. Il est alors nécessaire d'en instruire SAGIM pour qu'il ajoute de tels nouveaux événements

incidentels dans ses bases de connaissances. Il doit ainsi ajouter l'événement incidentel en question dans sa liste d'événements proposés à l'opérateur pour la saisie, ajouter un descripteur d'activité, dont le graphe contextuel est soit vide, soit issu d'une procédure, soit adapté de celui d'un événement incidentel proche (par différenciation) et enfin ajouter une arborescence dans les bases de cas.

Les connaissances sur les incidents évoluent plus souvent. L'ingéniosité des individus malveillants est sans limite et leur créativité fait émerger de nouvelles branches dans la hiérarchie des causes d'incidents. De même l'introduction de nouveaux appareils ou systèmes amène son lot de défaillances techniques possibles et modifie également cette hiérarchie. Quand des modifications sont apportées à cette liste des causes connues d'incidents, il est nécessaire d'ajouter autant de lignes et de colonnes à la matrice de comparaison des causes d'incident en vue du calcul de la mesure de similarité. L'ajout d'une activité n'est pas nécessaire (puisque certains incidents sont résolus automatiquement par leur identification), mais la prévision de celle-ci ne coûte rien.

Section 4 :

Prototype de SAGIM

SAGIM, dans son acception décrite ci-dessus, est en cours d'implémentation. Nous avons toutefois testé les idées proposées dans cette thèse dans un prototype de SAGIM. Ce prototype a pour but de vérifier la validité et la faisabilité des propositions ci-dessus, notamment des graphes contextuels et de l'acquisition incrémentale de connaissances stratégiques basée sur une forte interaction entre l'utilisateur et le système d'aide. Nous présentons, dans cette section, le prototype de SAGIM que nous avons développé et testé en conditions réelles. Cette section expose dans un premier temps les spécification du prototype, puis l'implémentation de celui-ci et finalement nous analysons les résultats obtenus pendant la phase de validation.

4.1 Spécifications

Le prototype de SAGIM que nous avons développé a pour but de tester la validité des propositions faites sur la modélisation des activités des opérateurs par des graphes contextuels et de leur consistance face à une mise en application en situation réelle. Ainsi le prototype ne teste pas les modes conseiller et courtier de SAGIM qui ne posent pas de problèmes particuliers, et se concentre sur le mode archiviste. Il ne teste pas non plus la partie sur les

connaissances du domaine, qui seront traitées par SAGIM comme elles le sont dans un système à base de connaissances classique.

Le prototype est ainsi constitué d'une interface graphique pour la saisie d'incidents et l'adaptation des structures d'activité aux nouvelles pratiques. Le but n'est pas de tester les communications avec l'opérateur, mais de tester le comportement des graphes contextuels. Ainsi il n'est pas nécessaire que l'interface soit optimale au niveau des échanges avec l'utilisateur.

Toutes les connaissances doivent être déclaratives : typologie d'incident, événements incidentels, actions, activités...

4.2 Implémentation

L'incertitude sur la plate-forme finale pour le test du prototype nous a contraint à un système portable. Nous avons donc décidé d'implémenter le prototype en HTML pour l'interface graphique et en C/C++ pour les programmes en tâche de fond. Nous présentons dans cette section les détails de l'implémentation des graphes contextuels et du modèle d'activités, des formats des fichiers d'entrée/sortie puis de l'interface du prototype de SAGIM.

4.2.1 Graphes contextuels et modèle d'activités

Les graphes contextuels et le modèle d'activités sont implémentés en C++. Nous avons ainsi défini 14 classes, dont 5 pour des raisons pratiques d'implémentation.

1. **ObjetSART** : cette classe est la mère de toutes les classes déclarées dans le projet SART, qu'elles concernent le modèle d'activité que nous décrivons ici ou les connaissances du domaine.
2. **Structure** : cette classe virtuelle définit une structure. Elle regroupe notamment la **BaseDeScheme**, la **ClasseVirtuelleDAction** et des structures communes à tout le projet telles que les listes chaînées de diverses natures et leurs maillons.
3. **ClasseVirtuelleDAction** (CVDA) : Cette classe virtuelle définit les éléments communs aux classes **Action** et **Scheme** (qui implémentent les activités) : nom, but, pré-conditions...
4. **Action** : cette classe décrit les actions. Elle hérite de CVDA et ajoute l'attribut de type chaîne de caractères **_realisation** qui décrit simplement la marche à suivre pour la réalisation de l'action.

5. **Scheme** : cette classe décrit les activités. Elle hérite de CVDA et ajoute l'attribut de type **GrapheContextuel** `_realisation` qui décrit le plan de diagnostic/action lié à l'activité modélisée.
6. **ObjetVirtuel** : cette classe est virtuelle et permet de regrouper notamment les classes **GrapheContextuel** et **ObjetGraphe**. Elle est en outre utilisée pour certaines connaissances du domaine.
7. **ObjetGraphe** : cette classe regroupe les différents composants des graphes contextuel. Elle définit les attributs : `_nb_max_precedents`, `_nb_max_suivants`, `_nb_precedents` et `_nb_suivants`, de type entier, et `_precedents` et `suivants`, de type liste de pointeurs vers un **ObjetGraphe**.
8. **ElementAction** : cette classe décrit un élément action d'un graphe contextuel. Elle hérite de **ObjetGraphe** et fixe `_nb_max_precedents` et `_nb_max_suivants` à 1. Elle ajoute de plus un attribut `_action` de type pointeur vers un objet **Action**.
9. **ElementScheme** : cette classe décrit un élément activité d'un graphe contextuel. Elle hérite de **ObjetGraphe** et fixe `_nb_max_precedents` et `_nb_max_suivants` à 1. Elle ajoute de plus un attribut `_scheme` de type pointeur vers un objet **Scheme**.
10. **ElementContextuel** : cette classe décrit un nœud contextuel d'un graphe contextuel. Elle hérite de **ObjetGraphe** et fixe `_nb_max_precedents` à 1. Elle ajoute deux attributs : `_condition`, de type chaîne de caractères, contenant la connaissance contextuelle associée au nœud et `_conditions`, de type liste de chaînes de caractères, contenant la liste des valeurs possibles pour cette connaissance contextuelle considérées dans ce nœud contextuel.
11. **ElementDeRecombinaison** : cette classe décrit un nœud de recombinaison d'un graphe contextuel. Elle hérite de **ObjetGraphe** et fixe `_nb_max_suivants` à 1.
12. **BranchementTemporel** : cette classe décrit un branchement temporel d'un graphe contextuel. Elle hérite de **ObjetGraphe** et fixe `_nb_max_precedents` et `_nb_max_suivants` à 1. Elle ajoute de plus un attribut `_schemes` de type liste de pointeurs vers un objet **Scheme**.
13. **GrapheContextuel** : cette classe décrit les graphes contextuels. Elle contient 5 listes de pointeurs vers des objets respectivement de type **ElementAction**, **ElementScheme**, **ElementContextuel**, **ElementDeRecombinaison** et **BranchementTemporel**. Elle a aussi deux attributs de type pointeur vers un **ObjetGraphe**, qui correspondent respectivement à la `_racine` et au `_puit`. Chaque objet de type **GrapheContextuel** contient l'ensemble des éléments constitutifs du graphe, mais aucune information directe sur la structure : le chaînage des

différents éléments constitutifs est réalisé par les éléments eux-mêmes. Les méthodes associées à la classe **GrapheContextuelle** vérifient toutefois la structure et sont les seules à autoriser la modification des liens entre les éléments.

14. **BaseDeSchemes** : cette classe catalogue toutes les actions et les activités. Elle contient une liste d'objet respectivement de type **Action** et de type **Scheme**. Elle est le point d'entrée de la navigation et contient toutes les connaissances de cette partie du système.

La Figure 6–7 résume les liens d'héritage entre ces 14 classes. Chaque classe est représentée avec ses principaux attributs et leurs types ; les attributs hérités ne sont pas repris.

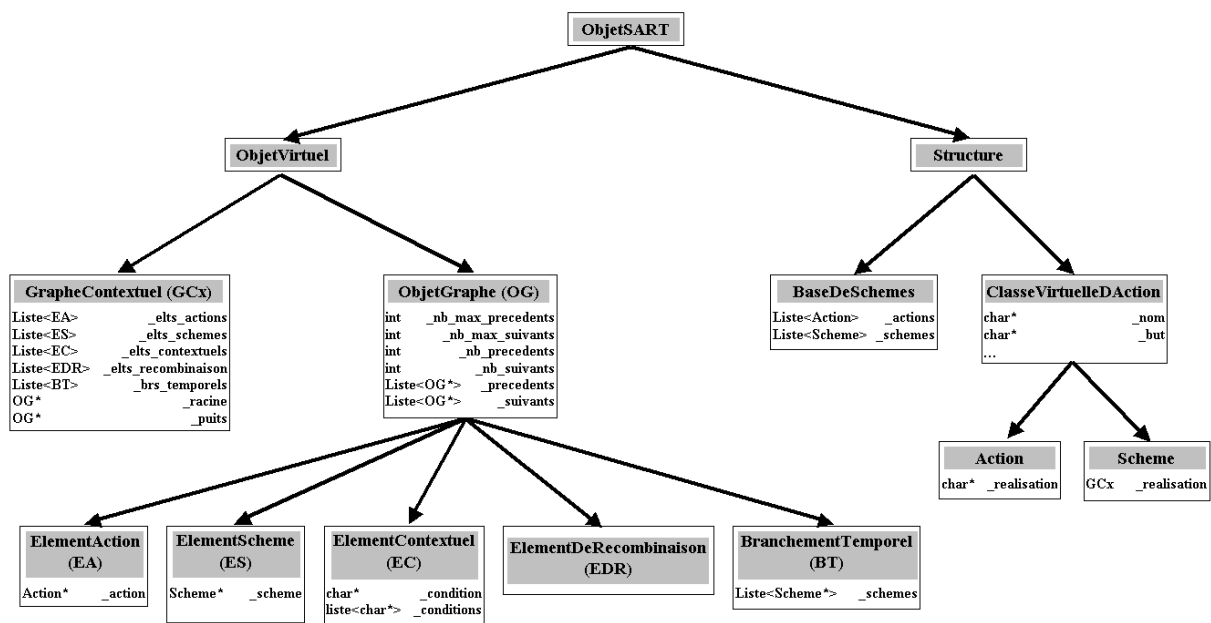


Figure 6–7: Hiérarchie d'héritage des classes du prototype de SAGIM

La Figure 6–8 représente les inclusions des 9 classes principales du prototype de SAGIM. Dans cet organigramme, un lien est établi d'une classe à une autre quand la première contient des attributs dont le type est défini par la seconde classe. Ce lien est indicé par le domaine de valeur du nombre de valence de l'attribut considéré. Par exemple, le lien allant de la classe **Scheme** à la classe **GrapheContextuel** est indicé par **1**, ce qui signifie qu'un objet de type **Scheme** a exactement 1 attribut de type **GrapheContextuel**. De même, le lien entre les classes **BaseDeSchemes** et **Scheme** est indicé par **0..n**, ce qui signifie qu'un objet de type **BaseDeScheme** a un nombre quelconque d'attributs de type **Scheme**. Un lien plein entre une classe X et une classe Y indique qu'un objet de type Y existe physiquement dans un objet de type X ; un lien pointillé indique qu'un objet de type Y n'est que référencé dans un objet de type X (par l'intermédiaire de pointeurs).

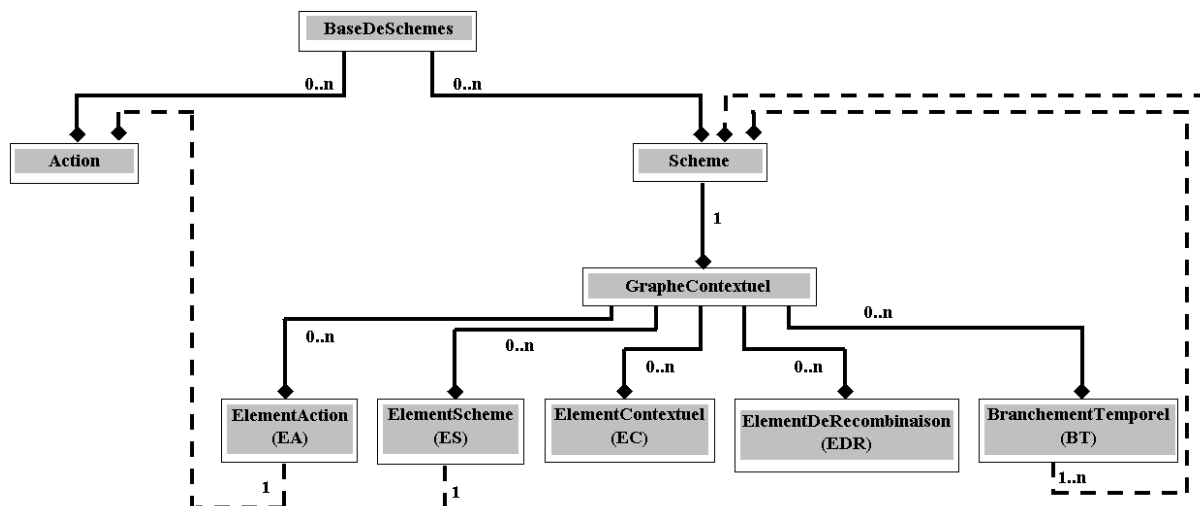


Figure 6-8: Hiérarchie d'inclusion des classes du prototype de SAGIM

4.2.2 Formats des fichiers de données

Quatre types de fichiers de données existent dans le prototype de SAGIM : des listes de données avec leurs paramètres, des fichiers de connaissances contextuelles, des bases d'actions et activités, des rapports d'incident.

4.2.2.1 Listes de données

Premièrement, nous avons défini des simples listes de données. Ces listes sont composées d'items définissant chacun une donnée. Chaque item est composé d'un sigle, d'un nom et d'un ensemble de paramètres de l'item. Chaque paramètre est composé d'un sigle, d'un type de donnée valide pour ce paramètre (et éventuellement la référence à un fichier listant les données admises pour ce paramètre) et un texte pour l'affichage.

L'exemple suivant décrit la donnée relative à l'événement incidentel « actionnement d'un signal d'alarme ». La première ligne définit d'abord le sigle (KSA) puis le nom de la donnée. Chacune des lignes suivantes décrit un des paramètres. Le premier paramètre, dont le sigle est défini par **\$Station\$**, correspond à la station à laquelle le train est arrêté (quand un signal d'alarme est tiré, le train s'arrête à la station suivante). Le champ **Liste(\$Ligne\$/Stations.lst)** indique que les valeurs admises pour ce paramètre sont décrites dans le fichier **Stations.lst** du répertoire correspondant à la ligne en cause.

KSA	{Actionnement d'un commutateur de signal d'alarme (KSA)}
\$Station\$	Liste(\$Ligne\$/Stations.lst) {Station}
\$Voie\$	Liste("Voie 1", "Voie 2", "Boucle") {Voie}
\$NumRame\$	Texte(2) {Numéro d'exploitation de la rame}
\$NumTrain\$	Texte(3) {Numéro de matériel du train}

```

$NumVoit$      Texte(1)      {Position de la voiture dans la rame}
$IdentVoit$    Texte(5)      {Numéro d'identification matériel de la
voiture}

```

Ce premier type de fichier est le plus utilisé dans notre prototype. Il définit les données relatives aux événements incidentels, aux incidents, aux actions locales et globales, aux contextes, aux types de lieux et toutes les données décrivant une ligne (stations, interstations, terminus, sections, sous-sections, signaux d'espacement, signaux de manœuvre, rupteurs, type de matériel roulant...).

4.2.2.2 *Connaissances contextuelles*

Le second type de fichier de données décrit les connaissances contextuelles. Chaque fichier correspond à une connaissance précise ; il définit les différentes valeurs connues de cette connaissance contextuelle et la matrice comparant les différentes valeurs possibles pour le calcul de la mesure de similarité. L'exemple suivant décrit la connaissance contextuelle **JourSemaine**. On peut remarquer au passage que la comparaison n'est pas symétrique : ainsi un lundi est moins proche d'un dimanche que le dimanche ne l'est du lundi (ceci est dû au fait que les solutions appliquées le lundi sont plus facilement applicable un dimanche que le contraire).

```

7
lundi
mardi
mercredi
jeudi
vendredi
samedi
dimanche
100 90 75 80 75 70 50
90 100 80 85 75 70 50
75 80 100 80 75 80 60
75 80 80 100 85 70 50
70 75 75 80 100 70 50
75 75 75 75 80 100 75
60 60 60 70 60 80 100

```

4.2.2.3 *Base d'actions et d'activités*

Le troisième type de fichiers contient les descripteurs d'action et d'activité et les graphes contextuels correspondant à ces activités. Ce fichier est composé de trois sections. La première définit les actions élémentaires. Celles-ci sont enregistrées sous la forme

```
{Sigle|Nom de l'action|Description de la mise en oeuvre}
```


La seconde définit les caractéristiques des activités. Elles reprennent la première partie de l'enregistrement des actions et se présentent sous la forme

{Sigle|Nom de l'activité}

La troisième définit les graphes contextuels correspondant aux activités. Ils sont en nombre égal aux activités définies dans la seconde section et sont définis dans le même ordre. Ils respectent la mise en forme suivante :

Un *graphe contextuel* est composé de série d'actions, d'activités et de sous-graphes. Le graphe contextuel est encadré par des accolades {...}.

Les *actions* et les *activités* sont référencées par leur sigle. Chaque élément de la série est séparé par un pipe |. Ainsi le graphe contextuel enregistré sous la forme :

{A2|A4|A13|A5|A3}

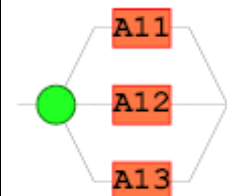
correspond au graphe :

A2 A4 A13 A5 A3

Les *nœuds contextuels* sont définis par {\$Cxt i|Cxt i.1{...}|...|Cxt i.n{}}}, où Cxt i correspond à l'élément contextuel impliqué et les Cxt i.1 à Cxt i.n sont les valeurs admises. Chaque valeur admise est immédiatement suivie du sous-graphe contextuel correspondant au plan de diagnostic/action adapté à cette situation. Ainsi le graphe contextuel enregistré sous la forme :

{{ \$Position par rapport a la station|a quai{A11}|partiellement a quai{A12}|en interstation{A13}}}

correspond au graphe :



Les *branchements temporels* sont sous la forme {**Activité_1**|...|**Activité_n**}, où **Activité_1**, ..., **Activité_n** sont les différentes activités à mener en parallèle. Ainsi le graphe contextuel enregistré sous la forme :

{A20 | {EvaRamAvar | EvaRamSec} | FairConv}

correspond au graphe :

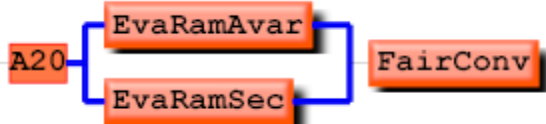


Tableau 6-1: Format d'enregistrement des graphes contextuels

Deux fichiers sont sous ce format, l'un décrivant les procédures, l'autre décrivant les stratégies réellement appliquées. Ce second fichier est initialisé avec les procédures puis évolue en intégrant les pratiques des opérateurs. L'annexe 2 montre le fichier correspondant à l'exemple que nous avons traité dans le troisième chapitre.

4.2.2.4 Fichier rapport d'incident

Chaque incident est enregistré dans un fichier contenant les descripteurs de cet incident : type, lieu, cause, contexte, pratique mise en œuvre pour gérer cet incident, actions entreprises pour gérer le trafic résiduel et conséquences sur le trafic. Chaque descripteur est enregistré sur une ligne de format

Descripteur {**SigleValeur**|**Parametre_1**|...|**Parametre_n**}.

Par exemple, la ligne

Evenement0 {STA|GNO|Voie 1|20|123}

indique qu'un événement incidentel de type stationnement s'est manifesté à Gare du Nord, voie 1, impliquant la rame de numéro d'exploitation 20, numéro de matériel 123.

Le nom du fichier indique la date, l'heure et la cause de l'incident. Par exemple le fichier dénommé **2001.12.19-11.20.23-AM.NDT.inc** contient les informations sur un incident survenu le 19 décembre 2001 à 11 heure 20 minutes et 23 secondes, causé par un

non-déblocage total d'une rame¹⁶. Voici le contenu du fichier rapport d'incident le concernant.

```
Ligne                {Ligne4}
Date                 {2001.12.19}
HDeb                 {11.20.23}
EtendueIncident     {Sta|GNO}
Evenement0           {STA|GNO|Voie 1|20|123}
Cause0              {AM.NDT|20|123}
IntervaleReel       {150}
IntervaleTheorique  {120}
RepercussionsTrafic {Pert|11.20.00|16|24}
Contexte0            {11.21.00|Réinitialisation possible|non}
Contexte1            {11.21.00|Possibilité de reprise de marche|oui}
Contexte2            {11.21.00|T4 avant le terminus|oui}
Contexte3            {11.30.00|Train réinitialisé|oui}
AL0                  {RepSerVoyT4|11.22.00|20}
AL1                  {RepSerVoy|11.30.00|20}
CommentaireActionsLocales {Le train ayant pu être réinitialisé en route,
celui-ci repart avec voyageur.}
NbAG                 {4}
AG0                  {DSO1|11.20.00|16|CHE|Voie1}
AG1                  {DSO1|11.20.30|24|BAR|Voie1}
AG2                  {DSO2|11.22.00|16|CHE|Voie1}
AG3                  {DSO2|11.23.00|24|BAR|Voie1}
CommentaireActionsGlobales {}
HFin                 {11.22.00}
RetardCumule        {2}
GrandIntervalle     {130}
```

Plusieurs exemples de fichiers d'incident sont présentés dans l'annexe 3.

4.2.3 Organisation des fichiers de données

L'ensemble des fichiers de données du prototype de SAGIM est organisé dans une arborescence de répertoire détaillée dans la Figure 6–9.

¹⁶ Pour des raisons de confidentialité, les fichiers sensibles ne sont pas donnés. Ainsi les fichiers de données sensibles ne seront pas présentés et les seuls fichiers rapport d'incident que nous insérons concernent des faux incidents.

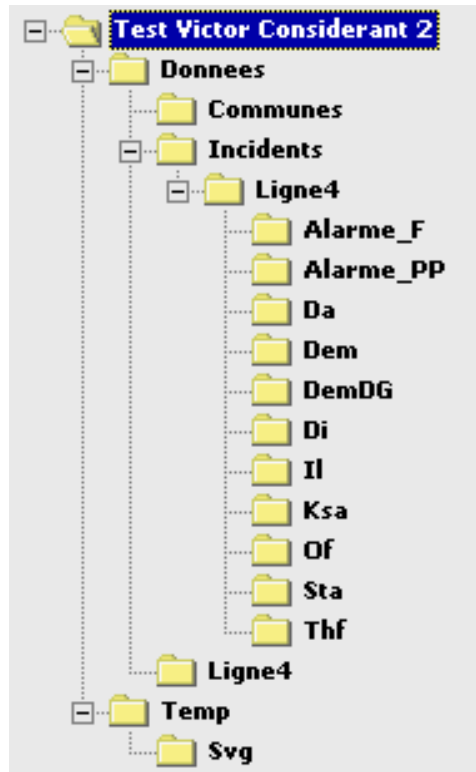


Figure 6-9: Arborescence du répertoire du prototype de SAGIM

La racine contient les exécutables et deux répertoires. Le premier répertoire (**Donnees**) contient les données, le second (**Temp**) est un répertoire de travail temporaire. Le répertoire de données est organisé en trois sous-répertoires : les données communes à toutes les lignes (**Communes**), les données spécifiques à la ligne concernée (**Ligne4**) et les données sur les incidents (**Incidents**). Le répertoire de la ligne contient les données spécifiques qui décrivent la ligne (stations, interstations, sections, sous-sections...) ; le répertoire de données communes contient les fichiers décrivant les autres données (événements incidentels, typologie d'incidents, actions locales et globales, description des activités, connaissances contextuelles...). Le répertoire **Incidents** contient les fichiers de rapport qui sont créés pour chaque incident. Ce dernier répertoire est lui-même décomposé en deux niveaux de sous-répertoires : le premier concernant la ligne, le second suivant l'événement incidentel. Le répertoire incident correspond à la base historique des incidents.

4.2.4 Interface du prototype

Le prototype de SAGIM est construit autour du modèle CGI : une première page fait appel à un programme en lui fournissant les valeurs des champs d'un formulaire, et ce programme construit la page suivante en fonction des valeurs reçues. Nous obtenons ainsi une suite de formulaires ciblés vérifiant les spécifications ci-dessus. Les programmes construisent la page suivante de saisie en s'appuyant sur les données transmises par le formulaire précédent et les

connaissances déclaratives enregistrées dans différents fichiers. Chaque programme effectue une part de la tâche du prototype et construit un formulaire quand des connaissances doivent être saisies par l'utilisateur.

La première page permet de recueillir les premières informations sur l'incident (date, heure, type de lieu, événement(s) incidentel(s), cause de l'incident et répercussions sur le trafic). Ces données sont enregistrées dans la fiche d'incident qui sera ensuite complétée à chaque étape. La seconde page permet de renseigner les paramètres des éléments renseignés dans le premier formulaire. L'algorithme d'acquisition de connaissances stratégique entre alors en jeu pour à la fois servir de support au renseignement de la séquence d'actions mises en œuvre et permettre l'adaptation des activités concernées par cette pratique.

L'algorithme d'acquisition incrémentale de connaissances stratégique est découpé en trois parties pour chaque activité : premièrement la sélection de la stratégie qui aurait dû être appliquée (par sélection du chemin d'action suivi), puis la définition de la pratique correspondante à partir de cette stratégie, et enfin, si nécessaire, l'insertion d'une divergence entre la stratégie et la pratique. Ces trois phases requièrent des connaissances particulières. La première recueille les valeurs courantes pour des connaissances contextuelles enregistrées dans le graphe contextuel, afin de sélectionner le chemin correspondant à la situation courante. La seconde demande à l'utilisateur quelles actions ou activités ont été réalisées parmi celles constituant la stratégie, puis demande les actions et activités ayant été réalisées en plus de celles-ci. La troisième n'est appliquée qu'en cas de divergence entre la stratégie et la pratique et permet la saisie de la connaissance contextuelle supportant cette divergence. Cette interaction est répétée pour toutes les activités mises en jeu.

Quand la pratique est saisie et introduite dans les activités, le prototype demande à l'utilisateur de compléter les paramètres des différentes actions. Ce formulaire présente alors les actions dans un ordre chronologique et représente également la structure des activités. Une fois ces informations saisies, l'utilisateur doit renseigner les actions globales mises en œuvre (qui correspondent aux actions concernant le trafic résiduel) puis compléter leurs paramètres. Enfin un dernier formulaire permet de recueillir les conséquences de cet incident sur l'état du trafic et sur les retards éventuels causés par celui-ci.

La Figure 6–10 présente un exemple de l'interface graphique. Cette image correspond à la phase de saisie de l'élément contextuel ayant conduit à une nouvelle pratique pour l'activité « Traiter l'incident : Avarie au matériel roulant – Dégagement de fumée – Élément d'une voiture en ignition ». Un exemple complet, en images, d'utilisation de SAGIM pour la saisie d'un incident est donné dans l'annexe 4.

Adresse [http://versa/SAGIM/Programme/Archiviste/Prototype/Test%20Victor%](http://versa/SAGIM/Programme/Archiviste/Prototype/Test%20Victor%20)

RATP
SART / SAGIM / Archiviste

**Action "Traiter l'incident : Avarie au matériel roulant - Dégagement de fumée - Element d'une voiture en ignition"
dans "THF"**

Contexte sélectionné Précédemment

Visite complémentaire nécessaire	non
Element en ignition	oui
Extincteur disponible	non
Incident clos	non
Service voyageur possible	non

Particularités du contexte actuel

Table n°1

Pour quelle raison aucune action n'a été réalisée, à la place de l'action "Remise sous tension sauf une ou plusieurs sous-section(s)", entre l'action "Couper le courant sur la section" et l'action "Visite d'installations" ?

Elément de la situation à analyser :	<input type="text"/>
Valeur de cet élément de situation dans le cas présent :	<input type="text"/>
Valeur habituelle de cet élément de situation :	<input type="text"/>

OK

Figure 6–10: Exemple d'une page de saisie de SAGIM

L'interface présente régulièrement à l'utilisateur le graphe contextuel de l'activité en cours de développement. Cette présentation est réalisée par un *plug-in* d'affichage des SVG (*Scalable Vectorial Graphics*, <http://www.w3.org/TR/SVG/>). Ce format est simple d'utilisation et autorise un redimensionnement de l'affichage en fonction des goûts de l'utilisateur.

4.3 Résultats obtenus

Le prototype de SAGIM est en service en situation réelle depuis janvier 2002. Les deux figures suivantes présentent l'évolution des trois bases de données principales de SAGIM soumises à évolution : d'abord la base des stratégies, qui contient les actions et les activités

déclarées dans le système (Figure 6–11), puis l’historique d’incidents et la base d’incidents typiques (Figure 6–12).

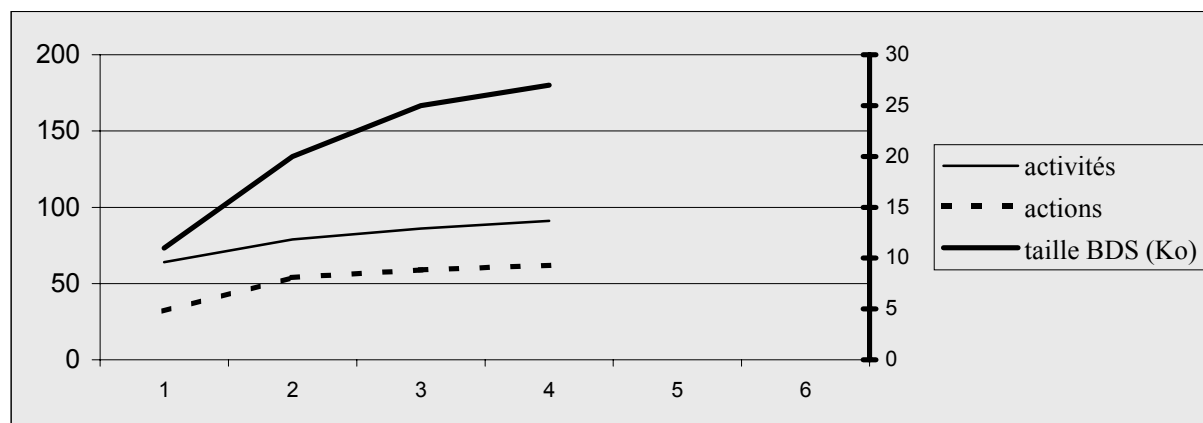


Figure 6–11: Evolution de la base de stratégies de SAGIM

Cette première figure représente l’évolution de la base de stratégies. Cette base est initialisée avec les procédures puis évolue avec l’acquisition de l’expertise des opérateurs. Cette évolution est de deux types. Premièrement, des actions et des activités peuvent être ajoutées pour élargir le champ d’opération des opérateurs. C’est ce que nous observons avec les deux courbes inférieures, qui représentent le nombre d’actions et d’activités enregistrées par mois (le nombre est indexé suivant le premier axe, à gauche). Le nombre d’actions et d’activités croît de moins en moins avec le temps, mais ne se stabilise pas tout à fait. Les actions se stabilisent rapidement pour prendre en compte des actions non prévues par la procédure. L’émergence de nouvelles actions ou d’activités simples est beaucoup plus rare. Régulièrement de nouvelles activités peuvent apparaître pour gérer des incidents nouveaux (second niveau d’activité). Les activités de plus haut niveau ne sont ajoutées que quand les systèmes d’assistance (systèmes d’alarme) sont ajoutés à l’environnement de travail des opérateurs (et est donc exceptionnel). Ainsi les courbes d’évolution du nombre d’actions et d’activités se stabilisent rapidement et n’évoluent guère après un temps d’adaptation des procédures aux pratiques. La seconde évolution présentée par le graphique correspond à la taille globale de la base de stratégies. Nous observons que la base de stratégies évolue plus que le nombre d’activités et d’actions qu’elle contient. Ce phénomène est normal puisque les activités évoluent sans cesse pour accumuler les différentes stratégies pouvant être mises en œuvre. Cette évolution présente deux phases : premièrement une phase d’extension rapide qui est due à la fois à l’augmentation du nombre d’actions et d’activité et à l’adaptation des procédures aux pratiques courantes ; deuxièmement, une phase d’évolution plus lente qui correspond principalement à l’acquisition de nouvelles pratiques pour les activités enregistrées dans la base. Cette évolution reste toutefois considérable, ce qui prouve le bien fondé de SAGIM quant à son rôle de mémoire d’entreprise.

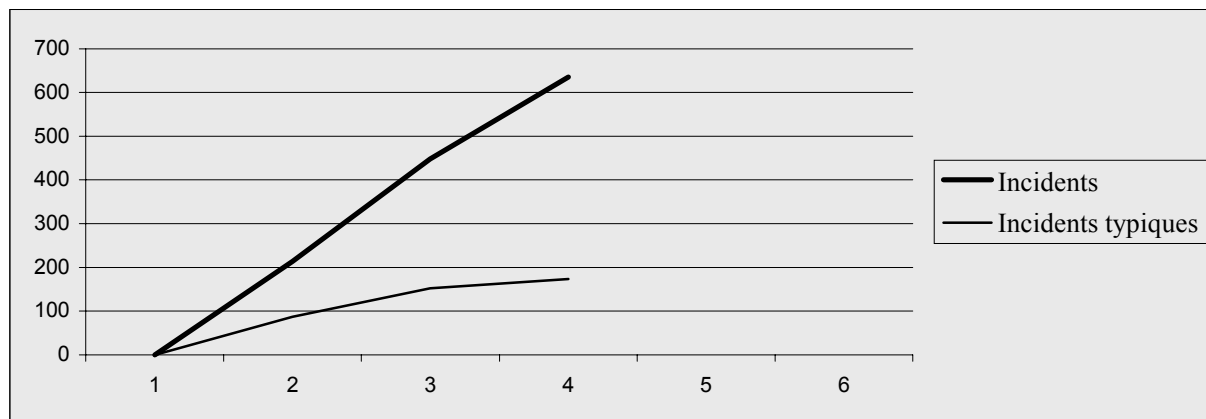


Figure 6-12: Evolution des bases d'incidents de SAGIM

Le second graphique présente l'évolution des deux bases d'incidents. La courbe supérieure correspond à l'historique des incidents et contient tous les incidents rencontrés. Sa croissance est quasiment linéaire, car il y a un nombre quasi-constant d'incidents par mois. La courbe inférieure représente l'évolution de la base d'incidents typiques. Cette base évolue de moins en moins, mais ne se stabilise pas : il existe toujours un certain nombre d'incidents originaux soit par leur contexte d'occurrence, soit par la stratégie appliquée.

D'une manière qualitative, ce test a confirmé la bonne appréciation par les opérateurs de la modélisation à base d'activités par les graphes contextuels. La quantité d'incidents enregistrés (environ 1000 en 5 mois) et le bon comportement de l'algorithme d'acquisition incrémentale de connaissances stratégiques sont prometteurs pour l'avenir de SAGIM : l'ensemble des activités et des graphes contextuels associés ont été parfaitement traités par le prototype.

4.4 Capacités actuelles et potentielles

Bien que s'adressant aux chefs de régulation, SAGIM constitue des bases de connaissances pouvant être utiles à plusieurs services de l'entreprise. Le Tableau 6-2 présente les fonctionnalités offertes actuellement aux utilisateurs par SAGIM.

<p><i>Navigation libre dans les structures d'activité</i></p>	<p>Cette navigation permet aux utilisateurs d'accéder à la stratégie construite par le système sur la base des procédures et des pratiques réelles, offrant ainsi à l'opérateur un support à l'analyse de ses propres stratégies et l'enrichissement de ses connaissances personnelles.</p>
---	---

<p><i>Partage de l'expérience</i></p>	<p>Les pratiques des différents opérateurs se partageant un poste sont enregistrées et structurées dans une unique base d'actions/activités. Ainsi les stratégies accessibles sont issues des pratiques de plusieurs opérateurs. Ceci permet un premier niveau de partage des connaissances stratégiques entre les opérateurs.</p>
<p><i>Support à la discussion constructive entre opérateurs</i></p>	<p>Le partage précédent peut être enrichi par une discussion entre les différents opérateurs concernés. Les stratégies enregistrées par le système intègrent les pratiques des différents opérateurs et chacun d'eux peut défendre ses opinions face à ses collègues. Les discussions de ce type font généralement ressortir un consensus sur les stratégies, liant ainsi les opérateurs en une communauté de pratique.</p>
<p><i>Support à la discussion entre les opérateurs et l'encadrement</i></p>	<p>Ces discussions peuvent également intégrer l'encadrement des opérateurs pour diriger plus fortement les stratégies et faire ressortir les risques de déviance de certaines pratiques des opérateurs. SAGIM offre ainsi une aide au retour d'expérience tel qu'il est actuellement pratiqué.</p>
<p><i>Accès simplifié aux enregistrements des incidents</i></p>	<p>La base d'incidents est simple d'utilisation : chaque incident est placé dans un fichier permettant ainsi un accès direct. Les informations qui sont enregistrées dans ce fichier sont structurées et mettent en avant la structure de la pratique mise en œuvre grâce à « l'emboîtement » des activités et au contexte ayant guidé les choix stratégiques. Ainsi une trace du raisonnement mis en application est conservée avec l'incident.</p>
<p><i>Différentes classifications des incidents</i></p>	<p>La base d'incident permet également d'obtenir une classification en fonction de différents critères : lieu, type, cause, mais aussi les éléments du contexte. Ceci permet de faire ressortir des dépendance statistiques entre les incidents et leurs caractéristiques.</p>

<p><i>Maintenance, par un « super-utilisateur »</i></p>	<p>Cette maintenance concerne les actions, les activités et les graphes contextuels représentant les stratégies. Cette opération de maintenance sert à conserver une cohérence dans la base d'actions/activités afin de simplifier l'analyse des stratégies.</p>
<p><i>Flexibilité du système face à des innovations technologiques</i></p>	<p>Le super-utilisateur doit simplement ajouter les actions et activités induites par ces changements dans l'environnement de travail des opérateurs pour que SAGIM les intègre.</p>

Tableau 6-2: Capacités actuelles de SAGIM

Les résultats obtenus et les discussions avec les opérateurs autour du prototype en opération ont fait ressortir différentes évolutions possibles de SAGIM. Ainsi nous pouvons établir comme objectif à moyen terme les propositions de fonctionnalités décrites dans le Tableau 6-3. Plusieurs de ces fonctionnalités s'adressent à d'autres utilisateurs que les chefs de régulation, notamment à ceux de services responsables de l'édition des procédures ou de la maintenance du matériel et des installations.

<p><i>Formation initiale des opérateurs</i></p>	<p>SAGIM peut aider à la formation des opérateurs sur plusieurs points. Premièrement, ce système met en avant l'organisation des diverses activités liées à la gestion des incidents. Cette organisation permet aux opérateurs de mieux structurer leurs raisonnements. Deuxièmement, pour chaque activité, SAGIM montre la dualité procédure/pratique et propose des stratégies plus complexes intégrant une grande variété d'éléments contextuels. Les graphes contextuels montrent ainsi comment les pratiques sont obtenues à partir des procédures. Ceci est important pour les futurs opérateurs, car cette adaptation du prescrit au réel est la base de leur nouveau métier.</p>
--	--

<p><i>Support de discussion entre un expert et un novice autour d'incidents complexes</i></p>	<p>Ceci est une extension des discussions entre collègues spécialisée dans l'accompagnement de nouveaux opérateurs. Actuellement la formation initiale des chefs de régulation se termine par un mois en situation réelle d'opération assisté par un chef de régulation expérimenté. SAGIM peut faciliter le transfert d'expertise en proposant l'analyse conjointe d'incidents de plus en plus complexes par les chefs de régulation expert et novice.</p>
<p><i>Recherche d'une stratégie optimale par application de pratiques généralisées</i></p>	<p>Cette recherche consiste en l'analyse des stratégies obtenues pour chaque activité et l'identification d'une pratique pouvant être appliquée dans tous les cas. Cette fonctionnalité est liée à la nature particulière de la gestion d'incidents sur une ligne de métro : chaque incident peut être géré de différentes manières et la méthode appliquée importe peu, du moment qu'elle respecte les consignes de sécurité. Aussi les stratégies obtenues par agrégation des différentes pratiques peuvent être épurées pour faire ressortir une stratégie donnant de bons résultats quel que soit le contexte de l'incident.</p>
<p><i>Enrichissement des procédures</i></p>	<p>Cette fonctionnalité permet d'accélérer la rétroaction des pratiques sur les procédures. Elle correspond à une recherche de stratégie optimale guidée par des super-utilisateurs.</p>
<p><i>Test d'hypothèses pour la gestion d'incidents</i> (conjointement avec la simulation)</p>	<p>Ceci consiste en la construction de stratégies et de la simulation de l'application de ces stratégies sur des incidents. La simulation pourrait être basée sur des scénarios construits à partir des incidents réellement rencontrés, afin de tester la validité de la stratégie proposée.</p>

<p style="text-align: center;"><i>Aide à la maintenance préventive</i></p>	<p>Les incidents répertoriés peuvent être mis en relation avec les installations et les matériels roulant impliqués. SAGIM pourrait détecter toute corrélation entre des incidents à répétition et des installations ou matériel, mettant alors en avant les risques liés à ces objets. Les éléments ainsi mis en exergue seraient alors plus particulièrement surveillés lors d'incidents à venir, voire insérés dans les plans de maintenance préventive.</p>
<p style="text-align: center;"><i>Maintien d'une communauté de pratique malgré la répartition des opérateurs</i></p>	<p>SAGIM permet de partager des connaissances entre les opérateurs sur un même poste, mais également sur des postes différents. Les connaissances stratégiques issues des pratiques observées sur une ligne peuvent être mises à disposition des systèmes fonctionnant sur d'autres lignes à des fins d'analyse. SAGIM peut ainsi pallier la suppression de la formation par friction actuellement rencontrée au PCC du métro parisien telle que nous l'avons décrite dans le second chapitre (section 4.1, page 81).</p>

Tableau 6-3: Capacités potentielles de SAGIM

Section 5 :

Conclusion

Le projet franco-brésilien SART vise le développement d'un Système d'Aide Intelligent en Contexte (SAIC) pour la gestion du trafic sur une ligne de métro. Le système proposé s'articule autour d'une architecture multi-expert. Il se compose de quatre agents : communication, configurateur de ligne, simulateur de trafic et gestionnaire d'incident. Les trois derniers agents communiquent entre eux et avec l'utilisateur par le biais de l'agent de communication, qui joue le double rôle de plate-forme d'intégration des agents informatiques et d'interface.

Les travaux présentés dans cette thèse concernent plus particulièrement l'agent gestionnaire d'incidents, aussi appelé Système d'Aide à la Gestion d'incidents dans le Métro (SAGIM). SAGIM a pour rôle d'accumuler et organiser les connaissances des experts relatives à la gestion d'incident, afin de pouvoir aider un opérateur pour un incident en cours

ou proposer des sessions de retour d'expérience ciblées. Pour cela, nous proposons de concevoir SAGIM, lui aussi, comme un SAIC.

Le rôle de SAGIM est triple : cet agent doit recueillir et organiser les connaissances des experts, aider un opérateur en temps réel et proposer une exploration libre des connaissances accumulées et structurées. Il propose ainsi trois modes d'utilisation (archiviste, conseiller et courtier) qui correspondent respectivement à ses trois rôles.

L'implémentation de SAGIM est construite autour de deux modes de raisonnement artificiels proches de raisonnements naturels pour les opérateurs. Le premier raisonnement proposé est basé sur la structuration des activités des opérateurs par des graphes contextuels, (présentée dans les troisième et cinquième chapitres). Le second est un raisonnement à partir de cas adapté à la prise en compte du contexte et à des bases de cas volumineuses et redondantes.

Ces deux modes de raisonnement ne sont pas indépendants l'un de l'autre et partagent notamment leurs connaissances sur les connaissances contextuelles, sur les incidents et sur les événements incidentels. Ainsi l'hybridation proposée par SAGIM ne se contente pas d'une juxtaposition des raisonnements proposés, mais d'une réelle connivence entre eux, permettant ainsi de produire des connaissances plus pertinentes.

SAGIM est actuellement en cours d'implémentation. Un prototype de SAGIM a été développé pour tester la validité des modèles proposés, notamment la structuration des activités par des graphes contextuels et l'algorithme d'acquisition incrémentale de connaissances stratégiques associé à ce modèle. Le test en conditions réelles (pendant 5 mois) de ce prototype confirme la bonne assimilation du modèle par les opérateurs et le bon comportement du module de raisonnement basé sur les graphes contextuels face à une utilisation intensive.

Ces résultats encourageant nous ont permis d'entrer dans une phase de développement plus approfondie de SAGIM. Une première version du programme est en cours de développement à Bangkok Le programme résultant devrait avoir une portée beaucoup plus large que celle de notre prototype en renforçant encore l'aspect déclaratif des connaissances manipulées et la généralité du moteur et de l'interface. Par ailleurs, un prototype a été développé pour porter SAGIM vers le RER et mis en interaction avec le prototype d'un agent de régulation de SART pour le RER.

Conclusion générale

Une ligne de chemin de fer métropolitain est un système dynamique très complexe. De nombreux incidents d'exploitations se produisent chaque jour. La quantité d'incidents traités depuis le début de l'exploitation du métro parisien est à l'origine de la grande expérience acquise par l'entreprise sur la gestion d'incidents. Cette expérience acquise par l'entreprise sur la gestion d'incident est traduite en procédures qui donnent un cadre au diagnostic et à l'action devant être réalisés pour gérer un incident ou un état particulier du système (constitué notamment de la ligne et des trains en ligne). Ces procédures sont des formes compilées des connaissances impliquées dans la gestion des incidents, mais sont trop générales pour être directement applicables à une situation particulière. Les chefs de régulation adaptent ces procédures à la situation réelle pour chaque incident pour prendre en compte les particularités du contexte. Les pratiques qu'ils mettent ainsi en œuvre sont différentes, pour un même incident, d'une situation à une autre et d'un opérateur à un autre.

L'adaptation des procédures par les chefs de régulation est rendue possible grâce à leur grande expérience du terrain et à l'expérience acquise en tant que chef de régulation. Cette dernière est issue soit des situations traitées directement par le chef de régulation, soit par les échanges entre collègues sur des situations particulières. Les chefs de régulation de toutes les lignes de métro étant rassemblés dans un même lieu de travail, les informations sur des situations particulières ou sur des stratégies originales sont rapidement partagées par tous. Les chefs de régulation forment ainsi une communauté de pratique. A l'heure où les PCC tendent à être décentralisés, cette communauté est mise en péril. Les échanges entre chefs de régulation de lignes différentes ne seront pas aussi efficaces que précédemment. Nous risquons de voir apparaître autant de communauté que de ligne de métro. Pourtant une grande part de leurs expériences est utile à l'ensemble de la communauté, bien que chaque ligne ait ses spécificités. Le développement d'outils de *knowledge management* sur la gestion d'incident devient nécessaire.

Le projet SART est un outil pouvant servir de liaison entre les différents postes de contrôle permettant ainsi de conserver une unité de la communauté de pratique. Ce projet vise la

réalisation d'un système d'aide pour la régulation du trafic. Nous nous intéressons plus particulièrement à un de ses composants : le gestionnaire d'incidents, également appelé SAGIM (pour Système d'Aide à la Gestion d'Incidents dans le Métro). SAGIM est un Système d'Aide Intelligent en Contexte (SAIC) dont le but est d'accumuler des connaissances sur la gestion d'incident pratiquée par les chefs de régulation, de les organiser, puis de les utiliser soit pour aider un chef de régulation pour traiter un incident en cours, soit pour proposer des sessions de retour d'expérience.

L'architecture SAIC est intermédiaire entre les Systèmes Interactifs d'Aide à la Décision et les Systèmes d'Assistance Intelligents. Des premiers, ils héritent la conception autour de Systèmes à Base de Connaissances (ou Systèmes Experts de deuxième génération). Les seconds apportent une vision plus globale des interactions système/utilisateur, donnant un rôle d'assistant au système. Ainsi le système est en charge de tâches automatisables de surveillance et de conseil. Ces deux types de systèmes partagent le besoin d'un modèle de l'utilisateur afin de cibler correctement l'aide ou l'assistance proposée. Les SAIC ont donc également besoin de tels modèles.

En outre, les SAIC placent le contexte au cœur de leur fonctionnement. Le contexte joue, en effet, un rôle prépondérant dans de nombreux raisonnements humains, il est donc nécessaire de l'introduire dans les systèmes d'aide à des tâches fortement dépendantes de la situation. L'architecture SAIC reprend une grande part de celle des Systèmes à Base de Connaissances : des connaissances sur les raisonnements permettent au système de manipuler des connaissances sur le domaine d'application afin de proposer des solutions à un problème posé. L'introduction du contexte intervient à tous les niveaux du développement du système : définition des connaissances du domaine, des connaissances de raisonnement, des modèles de l'utilisateur... Des travaux antérieurs ont introduit le contexte dans les connaissances du domaine. Dans cette thèse, nous introduisons le contexte dans les modèles de raisonnement.

SAGIM est ainsi construit sur deux modes de raisonnements basés sur le contexte. Premièrement, nous utilisons un modèle informatique de raisonnement par analogie : le Raisonnement à Partir de Cas. Ce type de raisonnement intègre naturellement le contexte et ses implications sur les solutions proposées. Les travaux de Mignot [1997] sur les mesures de similarité dépendantes du contexte sont exploités dans SAGIM. Nous introduisons toutefois une originalité par rapport à ce raisonnement artificiel : Le nombre de situations incidentelles rencontrées est considérable dans le métro : la plupart ne diffèrent que par des détails sans conséquences sur leur gestion, d'autres, plus rares, sont originaux et certains incidents sont exceptionnels (voir uniques). Nous rencontrons ainsi de nombreux incidents n'apportant aucune nouveauté (la très grande majorité des incidents) et quelquefois des incidents originaux. Nous traitons donc dans SAGIM deux bases d'incidents : premièrement une base

historique des incidents qui contient tous les incidents rencontrés (pour des fins statistiques) et une base de cas typiques qui contient tous les incidents ayant introduit une originalité (pour servir de base au Raisonnement à Partir de Cas).

Le deuxième mode de raisonnement utilisé par SAGIM est basé sur la modélisation des activités des chefs de régulation par le biais de graphes contextuels. Les activités correspondent à la mise en œuvre pratique de raisonnements appliqués à la planification du diagnostic et des actions visant à la réalisation d'une tâche particulière. De nombreuses études (en psychologie cognitive et en intelligence artificielle) décomposent les tâches en sous-tâches et organisent différentes méthodes possibles pour la réalisation de chaque tâche. Ceci permet d'obtenir une hiérarchie de tâche et sous-tâches plus ou moins adaptée à la situation rencontrée. Ces décompositions ne prennent toutefois pas en compte l'activité des opérateurs liée à chaque tâche. Cette activité n'est pas, comme ceci est supposé dans les décompositions classiques, composée d'un diagnostic suivi de plans d'actions, mais présente un enchevêtrement de ces deux composants. Les modèles classiques ne proposent pas de solution à la représentation d'une telle structure de diagnostic/action.

Nous proposons le modèle de graphes contextuels pour représenter de telles structures où les décisions et les actions sont intimement liées. Ces graphes sont composés de nœuds de décision (où un élément du contexte est analysé pour choisir entre différentes alternatives, d'où leur nom de nœuds contextuels), d'éléments action (qui font référence à une action à réaliser) et d'éléments d'activité (qui font référence à une activité secondaire). Ces éléments des graphes contextuels sont liés par des arcs orientés représentant une dépendance temporelle : un élément précède un autre quand le traitement du premier doit être terminé avant le traitement du second. Des branchements temporels sont également ajoutés pour indiquer quand des sous-structures de décision/action sont temporellement indépendantes et peuvent être réalisées en parallèle. Les graphes contextuels représentent ainsi l'évolution de la situation et des connaissances que nous avons sur celle-ci en fonction de l'état d'avancement de la réalisation de la tâche correspondante et des choix réalisés précédemment.

Les graphes contextuels ont une unique racine représentant le point de départ avant tout diagnostic. Tant qu'aucun diagnostic n'est réalisé, les situations ne sont pas différenciées et le traitement est identique quelle que soit la situation. Les graphes contextuels ont un unique point de sortie car les différents chemins d'action contenus dans le graphe représentent tous une manière d'atteindre un but commun et donc d'obtenir une situation donnée. La structure des graphes contextuels présente ainsi une série de fuseaux : chaque fuseau représente un raisonnement contextuel local où des connaissances contextuelles sont mises en avant pour sélectionner un chemin d'actions. Les connaissances contextuelles identifiées représentent des

éléments de diagnostic sur la situation. En fonction de ces diagnostics, des choix de plans de diagnostic/action spécifiques sont réalisés.

Les graphes contextuels sont ainsi des modèles de représentation des structures cognitives de diagnostic/action et permettent de structurer les activités de ce type observables chez les êtres humains. Ces activités, manifestation de la réalisation de tâches, sont, elles aussi, organisées de manière hiérarchique. Nous retrouvons donc les principes de la décomposition de tâches en sous-tâches. Toutefois, nous ne considérons non plus les tâches mais les activités induites par celles-ci et proposons un modèle représentant les interactions entre le diagnostic et l'action et le lien dynamique entre ces interactions et le contexte.

La représentation du contexte est, de plus, explicite dans notre formalisme de représentation. La grande majorité des choix stratégiques représentés sont basés sur des éléments du contexte. Ainsi les connaissances contextuelles sont intégrées au diagnostic et influent directement sur les plans d'actions. Notre formalisme de représentation est basé sur le contexte et sa dynamique en cours de construction du plan de diagnostic/action.

Nous proposons également un algorithme d'acquisition incrémentale de connaissances stratégiques qui permet d'introduire une nouvelle pratique correspondant à une tâche dans le graphe contextuel organisant l'activité correspondante. Cet algorithme est basé sur une forte interaction avec l'opérateur, le guidant dans cette opération et faisant ressortir des connaissances profondes dont l'opérateur n'avait pas conscience. Cette introspection guidée est bénéfique à l'opérateur car elle structure ses propres connaissances sur les raisonnements qu'il applique. L'expertise des opérateurs est ainsi acquise petit à petit par le système et organisée en vue d'une réutilisation future.

Le modèle ainsi obtenu est très proche des schèmes d'action, structures cognitives observée chez les chefs de régulation (et dans de nombreuses autres activités humaines). Ces structures sont des organisateurs de l'activité, ils contiennent des informations sur les conditions et moyens nécessaires à la réalisation de l'activité, des résultats attendus, des raisonnements mis en œuvre et des plans de décision/action. Ces structures cognitives évoluent en fonction des expériences pour intégrer les originalités de chaque situation afin d'améliorer leur efficacité. Cette proximité entre les structures cognitives observées chez les opérateurs et le modèle de raisonnement artificiel que nous proposons explique l'assimilation rapide de ce modèle et de son fonctionnement par les opérateurs.

La mise en œuvre, dans le prototype de SAGIM, de cette modélisation des activités des chefs de régulation est une réussite. Les opérateurs ont rapidement compris le fonctionnement du système. Le système c'est montré tout à fait capable de gérer les modèles d'activité et l'évolution de ceux-ci en fonction de l'évolution des pratiques des opérateurs.

Notre approche est intermédiaire entre plusieurs domaines, à différent niveaux. Premièrement, au niveau de l'architecture de SAGIM, nous proposons une hybridation d'un mode de raisonnement par analogie (le Raisonnement à Partir de Cas) et d'un raisonnement plus structuré (à base de modélisation des activités des opérateurs par des graphes contextuels). Cette hybridation est préconisée par de nombreux travaux sur les implications des modélisations de raisonnement [Davis, Shrobe et Szolovits, 1993]. En effet, chaque modélisation introduit des omissions par rapport à l'objet modélisé. Ainsi la combinaison de différents modèles permet d'avoir différents points de vue et donc une vision plus globale de l'objet modélisé. Dans notre cas, ce qui est modélisé correspond à des stratégies de gestion d'incident. Les conséquences d'omissions peuvent être dramatiques et il est plus prudent d'avoir des points de vue différents. La modélisation par le Raisonnement à Partir de Cas propose ainsi un mécanisme de raisonnement flou faisant émerger des similarités inattendues entre des situations considérées comme très différentes. Les solutions apportées par ce genre de raisonnement peuvent parfois sembler incongrues, mais mettent en avant des spécificités des situations qui peuvent avoir une grande importance. La modélisation des activités des opérateurs par des graphes contextuels permet d'obtenir un raisonnement fortement structuré. Cette structuration opère à deux niveaux : premièrement au niveau des activités entre elles (quelle activité est induite par quelle autre ?), deuxièmement, au sein des activités, ce modèle structure le diagnostic et l'action. Les résultats obtenus sont beaucoup plus classiques et n'introduisent pas de résultats inattendus. Ce mode de raisonnement permet ainsi de structurer les connaissances mises en œuvre dans ces activités et d'expliquer les raisonnements suivis.

Le modèle des graphes contextuels est lui même intermédiaire entre les domaines liés aux raisonnements à base de règles tels que ceux utilisés dans les premiers systèmes experts, les études sur la théorie de la décision et les *Context-Aware Systems*. Les raisonnements à base de règle ont apporté le principe des raisonnements appliqués, permettant ainsi de faire le lien entre les résultats du diagnostic et les actions entreprises. La théorie de la décision apporte quant à elle une structuration des étapes du diagnostic et permet de représenter une première structure de diagnostic/action. Toutefois ces structures de diagnostic/action séparent les deux composantes et ignorent l'impact des actions (et de leurs résultats) sur l'évolution de la situation. L'introduction du contexte montre que le diagnostic et l'action sont indissociables et que le raisonnement est fortement dynamique. Chaque action entreprise modifie la situation et influe donc sur les diagnostics réalisés par la suite. Le raisonnement des opérateurs est basé sur la situation et surtout sur son évolution en cours de traitement. Le contexte et son évolution sont donc au centre des raisonnements des opérateurs et du modèle que nous proposons. Les *Context-Aware Systems* s'intéressent, de plus, à une vision dynamique de l'environnement de l'utilisateur et adaptent leur comportement à celui de l'utilisateur. Notre

approche basée sur le contexte peut tirer parti de ces travaux qui en forment un complément essentiel.

Enfin, notre approche est intermédiaire entre l'Intelligence Artificielle et la psychologie cognitive. Le modèle des activités basé sur des graphes contextuels est en effet fortement lié à des travaux réalisés par la communauté scientifique de psychologie cognitive (sur les schèmes, la décomposition des tâches, le rôle du contexte dans les raisonnements humains...), à d'autres réalisés en Intelligence Artificielle (Systèmes Experts de deuxième génération, paradigme tâche/méthode, planification dynamique...) et enfin à des travaux déjà situés entre ces deux domaines (Frames, Scripts, MOPs...).

Les travaux que nous avons réalisés sur la modélisation de raisonnements en contexte, qui aboutissent à la modélisation des activités par les graphes contextuels, ont été mis en pratique dans le prototype de SAGIM. Ce Système d'Aide Intelligent en Contexte organise les connaissances sur la gestion des incidents et traite ainsi les activités des chefs de régulation du métro. Deux axes de développement de SAGIM sont en cours. Premièrement, une première version opérationnelle de SAGIM est en cours de développement. Elle suit les spécifications fonctionnelles décrites dans le sixième chapitre (sections 2 et 3) et reprend les principes de fonctionnement du prototype décrit dans cette thèse, tout en élargissant le champ d'application. Deuxièmement un prototype est actuellement développé pour adapter SAGIM au Réseau Express Régional (RER). Il introduit notamment une interface logicielle avec les bases de données d'incidents existantes pour le RER. Ce portage est intéressant à deux titres : premièrement il permet d'étendre le champ d'application de SAGIM au RER, qui est un système de transport assez différent du métro, ce qui permet de tester et de valider les idées de notre thèse dans d'autres conditions d'application ; deuxièmement, des bases de données d'incidents inspirées de celles du RER sont en cours d'implantation dans le métro, le portage vers le RER réalise donc une première étape d'intégration de SAGIM dans le nouvel environnement de travail des chefs de régulation du métro. De plus ce prototype devra interagir avec un agent de régulation de SART développé à cette occasion.

Outre les études statistiques traditionnelles, les bases d'incidents gérées par SAGIM permettent d'étudier les incidents typiques et les différentes pratiques mises en œuvre pour leur gestion. La base d'incident typique peut ainsi être le support pour une étude de cas menée par l'encadrement des chefs de régulation. Ces études pourraient être facilitées par un nouvel agent de SART spécialisé dans cette activité.

Une autre extension intéressante de SART serait le développement d'un agent d'aide à l'enrichissement des procédures. Cet agent partirait de la structuration des activités des chefs

de régulation pour proposer de nouveaux plans de diagnostic/action. Cet agent s'adresserait alors aux responsables de l'édition des procédures et guiderait ceux-ci vers une intégration plus importante du contexte et de ses conséquences dans les procédures. Le cycle d'adaptation des procédures aux pratiques serait ainsi accéléré et facilité par SART.

Enfin, l'application des graphes contextuels ne se limite pas à la description d'activités des chefs de régulation. D'autres activités peuvent être modélisées par ces graphes, notamment toutes les activités qui font apparaître une dualité procédure/pratique : gestion d'incident, supervision (check-listes), médecine... Plus généralement, les graphes contextuels sont applicables dans les domaines où une interprétation ou une adaptation de règles générales est nécessaire pour prendre en compte la richesse du contexte réel d'application.

Bibliographie

Références

- Aamodt, A. & Nygard, M.** (1995). Different roles and mutual dependencies of data, information and knowledge - An AI perspective on their integration. *Data and Knowledge Engineering*, **16**, pp. 191-222.
- Aamodt, A.** (1995). Knowledge acquisition and learning by experience - the role of case-specific knowledge. *Machine Learning and Knowledge Acquisition, Integrated Approach*. Boston, Massachusetts: Academic Press, pp. 197-245.
- Aamodt, A. & Plaza, E.** (1996). Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. *Artificial Intelligence Communications*, **7**(1).
- Anderson, J.** (1993). *Rules of the mind*. Hillsdale: Erlbaum.
- Bainbridge, L.** (1980). Le contrôleur de processus. *Bulletin de Psychologie*, **34**, pp. 813-832.
- Bainbridge, L.** (1997). The change in concepts needed to account for human behavior in complex dynamic tasks. *IEEE Trans. on Systems, Man and Cybernetics - Part A: Systems and humans*, **27**(3), pp. 351-359.
- Barlett, F.** (1932). *Remembering: An experimental and social study*. Cambridge: Cambridge University Press.
- Barlett, F.** (1958). *Thinking*. New York: Basic Books.
- Bastien, C.** (1992). Le décalage entre logique et connaissances. *Le Courrier du CNRS, numéro spécial "Science Cognitives"*, **79**, pp. 38.

- Baudin, C. & Pell, B.** (1994). Increasing levels of assistance in refinement of knowledge-based retrieval systems. *Knowledge Acquisition*(6), pp. 179-196.
- Berthouzoz, C.** (1999). A model of context adapted to domain-independent machine translation. *Second International and Interdisciplinary Conference on Modeling and Using Context*. Trento, Springer Verlag, pp. 54-66.
- Béguin, P.** (1994). Travailler avec la CAO en ingénierie industrielle : de l'individuel au collectif dans les activités avec instruments. Thèse pour le doctorat en ergonomie, CNAM, Paris.
- Bilgic, T. & Fox, M. S.** (1996). Constraint-based retrieval of engineering design cases. *Artificial intelligence in Design'96*, Kluwer Academic publisher, pp. 269-288.
- Bisson, G.** (1995). Why and how to define a similarity measure for object-based representation systems. *2nd international conference on building and sharing very large-scale knowledge bases (KBKS)*. Enschede (NL), IOS press, Amsterdam (NL), pp. 236 - 246.
- Boy, G.** (1991). *Intelligent Assistant Systems*. London: Academic Press.
- Brams, G. W.** (1983). *Réseaux de Petri: théorie et Pratique*: Masson.
- Brémond, F. & Thonnat, M.** (1997). Issues in representing context illustrated by scene interpretation applications. *1st International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT'97)*. Rio de Janeiro, pp. 323-332.
- Brézillon, P. & Cases, E.** (1995). Cooperating for assisting intelligently operators. *Design of Cooperative Systems (COOP-95)*, INRIA, pp. 370-384.
- Brézillon, P. et al.** (1997). SART: A system for supporting operators with contextual knowledge. *First International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT'97)*. Rio de Janeiro, Brasil, pp. 209-222.
- Brézillon, P., Pomerol, J.-C. & Saker, I.** (1998). Contextual and contextualized knowledge: an application in subway control. *International journal on Human-Computer Studies*, **48**(3), pp. 357-373.
- Brézillon, P.** (1998). Successes and failures of KBSs in real-world applications: Repport on the International Conference. *Knowledge - Based systems*. Elsevier. Paris, France. **10**, pp. 253-257.

- Brézillon, P. & Pomerol, J.-C.** (1998). Using contextual information in decision making. International Conference on Context-sensitive Decision Support Systems. Bled (Slovenia), Chapman & Hall, pp. 158-173.
- Brézillon, P.** (1999). Le projet SAIC: travaux en cours et perspectives. Paris: LIP6.
- Brézillon, P. & Pomerol, J.-C.** (1999). Contextual knowledge sharing and cooperation in intelligent assistant systems. *Le Travail Humain*, **62**(3), pp. 223-246.
- Brézillon, P.** (1999). Context in problem solving: A survey. *The Knowledge Engineering Review*, **14**(1), pp. 1-34.
- Brézillon, P., Pasquier, L. & Saker, I.** (1999). Context-based reasoning and decision graphs. Application in incident management on a subway line. CSAPC'99. Villeneuve d'Ascq, Presses Universitaires de Valenciennes, pp. 189-194.
- Brézillon, P., Pasquier, L. & Pomerol, J.-C.** (1999). Modelling Decision Making with Context-Based Reasoning and Contextual Graphs. Application in Incident Management on a Subway Line. Human Centered Processes HCP'99. Brest, France, Espace Edition ENST-Bretagne, pp. 331-336.
- Brézillon, P., Pasquier, L. & Saker, I.** (1999). Context-Based Decision Making in Incident Management on a Subway Line. Human Centered Processes HCP'99. Brest, France, Espace Edition ENST-Bretagne, pp. 129-134.
- Brézillon, P. et al.** (2000). SART: an intelligent assistant system for subway control. *Pesquisa Operacional*, **20**(2), pp. 247-268.
- Brézillon, P., Pasquier, L. & Pomerol, J.-C.** (2000). Representing operational knowledge by contextual graphs. SBIA-IBERAMIA 2000, Springer Verlag, pp. 245-258.
- Brézillon, P., Pasquier, L. & Pomerol, J.-C.** (2002). Reasoning with contextual graphs. *European Journal of Operational Research*(136), pp. 290-298.
- Britanik, J. & Marefat, M.** (1999). Hierarchically merging plans in decomposable domains. *IEEE Transactions on systems, man and cybernetics - Part A: Systems and Humans*, **29**(1), pp. 27-40.
- Brown, J. & Duguid, P.** (1991). Organizational learning and communities of practice: Towards a unified view of working, learning and organization. *Organization Science*, **2**(1), pp. 40-57.

- Buchanan, B. G.** (1984). Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project. Reading: Addison-Wesley.
- Burstein, M.** (1989). Analogy vs CBR: The purpose of mapping. Second Case-Based Reasoning Workshop. Pensacola Beach, DARPA, Morgan Kaufmann.
- Cahour, B. & Karsenty, L.** (1993). Context of dialogue: a cognitive point of view. IJCAI-93, Workshop on Using Knowledge In Its Context, Université Paris 6, pp. 20-29.
- Carenini, G. & Moore, J.** (1993). Generating explanations in context. International Workshop on Intelligent User Interfaces. Orlando.
- Cellier, J.-M., Keyser, V. & Valot, C.** (1996). La gestion du temps dans les environnements dynamiques. Paris: Presses Universitaires de France.
- Cellier, J.-M.** (1996). Exigences et gestion temporelle dans les environnements dynamiques. La gestion du temps dans les environnements dynamiques. Paris: PUF.
- Chandrasekaran, B.** (1986). Generic tasks in knowledge-based reasoning: High level building blocks for expert system design. IEEE Expert, **1**(3), pp. 23-30.
- Chandrasekaran, B., Johnson, T. & Smith, J.** (1992). Task-Structure Analysis for Knowledge Modeling. Communication of the ACM, **35**(9), pp. 124-137.
- Chi, M., Glaser, R. & Farr, M.** (1988). The nature of expertise. Hillsdale: Erlbaum.
- Clancey, W.** (1979). Tutoring rules for guiding a case method dialogue. International Journal of Man-Machine Studies, **11**(1), pp. 25-49.
- Clancey, W.** (1985). Heuristic classification. AI Journal, **27**, pp. 289-350.
- CONTEXT'97** (1997). International and Interdisciplinary Conference on Modeling and Using Context. Rio de Janeiro, Brazil.
- CONTEXT'99** (1999). Modeling and Using Context. Heidelberg, Germany: Springer Verlag.
- d'Andrade** (1995). The Development of Cognitive Anthropology: Cambridge University Press.
- Daniellou, F.** (1986). L'opérateur, la vanne, l'écran. L'ergonomie des salles de contrôle: Editions de l'ANACT, Collection Outils et méthodes.
- Davies, N. et al.** (1998). Developing a context-sensitive tourist guide. First Workshop on Human Computer Interaction with Mobile Devices, GIST Technical Report.

- Davis, R., Shrobe, H. & Szolovits, P.** (1993). What is knowledge representation. *AI Magazine*, 14(1), pp. 17-33.
- De Brito, G. & Boy, G.** (1999). Situation awareness and procedure following. *CSAPC'99*. Villeneuve d'Ascq, Presses Universitaires de Valenciennes, pp. 9-14.
- De Terssac, G.** (1992). *Autonomie dans le travail*. Paris.
- Decortis, F. & Cacciabue, P.** (1991). Modélisation cognitive et de l'activité. *Modèles en analyse du travail*. R. Amalberti, M. de Montmollin & J. Theureau. Bruxelles: Mardaga, pp. 85-118.
- Degler, D. & Battle, L.** (2000). Knowledge management in pursuit of performance: The challenge of context. *Performance Improvement Journal*, 39(6), pp. 25-31.
- Desvignes, M., Porquet, C. & Spagnou, P.** (1989). A tool for studying context in image sequences. 3rd IEE International Conference on Image Processing and Applications, pp. 467-472.
- Drummond, M.** (1989). *Situated Control Rules*. First International Conference on Principles of Knowledge Representation and Reasoning. Toronto, Morgan Kaufmann.
- Durkin, J.** (1993). *Expert Systems catalog of applications*. Akron, Ohio: Intelligent Computer Systems Inc.
- Duvenci-Lenga, S.** (1997). *Evolution de l'activité et des compétences en situation d'automatisation : le cas des machines- outils*. Thèse de doctorat d'ergonomie, CNAM, Paris.
- Dyer, M.** (1990). Intentionality and computationalism: minds, machines, Searle and Harnad. *Journal of Experimental and Theoretical Artificial Intelligence*, 2, pp. 303-319.
- Eco, U.** (1997). *Kant et l'Ornithorynque*.
- Edmonds, B.** (1999). The pragmatic roots of context. Second International and Interdisciplinary Conference on Modeling and Using Context. Trento, Springer Verlag, pp. 119-132.
- Ekdahl, B., Astor, E. & Davidsson, P.** (1995). Toward anticipatory agents. *Intelligent Agents*. M. Wooldridge & N. Jennings. Berlin: Springer Verlag, pp. 191-202.
- Farreny, H.** (1985). *Les Systèmes Experts: principes et exemples*. Toulouse.

- Fensel, D., Angele, J. & Landes, D.** (1991). Knowledge representation and acquisition language (KARL). 11th International workshop on expert systems and their applications (Volume: Tools and Techniques). Avignon, France, pp. 821-833.
- Fikes, R. & Nilsson, N.** (1971). STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, **2**, pp. 189-208.
- Fischer, G.** (1990). Communication requirements for cooperative problem solving systems. *Information Systems*, **15**(1), pp. 21-36.
- Frontin, J., Kacem, A. H. & Soubie, J. L.** (1993). Acquérir des connaissances et structurer le système pour coopérer. 2ndes journées Acquisition des Connaissances. St Raphaël.
- Galinier, V.** (1996). Apports de l'ergonomie à la conception d'instruments : concevoir autour des schèmes d'utilisation. un exemple dans le domaine du transport routier, CNAM, Paris.
- Garfinkel, A.** (1981). *Forms and explanation: Rethinking the questions in social theory*: Yale University Press.
- Gonzalez, A. & Ahlers, R.** (1994). A novel paradigm for representing tactical knowledge in intelligent simulated opponents. Seventh International Conference of Industrial Engineering Applications and A.I. and Expert Systems. Austin, Gordon and Breach Science Publishers, pp. 515-523.
- Gonzalez, A. & Ahlers, R.** (1995). Context-based representation of intelligent behavior in simulated opponents. Fifth Conference on Computer Generated Forces and Behavior Representation. Institute for Simulation and Training, Orlando, pp. 53-61.
- Hasher, L. & Zacks, R.** (1984). Automatic processing of fundamental information: The case of frequency of occurrence. *American Psychologist*, **39**, pp. 1372-1388.
- Hatchuel, A. & Weil, B.** (1992). *L'Expert et le système*. Paris, France: Economica.
- Hayes-Roth, B. & Hayes-Roth, F.** (1979). A cognitive model of planning. *Cognitive Science*, **3**, pp. 275-310.
- Healey, J. & Picard, R. W.** (1998). StartleCam: A cybernetic wearable camera. Second international Symposium on Wearable Computers. Pittsburgh, pp. 42-49.
- Hoc, J.-M.** (1996). *Supervision et contrôle de processus. La cognition en situation dynamique*. Grenoble: Presses Universitaires de Grenoble.

- Howard, R.** (1990). From influence to relevance. Influence Diagrams, Belief Nets and Decision Analysis. R. Oliver & J. Smith. New York: John Wiley & Sons, pp. 3-23.
- Humphreys, P. & Berkeley, D.** (1992). Support for the synthesis and analysis of organisational systems in deciding on change. Decision support systems: Experiences and Expectations. Tawfic Jelassi, Michel R. Klein & W.M. Mayon-White. Amsterdam: Elsevier Science Publishers, pp. 29-50.
- Jansen, B.** (1993). Context: A real problem for large and sharable knowledge bases. 1st Conference on Building and Sharing of Very Large-Scale Knowledge Bases. Tokyo, pp. 177-183.
- Jensen, A.** (1995). A probabilistic model based decision support system for mildew management in winter wheat, Aalborg University, Aalborg.
- Jensen, A.** (1996). An introduction to Bayesian networks. London: UCL Press.
- Juristica, I.** (1993). Representation and Management Issues for Case-Based Reasoning Systems. Toronto: Dep. of Computer Science, University of Toronto.
- Kambhampati, S. & Hendlar, J.** (1992). A validation-structure-based theory of plan modification and reuse. Artificial Intelligence, **55**, pp. 193-258.
- Karsenty, L.** (1994). L'explication d'une solution dans les dialogues de conception, Université Paris 8, Paris.
- Karsenty, L. & Brézillon, P.** (1995). Cooperative problem solving and explanation. International Journal of Expert Systems with Applications, **8**(4), pp. 445-462.
- Kohavi, R.** (1995). Wrappers for performance enhancement and Oblivious Decision Graphs, Computer Science, Stanford.
- Kolodner, J.** (1993). Case-Based Reasoning. San Mateo, Californie: Morgan Kaufmann.
- Kott, A., Saks, V. & Mercer, A.** (1999). A new technique enables dynamic replanning and rescheduling of aeromedical evacuation. AI Magazine, **20**(1), pp. 43-53.
- Lamming, M. & Flynn, M.** (1994). Forget-Me-Not: Intimate computing in support of human memory. International Symposium on Next Generation Human Interfaces. Tokyo, pp. 125-128.
- Langacker** (1987). Foundations of Cognitive Grammar, Theoretical Prerequisites. Stanford: Stanford University Press.

- Leake, D. B.** (1992). Evaluating explanations. Hillsdale: Lawrence Erlbaum Associates.
- Leake, D. B.** (1996). CBR in Context: The Present and Future. Case - Based Reasoning: Experiences, Lessons, and Future Directions. D. Leake. Menlo Park: AAAI Press / MIT Press, pp. Chapter 1.
- Leplat, J. & Hoc, J.-M.** (1983). Tâche et activité dans l'analyse psychologique des situations. Cahiers de Psychologie Cognitive, **3**, pp. 49-63.
- Leplat, J.** (1985). Erreur humaine, fiabilité humaine dans le travail. Paris: Armand Colin.
- Lester, J. C. & Porter, B. W.** (1991). Generating context-sensitive explanations in interactive Knowledge-Based Systems. AAAI'91, workshop on Comparative Analysis of Explanation Planning Architectures, pp. 27-41.
- Lévine, P. & Pomerol, J.-C.** (1989). Systèmes interactifs d'aide à la décision et systèmes experts. Paris: Hermès.
- Mandler, J.** (1984). Stories, scripts and scenes: Aspects of schema theory. Hillsdale: Erlbaum.
- Mark, B.** (1988). Explanation and interactive knowledge acquisition. AAAI'88, Workshop on Explanation.
- Maskery, H. & Meads, J.** (1992). Context: what does it mean to application design. SIGCHI Bulletin, **24**(2), pp. 12-21.
- McCarthy, J.** (1979). First-order theories of individual concepts and propositions. Machine Intelligence, **9**.
- McCarthy, J.** (1993). Notes on formalizing context. Thirteenth International Joint Conference on Artificial Intelligence (IJCAI'93). Chambéry, France, R. Bajcsy, pp. 555-560.
- Mignot, H.** (1997). Sensibilité au contexte lors de l'évaluation de similarité en raisonnement à partir de cas, Laboratoire de recherche en informatique, Université Paris-Sud, Paris.
- Mille, A. & Napoli, A.** (1997). Aspects du raisonnement à partir de cas. 6èmes Journées nationales PRC - GDR IA'97, pp. 261-287.
- Minsky, M.** (1975). A framework for representing knowledge. The psychology of computer vision. P. Winston. New York: McGraw-Hill.
- Neapolitan, R.** (1990). Probabilistic reasoning in expert systems. New York: John Wiley & Sons.

- Oliver, R. & Smith, J.** (1990). Influence diagrams, belief nets and decision analysis. New York: John Wiley & Sons.
- OMG** (1999). OMG Unified Modeling Language Specification. Framingham.
- Pasquier, L.** (1998). Raisonement basé sur le contexte. Application à la gestion d'incidents sur une ligne de métro. Paris: LIP6, Université Pierre et Marie Curie.
- Pasquier, L., Brézillon, P. & Pomerol, J.-C.** (1999). Context and Decision Graphs for Incident Management on a Subway Line. Modeling and Using Context. Paolo Bouquet, Luciano Serafini, Patrick Brézillon, Massimo Benerecetti & Francesca Castellani. Berlin, Germany: Springer Verlag. **1688**, pp. 499-502.
- Pasquier, L.** (2000). Gestion d'incidents sur une ligne de métro. Modélisation de raisonnements tenus en contexte et application aux agents d'aide à la gestion d'incidents de SART. Paris, France: RATP.
- Pasquier, L., Brézillon, P. & Pomerol, J.-C.** (2001). From representation of operational knowledge to practical decision making in operations. Decision Support through Knowledge Management. Sven Karlson, Patrick Brézillon, Patrick Humphreyset al. Edsbruck, Sweden: Akademitryck AB., pp. pp. 301-320.
- Pasquier, L. & Zanarelli, C.** (2001). Utilisation de recommandations d'ergonomie dans le développement d'un outil d'aide à la régulation de trafic du métro. Ingénierie des Connaissances 2001. Grenoble, France, Presses Universitaires de Grenoble, pp. 197-213.
- Pasquier, L., Brézillon, P. & Pomerol, J.-C.** (2002). Learning and explanation in a context-based representation: application to incident management on subway lines. Innovations in Knowledge Engineering. Colette Faucher, Lakhmi Jain & Nikhil Ichalkaranje: Physica-Verlag, pp. (to appear).
- Pastré, P.** (1994). Le rôle des schèmes et des concepts dans la formation des compétences. Performances, **7**.
- Pearl, J.** (1988). Probabilistic reasoning in intelligent systems. San Mateo: Morgan Kaufmann.
- Pereira, F. C. & Pollack, M. E.** (1991). Incremental interpretation. Artificial Intelligence, **50**(1), pp. 37-82.
- Piaget, J.** (1936). La naissance de l'intelligence chez l'enfant. Paris, Lausanne.

- Pomerol, J.-C. & Brézillon, P.** (1996). Are interactivity problems in KBSs similar to DSSs ones ? Success and failures of Knowledge-Based Systems in real-world applications. Dentcho Batanov & Patrick Brézillon. Bangkok, pp. 1-10.
- Pomerol, J.-C.** (1997). Artificial intelligence and human decision making. *European Journal of Operational Research*, **99**, pp. 3-25.
- Pomerol, J.-C. & Brézillon, P.** (1999). Dynamics between Contextual Knowledge and Proceduralized Context. *CONTEXT'99*. Trento, Italy, Springer, pp. 284-295.
- Pomerol, J.-C., Brézillon, P. & Pasquier, L.** (2001). Operational Knowledge Representation for Practical Decision Making. *Journal of Management Information Systems*, **18**(4), pp. 101-116.
- Prince, V.** (1996). *Vers une informatique cognitive dans les organisations . Le rôle central du langage*. Paris, France: Masson.
- Quinn, N. & Holland, D.** (1987). *Cultural models of language and thought*. New York: Cambridge University Press.
- Rabardel, P.** (1995). *Les Hommes et les technologies. Approche cognitive des instruments contemporains*. Paris: A. Colin.
- Raïffa, H.** (1968). *Decision Analysis*. New York: Mac Graw Hill.
- Rastier, F.** (1996). *Le problème épistémologique du contexte et le statut de l'interprétation dans les sciences du langage*. CES. Rome, Italy.
- Reddy, R.** (1996). The challenge of Artificial Intelligence. *Computer Journal*, **29**(10), pp. 86-98.
- Refanidis, I. & Vlahavas, I.** (2001). The GRT planning system: backward heuristic construction in forward state-space planning. *Journal of Artificial Intelligence Research*, **15**, pp. 115-161.
- Richard, J.-F.** (1983). *Logique du fonctionnement et logique de l'utilisation*. Le Chesnay (France): INRIA.
- Robert, J.** (1983). *Notre Métro*. Argenton-sur-Creuse: Imprimerie de l'Indre.
- Rocha, J., Ramos, C. & Vale, Z.** (1998). Complexity of precedence graphs for assembly and task planning. 11th international Conference on Industrial and Engineering Applications

of Artificial Intelligence and Expert Systems (IEA/AIE-98). Castellon (Spain), Springer-Verlag, pp. 149-158.

Rumelhart, D. (1980). Schemata: The building blocks of cognition. Theoretical issues in reading and comprehension. R. Spiro, B. Bruce & W. Brewer. Hillsdale: Erlbaum.

Saint-Sernin, B. (1968). Stratégie. Encyclopedia Universalis: Universalis. **15**, pp. 407-409.

Saker, I. (1999). Modélisation des connaissances en contexte. Application à la gestion d'incidents sur une ligne de métro, Laboratoire d'Informatique de Paris 6, Université Pierre et Marie Curie, Paris, France.

Samurçay, R. & Hoc, J.-M. (1996). Causal versus topographical support for diagnosis in a dynamic situation. *Le Travail Humain*, **59**(1), pp. 45-68.

Schank, R. & Abelson, R. (1977). Scripts, Plans, Goals and Understanding. Hillsdale: Erlbaum.

Schank, R. (1982). Dynamic memory: A theory of reminding and learning in computers and people. Cambridge: Cambridge University Press.

Schmidt, C. T. (1995). Information processing, context creation, setting minds in public arenas: investigative techniques for client/automaton dialogue design. IJCAI'95, workshop on Modeling Context in Knowledge Representation and Reasoning, Université Paris 6, pp. 121-131.

Senach, B. (1984). Assistance automatisée à la résolution d'incidents dans les systèmes dynamiques. Réduction de l'espace problème et routines cognitives. Thèse de 3e cycle en ergonomie, Université Paris V.

Shortliffe, E. H. (1976). Computer-Based Medical Consultations: MYCIN. New York: American Elsevier.

Sowa, J. (1984). Conceptual structures: information processing in mind and machine. Reading: Addison Wesley Publishing Company.

Sowa, J. F. (1992). Conceptual Graphs Summary. Conceptual Structures - current research and practice. T.E. Nagle, J.A. Nagel, L.L. Gerholz & P.W. Eklund. New York: Ellis Horwood, pp. 3-51.

Spérandio, J.-C. (1993). L'Ergonomie du contrôle de processus. L'ergonomie dans la conception des projets informatiques. Toulouse: Octarès.

- Stacey, M., Sharp, H. & Petre, M.** (1996). A representation scheme to support conceptual design of mechatronic systems. *Artificial Intelligence in Design*. J. Gero & F. Sudweeks. Dordrecht: Kluwer Academic Publishers.
- Steels, L.** (1992). Reusability and configuration of applications by non-programmers. Brussels: VUB AI Lab.
- Sun, R.** (1995). Robust reasoning: integrating rule-based and similarity-based reasoning. *Artificial Intelligence*, **75**(2), pp. 241 - 295.
- Suthers, D.** (1993). Influences of the epistemic context on explanation. *IJCAI'93*, workshop on Using Knowledge in its Context, Université Paris 6.
- Tansley, S. & Hayball, C.** (1993). *Knowledge-Based Systems analysis and design: a KADS developer's handbook*. Hertfordshire: Prentice Hall.
- Tiberghien** (1986). Context and cognition: Introduction. *Cahiers de Psychologie Cognitive*, **6**(2), pp. 105-119.
- Tijus, C.** (2001). *Introduction à la psychologie cognitive*. Paris: Nathan Université.
- Tijus, C.** (2001). Contextual categorization and cognitive phenomena. *CONTEXT 2001*, Springer Verlag, pp. 316-329.
- Tricoire, J.** (1999). *Un siècle de métro en 14 lignes*. Paris: La Vie du Rail.
- Turner, R.** (1989). When reactive planning is not enough: Using contextual schemas to react appropriately to environmental change. *Eleventh Annual Conference of the Cognitive Science Society*. Detroit, pp. 940-947.
- Turner, R.** (1995). Intelligent control of autonomous underwater vehicles: The ORCA project. *IEEE International Conference on Systems, Man and Cybernetics*. Vancouver.
- Turner, R. M.** (1998). Context-Mediated Behavior for Intelligent Agents. *International Journal of Human-Computer Studies*, **48**(3), pp. 307-330.
- Turner, E. & Turner, R.** (1998). A constraint-based approach to assigning system components to tasks. *11th international Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE-98)*. Castellon (Spain), Springer-Verlag, pp. 312-320.

- Turner, R.** (1999). Context-mediated behavior: An approach to explicitly representing contexts and contextual knowledge for AI applications. Working Notes of the Workshop on Modeling and Using Contexte in AI Applications. Orlando.
- Tversky, A.** (1977). Features of similarity. *Psychological Review*, **84**(4), pp. 327-352.
- Veloso, M.** (1994). Planning and Learning by Analogical Reasoning.
- Veloso, M.** (1997). Merge strategies for multiple case plan replay. ICCBR'97.
- Vergnaud, G.** (1985). Concepts et schèmes dans la théorie opératoire de la représentation. *Les Représentation, Psychologie Française*, **30**(3), pp. 245-252.
- Walter, E., Eriksson, H. & Musen, M. A.** (1992). Plug-and-play: Construction of task-specific expert-system shells using sharable context ontologies. Stanford: Stanford University.
- Want, R. et al.** (1992). The Active Badge location system. *ACM Transactions on Information Systems*, **10**(1), pp. 91-102.
- Want, R.** (1995). An overview of the PARCTAB ubiquitous computing environment. *IEEE Personal Communications*, **2**(6), pp. 28-43.
- Watson, I.** (1995). Case-Based Reasoning Tools : An Overview. 2nd United Kingdom CBR Workshop AI-CBR. Salford University (UK), pp. 71-88.
- Watson, I. & Perera, S.** (1998). A hierarchical case based representation using context guided retrieval. *Knowledge - Based Systems Journal*.
- Wheeler, M. & Clark, A.** (1999). Genic Representation: Reconciling Content and Causal Complexity. *British journal for the Philosophy of Science*, **50**(1), pp. 103-135.
- Wielinga, B., Schreiber, A. & Breuker, J.** (1992). KADS: a modeling approach to knowledge engineering. *Knowledge Acquisition*, **4**(1), pp. 162.
- Wielinga, B. & Schreiber, G.** (1997). Configuration-design problem solving. *IEEE-Expert*, **12**(2), pp. 49 - 56.
- Zanarelli, C.** (1998). Identifier des classes de situations caractéristiques : quels critères de différenciation des situations sont utilisés par les chefs de régulation et les chefs de départ. Paris, France: RATP.

- Zanarelli, C.** (1999). Réflexion sur l'évolution des modes de régulation d'un réseau ferroviaire urbain : application de l'approche instrumentale par identification de classes de situations caractéristiques. XXXIVe congrès de la SELF. Caen, France, pp. 485-494.
- Zanarelli, C.** (1999). Analyse de l'activité des chefs de régulation : de la structure de l'activité aux artefacts en usage. Paris: RATP.
- Zanarelli, C., Saker, I. & Pasquier, L.** (1999). Un projet de coopération ergonomes/concepteurs autour de la conception d'un outil d'aide à la régulation du trafic du métro. Ingénierie des Connaissances (IC'99). Palaiseau, France, Association Française pour l'Intelligence Artificielle, Chambéry, France, pp. 161-170.
- Zanarelli, C. & Pasquier, L.** (2000). Conception centrée sur l'activité réelle : développement d'un outil d'aide à la régulation de trafic du métro. Journée satellite de la SELF: Ergonomie dans les Transports Ferroviaires. Toulouse, France, pp. pp 204-217.
- Zanarelli, C.** (2002). Caractérisation des stratégies instrumentales de gestion d'environnements dynamiques. Université Paris 8, Paris.

Publications

- Pasquier, L.** (1998). Raisonement basé sur le contexte. Application à la gestion d'incidents sur une ligne de métro. Paris: LIP6, Université Pierre et Marie Curie.
- Zanarelli, C., Saker, I. & Pasquier, L.** (1999). Un projet de coopération ergonomes/concepteurs autour de la conception d'un outil d'aide à la régulation du trafic du métro. Ingénierie des Connaissances (IC'99). Palaiseau, France, Association Française pour l'Intelligence Artificielle, Chambéry, France, pp. 161-170.
- Brézillon, P., Pasquier, L. & Pomerol, J.-C.** (1999). Modelling Decision Making with Context-Based Reasoning and Contextual Graphs. Application in Incident Management on a Subway Line. Human Centered Processes HCP'99. Brest, France, Espace Edition ENST-Bretagne, pp. 331-336.
- Brézillon, P., Pasquier, L. & Saker, I.** (1999). Context-based reasoning and decision graphs. Application in incident management on a subway line. CSAPC'99. Villeneuve d'Ascq, Presses Universitaires de Valenciennes, pp. 189-194.
- Brézillon, P., Pasquier, L. & Saker, I.** (1999). Context-Based Decision Making in Incident Management on a Subway Line. Human Centered Processes HCP'99. Brest, France, Espace Edition ENST-Bretagne, pp. 129-134.
- Pasquier, L., Brézillon, P. & Pomerol, J.-C.** (1999). Context and Decision Graphs for Incident Management on a Subway Line. Modeling and Using Context. Paolo Bouquet, Luciano Serafini, Patrick Brézillon, Massimo Benerecetti & Francesca Castellani. Berlin, Germany: Springer Verlag. **1688**, pp. 499-502.
- Pasquier, L.** (2000). Gestion d'incidents sur une ligne de métro. Modélisation de raisonnements tenus en contexte et application aux agents d'aide à la gestion d'incidents de SART. Paris, France: RATP.
- Brézillon, P., Pasquier, L. & Pomerol, J.-C.** (2000). Representing operational knowledge by contextual graphs. SBIA-IBERAMIA 2000, Springer Verlag, pp. 245-258.

- Zanarelli, C. & Pasquier, L.** (2000). Conception centrée sur l'activité réelle : développement d'un outil d'aide à la régulation de trafic du métro. Journée satellite de la SELF: Ergonomie dans les Transports Ferroviaires. Toulouse, France, pp. pp 204-217.
- Pasquier, L., Brézillon, P. & Pomerol, J.-C.** (2001). From representation of operational knowledge to practical decision making in operations. Decision Support through Knowledge Management. Sven Karlson, Patrick Brézillon, Patrick Humphreyset al. Edsbruck, Sweden: Akademitryck AB., pp. pp. 301-320.
- Pasquier, L. & Zanarelli, C.** (2001). Utilisation de recommandations d'ergonomie dans le développement d'un outil d'aide à la régulation de trafic du métro. Ingénierie des Connaissances 2001. Grenoble, France, Presses Universitaires de Grenoble, pp. 197-213.
- Pomerol, J.-C., Brézillon, P. & Pasquier, L.** (2001). Operational Knowledge Representation for Practical Decision Making. Journal of Managment Information Systems, **18**(4), pp. 101-116.
- Brézillon, P., Pasquier, L. & Pomerol, J.-C.** (2002). Reasoning with contextual graphs. European Journal of Operational Research(136), pp. 290-298.
- Pasquier, L., Brézillon, P. & Pomerol, J.-C.** (2002). Learning and explanation in a context-based representation: application to incident management on subway lines. Innovations in Knowlegde Engineering. Colette Faucher, Lakhmi Jain & Nikhil Ichalkaranje: Physica-Verlag, pp. (to appear).

Bibliographie complémentaire

- Agosta, J.** (1996). Constraining Influence Diagram Structure By Generative Planning: An Application To The Optimisation Of Oil Spill Response. 12 th Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann, pp. 11-19.
- Bainbridge, L.** (1987). Ironies of automation. New technology and human error. J. Rasmussen, K. Duncan & J. Leplat. Chichester: Wiley, pp. 271-284.
- Boy, G.** (1998). L'Interaction homme machine: une approche de l'ingénierie cognitive pour la conception centrée sur l'homme. Sécurité et Cognition. Jean-Gabriel Ganascia, pp. 79-101.
- Buvac, S.** (1995). Resolving Lexical Ambiguity using a Formal Theory of Context.
- Falzon, P.** (1998). "Qu'est-ce que la recherche en ergonomie ?". 2èmes journées de Recherche et Ergonomie. Toulouse, pp. 191-196.
- Farenc, C. & Barthet, M.-F.** (1998). L'évolution de l'intégration de l'ergonomie dans le développement des applications informatiques. Deuxièmes journées "Recherche et Ergonomie". Université de Toulouse - Le Mirail, pp. 53-56.
- Fontaine, D.** (1996). Une approche par graphes pour la reconnaissance de scénarios temporels. Revue d'Intelligence Artificielle, **10**(4), pp. 439-468.
- Gil, Y. & Paris, C.** (1994). Towards method-independant knowledge acquisition. Knowledge Acquisition(6), pp. 163-178.
- Gil, Y. & Tallis, M.** (1997). A Script-Based Approach for Modifying Knowledge Bases. Fourth national conference on Artificial Intelligence (AAAI - 97). Providence, Rhode Island.
- Grolimund, S. & Ganascia, J.-G.** (1995). Integrating case based reasoning and tabu search for solving optimisation problems. International Conference on Case-based Reasoning. Sesimba, Portugal.
- Grolimund, S.** (1996). Accélération des mémoires de cas du type plus proches voisins. Actes des journées Françaises d'Apprentissage. Sète, France.

- Grosjean, V. et al.** (1998). Reproduire en situation simulée les contraintes cognitives futures: une démarche ergonomique de prévention. *Sécurité et Cognition*. Jean-Gabriel Ganascia, pp. 39-47.
- Hansen, B. & Riordan, D.** (1998). Fuzzy case-based prediction of ceiling and visibility. 1st conference on Artificial Intelligence of the American Meteorological Society, pp. 118-123.
- Hartley, R. & Coombs, M.** (1991). Reasoning with graph operations. *Principles of semantic networks*. John Sowa. San Mateo (California): Morgan Kaufmann, pp. 487-505.
- Hoc, J.-M.** (1998). Conditions et enjeux de la coopération homme-machine dans le cadre de la fiabilité des systèmes. *Sécurité et Cognition*. Jean-Gabriel Ganascia, pp. 147-164.
- Hoc, J.-M.** (1998). L'ergonomie cognitive: un compromis nécessaire entre des approches centrées sur la machine et des approches centrées sur l'homme. 2èmes Journées Recherche et Ergonomie. Toulouse (France), pp. 1-10.
- Kim, J. K., Lee, J. K. & Kim, S. H.** (1998). An interactive approach to build an influence diagram based on neural networks. .
- Le Blaye, P.** (1998). Collaboration ONERA-NASA-ASRA sur les méthodes d'analyse de rapports d'incidents aéronautiques. *Sécurité et Cognition*. Jean-Gabriel Ganascia, pp. 223-235.
- Leake, D. B. & Wilson, D. C.** (1998). Categorizing Case-Base Maintenance: Dimensions and Directions. *Advances in Case-Based Reasoning: proceedings of EWCBR-98*. Berlin: Springer Verlag.
- Lenat, D.** (1998). *The dimensions of context-space*. Austin: Cycorp.
- M.R.T., G.** (1996). *Méthode de Raisonnement Tactique*: Ecole Nationale Supérieure des Officiers Sapeurs-Pompiers.
- McCarthy, J.** (1984). Some Expert System need common sense. *Annals of the New York Academy of Sciences*. Heinz Pagels.
- Millot, P.** (1998). La Supervision et la coopération homme-machine dans les grands systèmes industriels ou le transport. *Sécurité et Cognition*. Jean-Gabriel Ganascia, pp. 125-145.

- Motschnig-Pitrik, R.** (1995). An Integrating View on the Viewing Abstraction: Contexts and Perspectives in Software Development, AI, and Databases. *Journal of Systems Intergration*(5), pp. 23 - 60.
- Mugnier, M.-L. & Chein, M.** (1996). Représenter des connaissances et raisonner avec des graphes. *Revue d'intelligence artificielle*, **10**(1), pp. 7 - 56.
- Napoli, D.** (1991). Historic Preservation in Point Arena, California: A Model for Context-Based Planning. *Small Town*(21), pp. 4-9.
- Reuzeau, F. & LeDraoullec, C.** (1998). Evaluation de l'impact de l'installation d'un système d'aide à la prise de décision sur le comportement d'un équipage. *Sécurité et Cognition*. Jean-Gabriel Ganascia, pp. 63-78.
- Rogalski, J. & Samurçay, R.** (1993). Représentation de référence : outils pour le contrôle d'environnements dynamiques. *Représentations pour l'action*. Annie Weil-Fassina, Pierre Rabardel & Danièle Dubois: Octares, pp. 183-207.
- Rosenstein, M. et al.** (1997). Action representation, prediction and concepts. *AAAI Workshop on Robots, Softbots, Immobots: Theories of Action, Planning and Control*.
- Silverman, B. G.** (1990). Critiquing Human Judgement Using Knowledge-Based Systems. *AI Magazine*, pp. 60-79.
- Theodorakis, M. et al.** (1998). A Theory of contexts in information bases. Hellas: ICS Forth.
- Trichet, F., Leclère, M. & Choquet, C.** (2000). Construire un système à base de connaissances de type tâche/méthode à l'aide des graphes conceptuels. *Ingénierie des Connaissances (IC'2001)*. Toulouse (France).
- Watson, I.** (1995). An Introduction to Case-Based Reasoning. *Progress in Case-Based Reasoning*, pp. 3 - 16.
- Watson, I. & Perera, S.** (1997). The evaluation of a hierarchical case representation using context guided retrieval. *Case-based erasoning research & development*. David Leake & E Plaza. berlin: Springer-Verlag. **1266**.
- Watson, I.** (1998). Is CBR a technology or a methodology. *11th international Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE-98)*. Castellon (Spain), Springer-Verlag, pp. 525-534.

Wielinga, B. & Schreiber, G. (1998). Knowledge technology: moving into the next millennium. 11th international Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE-98). Castellon (Spain), Springer-Verlag, pp. 1-20.

Sigles et abréviations

CDV	Circuit De Voie Circuit électrique associé à une portion de voie fermé par la présence d'une rame. Ce dispositif permet donc de savoir quelles sont les portions de voies occupées. <u>Par extension</u> : CDV désigne également la portion de voie associée au dispositif électrique
CREG	Chef de REGulation Agent de la RATP chargé de la régulation du trafic de rames de métro sur une ligne et de la gestion des incidents d'exploitation.
CS	Contacteur de Sectionnement Dispositif électrique situé entre deux sections voisines permettant, quand il est fermé, l'alimentation de l'une par l'autre.
CVDA	Classe Virtuelle D'Action Classe (au sens du C++) virtuelle regroupant les attributs et les méthodes communs aux classes Action et Scheme .
CxBR	Context-Based Reasoning Raisonnement basé sur le contexte défini par Gonzalez et Ahlers [1994]
DA	Disjonction d'Alarme Mise hors tension d'une section par actionnement d'un rupteur d'alarme (voir RA).

DG	Déclenchement Général Mise hors tension de toute la ligne par action volontaire du CREG.
DI	Disjonction d'Intensité Mise hors tension automatique plus ou moins fugitive par des disjoncteurs réagissant à toute perte d'intensité entre la masse électrique et le +750V.
DSO	Départ Sur Ordre Procédure de rétention d'une rame en station pour augmenter (respectivement diminuer) son intervalle avec la rame précédente (respectivement suivante). <u>Par extension</u> : dispositif placé en tête de chaque quai indiquant au conducteur que la procédure est mise en œuvre. Ce dispositif s'allume soit sur commande du CREG, soit automatiquement, et donne l'ordre au conducteur de ne pas démarrer L'extinction du dispositif annonce l'ordre de départ au conducteur.
GPS	Global Position System Système de positionnement global permettant à un récepteur de connaître la longitude et la latitude de sa position actuelle par triangulation de signaux émis par des satellites.
HLP	Haut Le Pied Procédure de rapatriement d'une rame sans voyageur et de manière autonome.
IA	Intelligence Artificielle Branche de recherche informatique dont le but est de concevoir des systèmes ayant un comportement intelligent.
IL	Incident Ligne Mise hors tension automatique d'une section suite à une répétition de disjonctions d'intensité dans un laps de temps court. La section ne se remet pas automatiquement sous tension.

IPEX	Inspecteur Principal d'EXploitation Agent de la RATP chargé de la coordination des moyens extérieurs nécessaires à la gestion d'un incident (police, pompiers, EDF...).
KSA	Actionnement d'un Signal d'Alarme Autre nom de l'actionnement d'un frein de secours par un voyageur.
MOP	Memory Organization Packet Formalisme proposé par Schank [1982] pour la modélisation de mémoires dynamiques.
PAI	Plate-forme Autonome Intelligente
PCC	Poste de Contrôle Centralisé Lieu de travail des CREG. Pour chaque ligne le PCC est composé d'un TCO et d'une console, offrant diverses sources d'information, commandes et systèmes de communication.
PDA	Personal Digital Assistant Dispositif électronique personnel offrant différentes fonctionnalités telles qu'agenda, répertoire... (également appelé agenda électronique).
PML	Poste de Manœuvre Local Lieu offrant les commandes et moyens d'information nécessaires à la gestion locale des trains dans un terminus.
PR	Poste de Redressement Poste électrique alimentant la ligne de métro en courant continu.
PREDIT	Programme national de REcherche et D'Innovation dans les Transports terrestres
RA	Rupteur d'Alarme Dispositif de coupure volontaire du courant placé régulièrement le long des voies et sur les quais.

RàPC	Raisonnement à Partir de Cas Raisonnement artificiel apparenté au raisonnement par analogie.
RATP	Régie Autonome des Transports Parisiens Entreprise de transport publique de Paris et de sa proche banlieue.
RER	Réseau Express Régional Système de transport ferré de grand gabarit desservant Paris et sa banlieue.
SAGIM	Système d'Aide à la Gestion d'Incidents dans le Métro Application issue des travaux présentés dans cette thèse.
SAI	Système d'Assistance Intelligent Modèle de systèmes proposé par Boy [1991] basé sur une forte interactivité avec l'utilisateur et entretenant des relations de type opérateur/assistant.
SAIC	Système d'Aide Intelligent en Contexte Modèle de systèmes proposé par Brézillon [1999a] reprenant les principes des SAI et plaçant le contexte au cœur des fonctionnalités.
SART	Système d'Aide à la Régulation du Trafic Projet encadrant le système SAGIM.
SBC	Système à Base de Connaissances Système expert de seconde génération proposant notamment une gestion simplifiée des connaissances grâce à une hiérarchisation de celles-ci.
SE	Système Expert Programme informatique présentant une certaine capacité d'expertise, généralement fondée sur la logique propositionnelle.
SIAD	Système Interactif d'Aide à la Décision

SIT	Sectionneur d’Isolement Télécommandé Dispositif électrique habituellement fermé permettant l’alimentation des sous-sections non-munies de PR.
SP	Service Provisoire Procédure permettant de retourner des trains en ligne pour rééquilibrer la charge sur les deux voies.
STA	STAtionnement Stationnement d’une rame
TCO	Tableau de Contrôle Optique Tableau synoptique s’adressant aux CREG, portant de nombreuses informations et plusieurs commandes pour la gestion de l’énergie électrique notamment.
THF	Téléphonie Haute Fréquence Système de télécommunication entre les conducteurs et le CREG ayant pour support les rails.

Annexes

Annexe 1 :

Typologie des incidents et des causes

Classification des incidents par famille/nature/motifs

Installations Fixes

Voie

- appareil de voie faussé
- voie affaissée
- rail cassé

Alimentation traction

- section HT
- diminution d'énergie

Signalisation

- signal fermé anormalement
- avarie = marche à vue
- itinéraire non autorisé
- divers

Voyageurs

Portes du train

- voyageur coincé par une porte
- entrave à la fermeture des portes du train

Voies

- intrusion dans les voies
- descente d'un voyageur sur les voies

Actionnement du signal d'alarme du train

Actionnement du rupteur d'urgence

Accidents

Accidents graves (Ex : suicide ou tentative)

Matériel Roulant

Portes du train

- non fermeture
- non fermeture d'une porte
- fermeture lente d'une porte
- non ouverture d'une porte

Traction

- réduction de l'effort de traction
- non démarrage total
- marche lente

Freinage

- réduction de l'effort de freinage
 - non desserrage
- Dégagement de fumée sous un train
Signal d'alarme du train

Conducteurs

Retard

- à la prise de service
- au départ du terminus
- en ligne

Conditions de travail

Grève

Divers

Dégagement de fumée

Animal sur la voie

Déraillement

Classification des incidents par cause

Sinistre

Inondation (naturelle, d'une canalisation)

Incendie

Tremblement de terre

Météo (neige, brouillard, glace, pluie forte)

Explosion

Fuite de gaz

Fuite d'un liquide dangereux (acide, huile, essence)

Dégagement de fumée

Événement exceptionnel

(vol, crime, fusillade, trafic trop important, retard de prise de service)

Équipement

Matériel roulant

Énergie

Disjoncteur

Transformateur

Propulsion

- traction

- freinage

Électrique

Éclairage

Signalisation (nom du train ou du terminus)

Signal d'alarme

Avertisseur sonore (klaxon)

Électronique

Batterie
Pilotage automatique
THF
Sonorisation (des wagons)

Electromécanique
Climatisation (chauffage, réfrigération, ventilation)
Porte (non fermeture, fermeture lente, non ouverture,
- bloquée par une personne)
Frein
Console
- manette de traction/freinage
- clé
- manette de freinage
Bogie
- roue
- moteur
Compresseur (système pneumatique)

Mécanique
Porte
- d'intercommunication
- des panneaux de contrôle
Roue
Attelage des wagons
Siège (passager)
Vitre
Chassis de wagon
Essuie-glaces
Dégradation/Vandalisme
Déraillement

Traction

Installation fixe
Energie
Rail de traction
Redresseur 750Vcc
Disjoncteur
Sectionneur
Rupteur d'alarme
Transformateur

Electrique
Signalisation de quai
SS auxiliaire
Poste redresseur
Rupteur d'urgence

Electronique
Signalisation
Pilotage automatique
Communication
Téléphonie directe

- Téléphonie à distance
- Sonorisation
- Télévision (pour le public, pour le conducteur)
- Balise
- Console
- Télétransmission
- TCO
- Electromécanique
 - Climatisation (réfrigération, chauffage, ventilation)
 - Groupes électrogènes
 - Portes automatiques
 - Portes palières
- Voie
 - Rail (rail en fer, rail pour pneu, guide)
 - Aiguillage
 - Traverse
 - Ballast
 - Changement d'état (voie faussée, affaissée, cassée)
- Bâtiment
 - PCC
 - TCO
 - Console
 - Moyen de communication
 - Image
 - Ordinateur
 - Affaissement d'un quai, d'un couloir
 - Affaissement d'une voie
 - Canalisation (eau, gaz)
 - Chauffage
 - Voie (tunnel, station)
 - Station
 - partie publique
 - partie technique
 - sous-station
 - local technique
- Etres vivants
 - Humain
 - Voyageur
 - Agent de la RATP
 - Agent de l'extérieur (police, pompier, ambulancier)
 - Autre (Enfant, SDF, Délinquant, Autres)
 - Groupe
 - groupe organisé
 - groupe non organisé
 - Animal
- Objet
 - Abandonné

Trouvé
Réclamé
Perdu
Jeté
Encombrant

Annexe 2 :

Exemple d'un fichier d'actions et d'activités

Le *listing* qui suit correspond au fichier contenant la définition des actions et des activités impliquées dans l'exemple que nous avons traité dans le troisième chapitre. La description du format de ce fichier est donnée dans le sixième chapitre (section 4.2.2.3, page 192).

20 Actions

- A0 : {A1|Couper le courant sur toute la ligne|Actionner le commutateur DG au pupitre ou au TCO}
- A1 : {A2|Couper le courant sur la section|Actionner le commutateur DS au TCO}
- A2 : {A3|Retablir le courant sur la section|Actionner le commutateur DS au TCO}
- A3 : {A4|Confirmer un DS|Actionnement local d'un rupteur d'alarme}
- A4 : {A5|Autoriser RST sur la section|Remettre la barette du rupteur d'alarme en place}
- A5 : {A6|Reprendre le service voyageur|La rame repart avec voyageurs}
- A6 : {A7|Reprendre le service voyageur jusqu'au prochain T4|La rame repart avec voyageurs, effectue son service normal jusqu'a la station precedent le T4}
- A7 : {A8|Partir sans voyageurs|La rame repart sans voyageurs}
- A8 : {A9|Garer le train endommagé|Garer le train endommagé sur une voie secondaire du terminus en vu d'une revision}
- A9 : {A10|Reparer le train sur place|Suivre les indications du manuel du conducteur pour tenter de redemarrer le train}
- A10 : {A11|Evacuer les voyageurs d'une rame|Ouvrir les portes a quai et demander aux voyageurs de descendre}
- A11 : {A12|Evacuer les voyageurs d'une rame|Ouvrir les portes des voitures a quai et demander aux voyageurs de descendre en passant de voiture en voiture par les portes d'intercommunication}
- A12 : {A13|Evacuer les voyageurs d'une rame|Faire descendre les voyageurs sur les voies et les guider vers la station la plus proche}
- A13 : {A14|Canaliser des voyageurs|Diriger les voyageurs de la rame vers la station la plus appropriée}
- A14 : {A15|Rejoindre le train en detresse|Le train de secours rejoint le train en detresse}
- A15 : {A16|Lier les trains|Lier les deux trains}
- A16 : {A17|Convoyer|Le train de secours pousse le train en detresse jusqu'au terminus}
- A17 : {A18|Delier les trains|Liberer le train en detresse du train de secours}
- A18 : {A19|Rejoindre la prochaine station|Le train de secours rejoint la prochaine station}

A19 : {A20|Gerer le trafic residuel|Derer le trafic residuel en mettant des trains sous DSO, liberant des trains, organisant des SP...}

9 Activites

Act0 : {EvaRamVoies|Evacuer les voyageurs d'une rame}

Act1 : {EvaRamQuai|Evacuer les voyageurs d'une rame}

Act2 : {EvaRamAvar|Evacuer les voyageurs d'une rame en detresse}

Act3 : {EvaRamSec|Evacuer les voyageurs d'une rame de secours}

Act4 : {RepSerVoy|Reprendre le service voyageur jusqu'au terminus}

Act5 : {ArrSerVoy|Repartir sans voyageur}

Act6 : {FairConv|Ramener train non autonome en terminus}

Act7 : {PorSec|Ramener train non autonome en terminus}

Act8 : {IncTrac|Gerer l'incident traction}

GC 0 : {A2|A4|A13|A5|A3}

GC 1 : {A19|A11}

GC 2 : {{{\$Position par rapport a la station|a quai{A11}|partiellement a quai{A12}|en interstation{A13}}}}

GC 3 : {{{\$Position par rapport a la station|a quai{A11}|partiellement a quai{{{ \$Station libre entre les deux trains|oui{EvaRamQuai}|non{A12}}}|en interstation{{{ \$Station libre entre les deux trains|oui{EvaRamQuai}|non{A13}}}}}}}}

GC 4 : {A6|A9}

GC 5 : {A11|A8|A9}

GC 6 : {A15|A2|A4|A16|A5|A3|A17|A9|A18}

GC 7 : {A20|{EvaRamAvar|EvaRamSec}|FairConv}

GC 8 : {{{\$possibilite de reparation immediate|oui{A20|A10|RepSerVoy}|non{ \$Assez de puissance pour repartir seul|oui{{{ \$T4 avant terminus|oui{ RepSerVoy}|non{A7|ArrSerVoy}}}|non{PorSec}}}}}}

Annexe 3 :
Exemples de fichiers d'incidents

Annexe 4 :

Suivi de l'acquisition d'une pratique avec SAGIM

Dans cette annexe, nous présentons la saisie d'un incident avec SAGIM et l'acquisition de la nouvelle pratique dans la base de stratégie. L'incident que nous présentons ici (qui est virtuel, pour des raisons de confidentialité, mais entièrement plausible) est un dégagement de fumée sur matériel roulant (une rame) dû à un sabot de freinage en ignition.

L'incident est annoncé par THF par le conducteur de la rame qui croise la rame concernée à la station *Les Halles* où les deux rames stationnent. L'événement incidentel correspond ainsi à un *autre appel par THF*¹⁷, le chef de régulation entre alors en activité de *gestion d'un autre appel par THF*. L'annonce n'ayant pas été claire, le chef de régulation demande, dans un premier temps, au conducteur ayant fait l'annonce de réitérer celle-ci. Cette communication donne assez d'informations pour établir le diagnostic lié à la cause de l'incident et permet donc au chef de régulation de passer en phase de gestion d'un incident de type connu (voir deuxième chapitre, section 2.4.3, page 74). Ceci se traduit par un changement d'activité de l'opérateur (la nouvelle activité est : *traitement d'un incident d'élément de voiture en ignition*). Les actions entreprises consistent à faire visiter la rame par le conducteur, puis de tenter une extinction par serrage des freins. Cette manipulation étant infructueuse, l'utilisation d'un extincteur est préconisée. Cette action nécessite la coupure du courant de traction et la confirmation de cette coupure, puis, une fois le feu éteint, la déconsignation (autorisation de ré-alimentation) et la ré-alimentation de la section concernée. Nous détaillons maintenant l'utilisation du prototype de SAGIM pour la saisie de cet incident et la mise à jour de données du système.

Pages d'accueil

L'utilisateur croise d'abord deux pages d'accueil. La première donne accès aux trois modes d'utilisation de SAGIM : l'*archiviste*, le *conseiller* et le *courtier*.

¹⁷ Nous rappelons, comme précisé dans le deuxième chapitre (section 2.4.2, page 72) que deux types d'appel par THF sont différenciés : la demande de DG et l'annonce d'un KSA, car l'activité qui suit ces événements incidentels est très différente de l'activité faisant suite à un autre appel par THF.



L'exemple que nous présentons ici consiste en la saisie d'un incident et correspond donc au mode *archiviste*. La seconde page d'accueil est celle du mode *archiviste*. Elle propose la saisie d'incidents simple, double ou triple, correspondant respectivement à des incidents se manifestant par 1, 2 ou 3 événement(s) incidentel(s). Les événements incidentels orientent d'emblée l'activité des chefs de régulation et la survenue de plusieurs événements incidentels pour un même incident n'est pas exceptionnelle : elle modifie toutefois l'activité qui se trouve alors intermédiaire entre les activités liées aux événements incidentels simples. L'image suivante correspond à ce choix du nombre d'événement(s) incidentel(s) pour l'incident à saisir.



Saisie des premiers éléments concernant l'incident

La première page de saisie est alors un formulaire servant à renseigner les premières informations sur l'incident : date, heure, ligne, type de lieu concerné, événement incidentel, cause, répercussion sur le trafic et intervalles théorique et réel entre les rames à cet instant.



Description de l'incident

Date / Heure

Date

Heure de début de l'incident

Identification de l'incident

Ligne

Etendue de l'incident

Événement incidentel

Cause de l'incident

Estimations

Intervalle réel entre les trains (secondes)

Intervalle théorique entre les trains (secondes)

Répercution sur le trafic Peu perturbant

Perturbant

Très perturbant

Soumettre

En fonction des éléments donnés dans ce formulaire, des informations complémentaires sont demandées par SAGIM. Connaissant le type de lieu concerné (station, interstation, sous-section, section ou ligne entière), le lieu peut être précisé. Le système s'enquiert également de la valeur des paramètres de(s) l'événement(s) incidentel(s) et de la cause de l'incident auprès de l'opérateur. Enfin, en fonction de la répercussion sur le trafic, un nombre plus ou moins important de rames entourant la rame concernée risque de jouer un rôle actif dans la gestion de l'incident. Aussi SAGIM demande-t-il les numéros d'exploitation de ces rames.



Complement d'informations sur l'incident

Lieu de l'incident : Station(s)

Station(s)

Evénement incidentel : Autre appel par THF

Station
Positions par rapport à la station
Voie
Numéro d'exploitation de la rame
Numéro de matériel du train

Cause : Avarie au matériel roulant - Dégagement de fumée - Element d'une voiture en ignition

Numéro d'exploitation de la rame
Numéro de matériel du train
Position de la voiture dans la rame
Numéro d'identification matériel de la voiture
Element en ignition

Répercussions sur le trafic : Perturbant

Heure du début de la perturbation
Numéro d'exploitation du train précédent
Numéro d'exploitation du train suivant

Commentaire général sur l'incident

Incident annoncé par le conducteur du train d'en face.

Soumettre

Saisie des composantes de pratique liées à chaque activité

SAGIM propose ensuite à la fois de guider l'opérateur dans la saisie de la pratique (la suite des actions mises en œuvre) et de compléter sa base de stratégies en fonction de cette pratique. Cette saisie est réalisée activité par activité afin de structurer l'échange entre le système et l'opérateur (voir troisième chapitre, section 4.3, page 110). Pour chaque activité,

l'opérateur sélectionne le chemin d'action qui aurait dû être suivi (par la sélection des valeurs des connaissances contextuelles dans la structure d'activité enregistrée par le système), puis indique les points communs et différence entre la séquence d'actions et d'activités secondaires proposée et la composante de la pratique correspondant à l'activité en cours de traitement. Ensuite l'opérateur doit saisir les différences (quand il y en a) et les connaissances contextuelles sur la base desquelles sa nouvelle pratique est établie pour permettre au système d'intégrer cette pratique aux stratégies connues liées à l'activité concernée.

Activité de gestion de l'événement incidentel

Connaissant l'événement incidentel, SAGIM connaît l'activité initiale de l'opérateur : il s'agit de l'activité de gestion de(s) l'événement(s) incidentel(s) (*gestion d'un appel par THF*, dans notre exemple). Le choix du chemin d'action se fait par la sélection successive de la valeur courante des diverses connaissances contextuelles croisées dans le graphe contextuel enregistré par SAGIM pour l'activité en cours de traitement. Pour la gestion d'un appel par THF, l'activité consiste à déterminer quelle est la cause de l'incident et de passer à une activité spécialisée dans le traitement de ce type de situation. Le cas de notre exemple correspond à *AM.DFM.EVI – Element de Voiture en Ignition*.

The screenshot shows a web browser window with the URL <http://versa/SAGIM/Programme/Archiviste/Prototype/Test%20Victor%20>. The page title is "Action 'Gerer l'événement : Autre appel par THF'".

At the top left is the RATP logo. Below it, the text "SART / SAGIM / Archiviste" is displayed. The main content area features a context graph with nodes labeled: AM.MI, AM.PR.FL, AM.PR.NF, AM.PR.NO, AM.DFM, AM.DFM.EVI, A.Vieille, A.AutEclairage, and TE.ASE. A green dot is positioned on the "A.Vieille" node.

Below the graph is a section titled "Sélection des éléments de contexte". It contains a search box labeled "Cause" and a dropdown menu. The menu is open, showing a list of causes:

- Avarie Matériel
- AM.MI - Motrice(s) Inactive(s)
- AM.PR.FL - Fermeture Lente
- AM.PR.NF - Non Fermeture
- AM.PR.NO - Non-ouverture
- AM.DFM - Dégagement de Fumée sur le Matériel
- AM.DFM.EVI - Element de Voiture en Ignition**
- Objet supposé suspect
- AM.AEAU - Déclenchement intempestif de l'AEAU
- TE.ASE - Avarie à la signalisation d'espacement
- OV - Objet sur les voies

Une fois que le chemin d'action attendu est déterminé, SAGIM affiche la liste des actions et activités secondaires attendues et demande à l'opérateur de renseigner la composante de la

pratique correspondant à cette activité sur la base de la séquence affichée en décochant les actions attendues et non-réalisées et en ajoutant des actions inattendues réalisées.

Action "Gerer l'événement : Autre appel par THF"

Parcours sélectionné

Cause AM.DFM.EVI - Element de Voiture en Ignition

Traiter l'incident : Avarie au matériel roulant - Dégagement de fumée - Element d'une voiture en ignition (A compléter par la suite)

Vous devez décocher les actions non réalisées ou réalisées à un autre instant

D'autres actions ont-elles été réalisées ?

Si des actions supplémentaires sont annoncées par l'opérateur, le système demande combien d'actions ont été ajoutées entre chacun des composants de la séquence d'actions/activités attendues, comme montré ci-dessous.

Action "Gerer l'événement : Autre appel par THF"

Invite utilisateur de Internet Explorer

Invite de script :
Nombre d'actions ayant été réalisées avant "EC Cause"

1

L'opérateur doit ensuite sélectionner ces actions.

Contexte sélectionné

Cause AM.DFM.EVI - Element de Voiture en Ignition


Actions supplémentaires réalisées

Action réalisée avant l'élément de contexte "Cause"

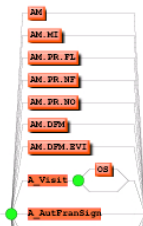
Une fois la composante de pratique déterminée, quand celle-ci diffère du chemin d'actions/activités attendues, l'opérateur doit indiquer quelle connaissance contextuelle à

guidé son choix. Il doit également donner la valeur de cette connaissance pour l'incident courant et le cas habituel.

Adresse <http://versa/SAGIM/Programme/Archiviste/Prototype/Test%20Victor%20> OK

 **SART / SAGIM / Archiviste**

Action "Gerer l'evenement : Autre appel par THT"



Contexte sélectionné Précédemment

Cause

Particularités du contexte actuel

Table n°1

Pour quelle raison l'action "Communication" a-t-elle été réalisée avant l'élément de contexte "Cause" ?

Elément de la situation à analyser :	<input type="text"/>
Valeur de cet élément de situation dans le cas présent :	<input type="text"/>
Valeur habituelle de cet élément de situation :	<input type="text"/>

OK

Contexte sélectionné Précédemment

Cause

Particularités du contexte actuel

Table n°1

Pour quelle raison l'action "Communication" a-t-elle été réalisée avant l'élément de contexte "Cause" ?

Elément de la situation à analyser :	<input type="text" value="Message compris"/>
Valeur de cet élément de situation dans le cas présent :	<input type="text" value="non"/>
Valeur habituelle de cet élément de situation :	<input type="text" value="oui"/>

OK

Une fois ces connaissances contextuelles renseignées, le graphe contextuel associé à l'activité en cours de traitement par SAGIM est mis à jour pour intégrer cette nouvelle composante de pratique. Ces étapes sont ensuite reprise pour chacune des activités

secondaires rencontrées dans la pratique de l'opérateur jusqu'à ce qu'elles soient toutes détaillées.

Le système enregistre la liste des activités à développer dans une pile. Ainsi quand une activité vient d'être traitée, elle est extraite du dessus de la pile et les sous-activités induites par celle-ci sont ajoutées à la pile de telle manière que la première développée soit la première dans l'ordre chronologique. Nous obtenons ainsi un système de développement des activités en profondeur d'abord, reprenant la chronologie des activités et actions.

Dans notre exemple, la composante de la pratique de l'opérateur concernant l'activité *gestion d'un appel par THF*, fait appel uniquement à l'activité *traitement de l'incident AM.DFM.EVI*, qui se trouve alors ajoutée à la pile des activités à développer (qui contient alors un élément), puis développée.

Activité de traitement d'un l'incident de cause connue

L'opérateur doit alors reprendre les trois étapes décrites ci-dessus pour l'activité suivante. D'abord la sélection de la valeur des connaissances contextuelles rencontrées pour établir le chemin d'actions/activités attendues.

Sélection des éléments de contexte (suite)

Visite complémentaire nécessaire	non
Element en ignition	oui
Extincteur disponible	non
Incident clos	non
Service voyageur possible	non

Continuer

Puis la définition de la composante de la pratique concernée par cette activité en se basant sur les actions et les activités attendues. Dans notre exemple, l'opérateur doit décocher

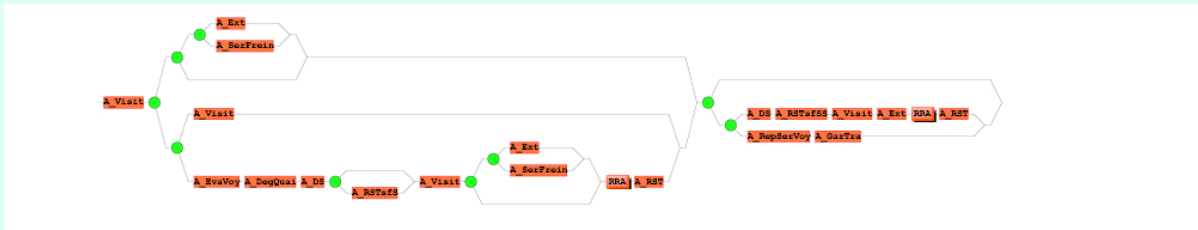
l'action *Remettre sous tension sauf une ou plusieurs sous-section(s)* et préciser qu'aucune action supplémentaire n'a été réalisée.



SART / SAGIM / Archiviste

Action "Traiter l'incident : Avarie au matériel roulant - Dégagement de fumée - Element d'une voiture en ignition"

dans "THF"



Parcours sélectionné

<input checked="" type="checkbox"/> Visite d'installations	
Visite complémentaire nécessaire	non
Element en ignition	oui
Extincteur disponible	non
<input checked="" type="checkbox"/> Serer les freins	
Incident clos	non
Service voyageur possible	non
<input checked="" type="checkbox"/> Couper le courant sur la section	
<input type="checkbox"/> Remise sous tension sauf une ou plusieurs sous-section(s)	
<input checked="" type="checkbox"/> Visite d'installations	
<input checked="" type="checkbox"/> Utilisation d'un extincteur	
<input checked="" type="checkbox"/> Remise d'un rupteur d'alarme (A compléter par la suite)	
<input checked="" type="checkbox"/> Remise sous tension de toute la ligne	

Veuillez décocher les actions non réalisées ou réalisées à un autre instant

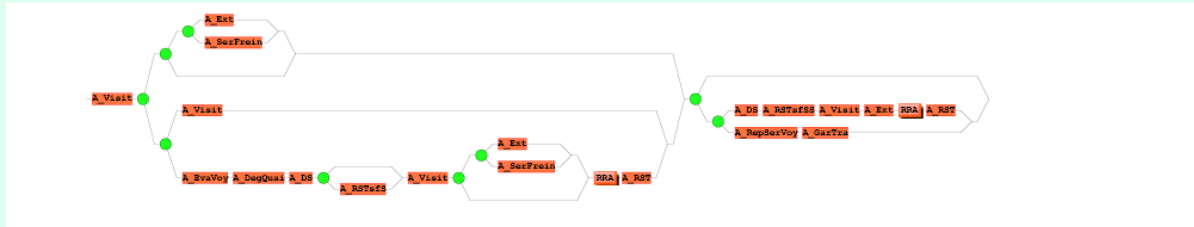
D'autres actions ont-elles été réalisées ?

Enfin, la saisie de la connaissance contextuelle guide ses choix stratégiques. Dans notre cas, la remise sous-tension des sous-sections non-concernées par l'incident n'est pas nécessaire car la répercussion de l'incident sur le trafic est faible.



Action "Traiter l'incident : Avarie au matériel roulant - Dégagement de fumée - Element d'une voiture en ignition"

dans "THF"



Contexte sélectionné Précédemment

Visite complémentaire nécessaire	non
Element en ignition	oui
Extincteur disponible	non
Incident clos	non
Service voyageur possible	non

Particularités du contexte actuel

Table n°1

Pour quelle raison aucune action n'a été réalisée, à la place de l'action "Remise sous tension sauf une ou plusieurs sous-section(s)", entre l'action "Couper le courant sur la section" et l'action "Visite d'installations" ?	
Élément de la situation à analyser :	Répercussion sur le trafic
Valeur de cet élément de situation dans le cas présent :	faible
Valeur habituelle de cet élément de situation :	importante

OK

Ces informations permettent de compléter le graphe contextuel représentant la stratégie de traitement de l'incident. La composante de la pratique correspondant à cette activité ne fait appel qu'à une seule activité : *RRA*. Celle-ci est alors ajoutée à la pile (qui ne contient encore qu'un seul élément puisque l'activité qui vient d'être développée a été ôtée).

Autres activités

Cette dernière activité (*RRA*) correspond en fait au suivi d'une procédure stricte précédant la remise sous-tension d'une section et n'offre que peu de degré de liberté à l'opérateur. Ce dernier se contente alors de sélectionner le chemin d'action suivi et de confirmer les options proposées par SAGIM. D'autres activités secondaires offrent une plus grande variabilité, mais plus nous avançons dans les détails, plus les activités sont techniques et donc proches des procédures.



Action "Remise d'un rupteur d'alarme"

dans "THF" => "AM.DFM.EVT"



Sélection des éléments de contexte (suite)

Etat du(es) voyant(s) du(es) RA allumé

Etat du(es) voyant(s) du(es) RA
éteint
allumé
- Autre cas -



Action "Remise d'un rupteur d'alarme"

dans "THF" => "AM.DFM.EVT"



Parcours sélectionné

<input checked="" type="checkbox"/> Remise d'un rupteur d'alarme
Etat du(es) voyant(s) du(es) RA allumé
<input checked="" type="checkbox"/> Remise d'un rupteur d'alarme
Etat du(es) voyant(s) du(es) RA éteint

Veillez décocher les actions non réalisées ou réalisées à un autre instant

D'autres actions ont-elles été réalisées ?

Saisie des paramètres des actions réalisées

Quand la saisie de la dernière activité est terminée, SAGIM propose alors de continuer la saisie de l'incident. Cette saisie correspond au renseignement des paramètres des actions réalisées. Cette saisie est guidée par un unique formulaire présentant toutes les actions réalisées dans un ordre chronologique et rassemblées par activités. Chaque action est présentée avec les champs correspondant à ses paramètres spécifiques. La structure des activités est représentée par un emboîtement de cadres, chacun d'eux regroupant les actions et activités secondaires impliquées dans cette activité. L'image suivante est le formulaire correspondant à l'exemple traité et se poursuit sur les pages suivantes.

Complement d'informations sur les actions locales

Détail de l'action Gerer l'événement : Autre appel par THF

Communication

Heure de l'appel

Correspondant

Message

Détail de l'action Traiter l'incident : Avarie au matériel roulant - Dégagement de fumée - Element d'une voiture en ignition

Visite d'installations ou de matériel

Heure de début de la visite

Personne(s) effectuant la visite

Objet de la visite

Heure de fin de la visite

Résultat de la visite

Serer les freins

Heure

Mise hors tension par section

Heure de la mise hors tension par section

Sections mises hors tension
Porte de Clignancourt - Barbes Rochechouart
Barbes Rochechouart - Cite
Cite - Saint-Placide
Saint-Placide - Porte d'Orleans

Personne ayant demandé la mise hors tension

Rupteurs d'alarme actionnés pour confirmation
1.1 - Rupteur boucle Porte de Clignancourt
1.2 - Rupteur boucle Porte de Clignancourt
1.3 - Rupteur boucle Porte de Clignancourt
1.4 - Rupteur boucle Porte de Clignancourt

Visite d'installations ou de matériel

Heure de début de la visite

Personne(s) effectuant la visite

Objet de la visite

Heure de fin de la visite

Résultat de la visite

Utilisation d'un extincteur

Heure

Extincteur utilisé

Résultat Feu maîtrisé

Détail de l'action Remise d'un rupteur d'alarme

Remise d'un rupteur d'alarme

Heure de la remise en place du(des)
rupteur(s) d'alarme

Sections concernées par ce rupteur
d'alarme
Porte de Clignancourt - Barbes Rochechouart
Barbes Rochechouart - Cite
Cite - Saint-Placide
Saint-Placide - Porte d'Orleans

Numéro du rupteur actionné 1.1 - Rupteur boucle Porte de Clignancourt

Personne ayant actionné le rupteur
d'alarme

Etat du(es) voyant(s) éteint(s)

Remise d'un rupteur d'alarme

Heure de la remise en place du(des)
rupteur(s) d'alarme

Sections concernées par ce rupteur
d'alarme
Porte de Clignancourt - Barbes Rochechouart
Barbes Rochechouart - Cite
Cite - Saint-Placide
Saint-Placide - Porte d'Orleans

Numéro du rupteur actionné 1.1 - Rupteur boucle Porte de Clignancourt

Personne ayant actionné le rupteur
d'alarme

Etat du(es) voyant(s) éteint(s)

Remise d'un rupteur d'alarme

Heure de la remise en place du(des)
rupteur(s) d'alarme

Sections concernées par ce rupteur d'alarme

Porte de Clignancourt - Barbes Rochechouart
Barbes Rochechouart - Cite
Cite - Saint-Placide
Saint-Placide - Porte d'Orleans

Numéro du rupteur actionné

Personne ayant actionné le rupteur d'alarme

Etat du(es) voyant(s)

Remise sous tension de toute la ligne

Heure de la remise sous tension générale

Commentaire sur les actions menées

Nombre d'actions globales réalisées

Saisie des autres informations concernant l'incident

Une fois ce formulaire rempli, SAGIM demande des informations sur les actions globales réalisées. Ces actions concernent indirectement l'incident et permettent de gérer le trafic résiduel (voir deuxième chapitre, section 2.4, page 72). Cette saisie est opérée comme les premières informations en deux temps : la sélection des actions, puis le renseignement des paramètres des actions sélectionnées.



Description des Actions Globales effectuées

Actions Effectuées

Action effectuée

Action effectuée

Action effectuée

Action effectuée

Soumettre

- Mise sous DSO
- Ordre de départ (fin DSO)
- Faire dériver un train
- Evacuer une rame
- Etablissement d'un service provisoire
- Trafic interrompu
- Reprise du trafic



Description des Actions Globales effectuées

Actions Effectuées

Action effectuée

Action effectuée

Action effectuée

Action effectuée

Soumettre



Complement d'informations sur les actions globales

Mise sous DSO

Heure de mise sous DSO

Numéro d'exploitation

Station

Voie

Mise sous DSO

Heure de mise sous DSO

Numéro d'exploitation

Station

Voie

Ordre de départ (fin DSO)

Heure de l'ordre de départ

Numéro d'exploitation

Station

Voie

Ordre de départ (fin DSO)

Heure de l'ordre de départ

Numéro d'exploitation

Station

Voie

Commentaire sur les actions globales menées

Soumettre



Complement d'informations sur les actions globales

Mise sous DSO

Heure de mise sous DSO
Numéro d'exploitation
Station
Voie

Mise sous DSO

Heure de mise sous DSO
Numéro d'exploitation
Station
Voie

Ordre de départ (fin DSO)

Heure de l'ordre de départ
Numéro d'exploitation
Station
Voie

Ordre de départ (fin DSO)

Heure de l'ordre de départ
Numéro d'exploitation
Station
Voie

Commentaire sur les actions globales menées

Soumettre

Une fois les actions globales et leurs paramètres saisis, l'opérateur complète un formulaire sur les conséquences de l'incident, notamment le retard cumulé et l'intervalle au départ.



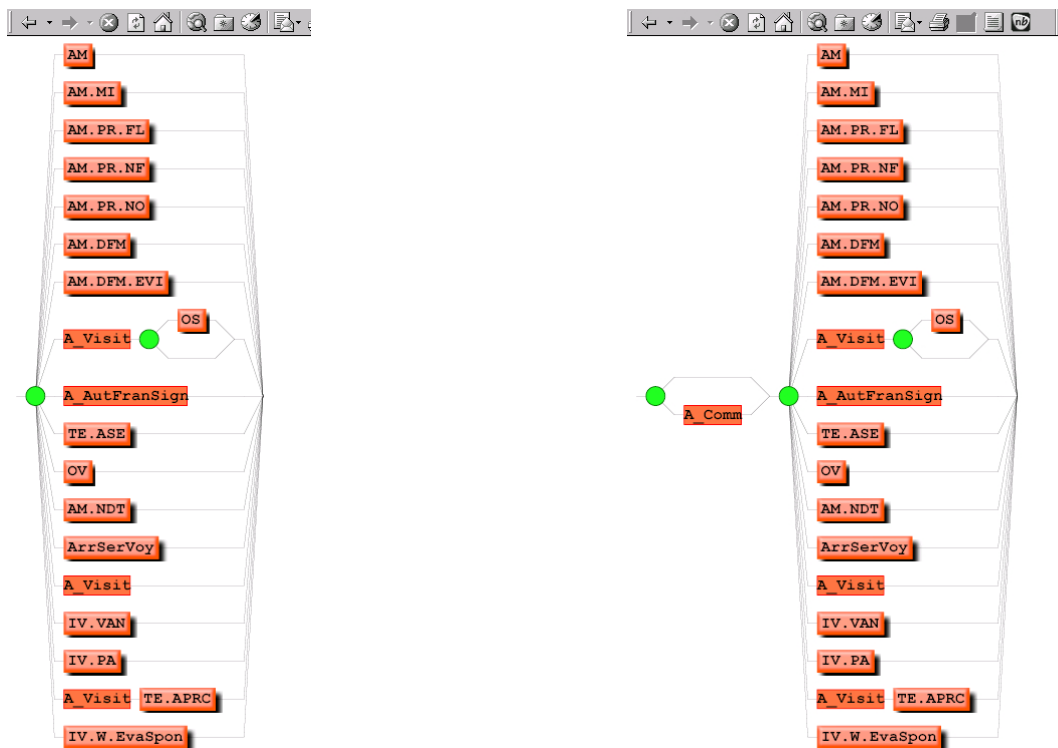
The screenshot shows a web browser window with the address bar containing the URL: [http://versa/SAGIM/Programme/Archiviste/Prototype/Test%20Victor%](http://versa/SAGIM/Programme/Archiviste/Prototype/Test%20Victor%20). The page header features the RATP logo and the text "SART / SAGIM / Archiviste". The main content area is titled "Conséquences" and contains the following form elements:

- Heure de fin de l'incident:
- Retard cumulé (minutes):
- Intervalle au départ (secondes):

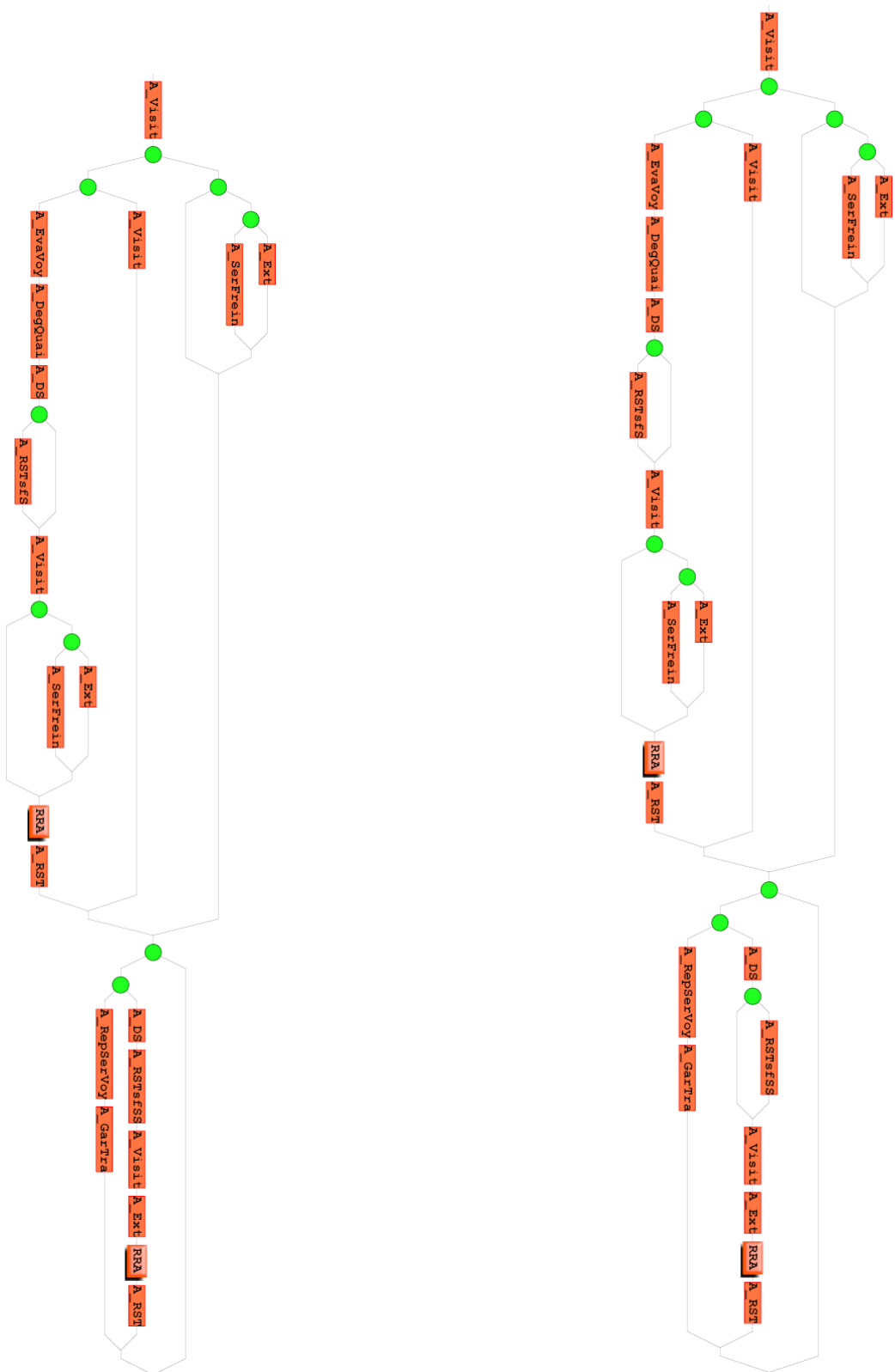
Commentaire final

Modification des graphes contextuels lors de l'acquisition de cet incident

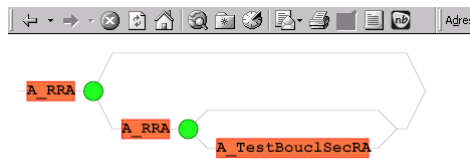
Dans cet exemple d'acquisition, trois activités entrent en jeu. La première correspond à la gestion de l'événement incidentel *Autre appel par THF*. Cette activité est modifiée par la pratique mise en œuvre par l'opérateur car ce dernier n'ayant pas compris le premier message du conducteur lui demande de réitérer sa communication. Ainsi le graphe contextuel s'enrichit d'un fuseau composé d'un nœud contextuel étiqueté « Message compris » d'où partent deux branches étiquetées respectivement « oui » et « non ». La première branche ne porte aucune action, la seconde porte l'action « Communication ». Ces deux branches se rejoignent en un nœud de recombinaison précédant le nœud contextuel établissant le diagnostic sur la cause de l'incident. Les deux images suivantes montrent les graphes contextuels de cette activité avant et après l'acquisition de cette composante de pratique.



La seconde activité en jeu concerne le traitement de l'incident de cause connue « Élément de Voiture en Ignition ». La pratique mise en œuvre par l'opérateur pour cet incident ajoute un nœud contextuel et un nœud de recombinaison dans la deuxième branche du deuxième fuseau. Ce nœud contextuel permet d'éviter une remise sous-tension partielle de la section quand le trafic n'est que faiblement perturbé par l'incident.



La dernière activité correspond à une procédure très technique directement liée à la sécurité et la pratique pour cet incident ne la modifie pas. Le graphe contextuel correspondant reste inchangé.



Traductions de l'introduction et de la conclusion

Introduction

Framework of the study

The very first goal of computers was to help man in long and fastidious calculations. The first operational computing automaton, Pascal's Machine, was a good starting point in the development of the first computers. Turing's theory on universal machines, the definition of Von Neumann's architecture and the evolution of the electronics were key points in the development of modern computing technologies. The latter one allowed us to assist, at the same time, to the development of computer science. In the 1980's, the development of the economic field of home computers accelerated the development of computers in number as well as in computing power. The computers go beyond their very first simple function of computing machines. The current goals are more ambitious. They now have to simulate complex systems such as interactions between numerous elements, sociological phenomenon, biological systems, decision processes...

Just after World War Two the science field of Artificial Intelligence (AI) took off. Its aim is to reproduce intelligent behaviors. This research field remained purely theoretical during the first half of the twentieth century. The advent of electronic universal computers allowed to implement the theories obtained previously. AI, in its diverseness, searches either to make systems autonomous or to help man in his daily activities. To achieve this, AI draws its inspiration from human nature: reasoning, sociology, psychology, cerebral structure; or from surrounding nature: ethology, interactions between species, evolution... Several ways are explored and numerous specific programming languages accelerate the development of research projects. In the 1970's, the first expert systems are developed. They are based on rules and a logical motor. They allow to model highly structured human reasoning, based on propositional logics.

The design of expert systems is strongly linked to research fields on cognition, which study the intelligence of the human beings. These human sciences show that “the reasoning” can not just be limited to simple rule-based reasoning, such as those used by the first expert systems. Propositional logic is not very natural for man. It must be made more flexible to develop more accurate systems, by taking into account more complex notions inspired from human logic. In this scope AI proposed fuzzy, modal or temporal logics.

The design of modern expert systems, called *knowledge based systems*, needs to collect and structure knowledge on the considered application field as well as on the reasoning followed by the experts. This stage of analysis can be carried out in two ways. First the designer can define the different operations realized by the expert, and can then analyze their interaction and their organizational structure. This engineering approach has an upward trend: the designer starts from the smallest element (the elementary operations realized by the expert) to finally lead to the most complex elements (the activities and the acting rules). Numerous methodologies were developed on this idea, such as KADS [Wielinga, Schreiber and Breuker, 1992], KARL [Fensel, Angele and Landes, 1991] or De-KART [Baudin and Pell, 1994] (among others). Secondly the designer can analyze the experts’ activities and deduce the elementary actions and the underlying reasoning. This cognitive approach has a downward trend: the conceiver starts from the more complex elements (the expert activities) to finally reach the most simple ones (the elementary actions). Both approaches are complementary, but they are rarely jointly used. The first one is most often used by computer scientists, the second by ergonomic specialists.

Regardless of the approach chosen during the conception of the expert systems, one can underline the major role played by the context in many daily situations. It intervenes in decision taking or in the understanding of situations, objects and talks. Let’s first make a simple remark about talks, its context and its understanding. Everyone entering a debate in the middle of a discussion needs some time to give a sense to each sentence. This period of adaptation allows the newly arrived participant to gather the elements necessary to understand the conversation previously shared by the other participants. These elements are part of the context of the discussion. Concerning objects and situations, the context plays a particular role, namely when we try to categorize them. The result of the categorization highly depends on the point of view chosen and thus on the context in which the categorization is done. The importance of the context on the understanding of the world have great consequences in decision making. The similarity between several situations depends of the context, and reasoning based on analogy must take the role of the context into account. Deductive reasoning must also take into account the context from the definition of their rules, to be accurate enough. The context plays a capital role at different levels during the decision

process. Numerous current works aim to integrate this notion to render the expert systems more flexible in response to the different encountered situations.

Concurrent to the expert systems, decision support systems were developed. The purpose of these systems is to help the users achieve a task. They are more high-level tools given to users than assistants. Guy Boy [1991] proposes the design of intelligent assisting systems. These systems are halfway between the decision support systems and the knowledge-based systems: they enlarge the role of support systems with the capacity and functionality of knowledge-based systems. They are not autonomous systems but try to assist the operator in his daily activities. In this scope, they don't have the constraint of being complete systems like the autonomous ones; they can acquire new knowledge through interaction with the users. They are however based on the same principles than knowledge-based systems. These principles give them some reasoning and knowledge manipulation abilities.

The different solutions proposed by tools based on computer science are operated around us in several application fields. Particularly in transportation (airway or railway, for freight or travelers, urban or interurban), as shown by the success of the PREDIT French research program. The computers "super-calculating machines" give way to more sophisticated tools. Several automated tasks are simply supervised by operators and controlled by autonomous systems. The non automated tasks may nevertheless be computer-assisted. This is the case of complex tasks such as regulation of operations or dysfunction diagnosis. Today, computer knowledge and techniques allow to propose assisting tools for tasks which require a high human expertise level and poorly formalized such as incident management.

Problematics

Traffic regulation in operation of a subway line is a difficult task, where each decision is taken according to the current situation. In Paris the operators in charge of this task are all experts. They perfectly know the system to regulate : the subway line, its operation, the possible incidents, the consequences of their decisions and actions... Managing operation incidents occurring on the line is without a doubt the most complex activity of these operators. Headway is short (90 seconds during peak-hour) and operators only have short time to elaborate a strategy to eradicate the cause of the incident. They have to take into account more or less partial knowledge that they have on the situation (cause of the incident, conditions of the traffic, external events ...) and have to respect the rules in usage (railway regulation, safety rules, commercial role of the company...). The performances of the operators are highly linked to their expertise. This expertise is acquired in two time : first the operators follow an initial training to learn the skill of subway regulation, during which they acquire the theoretical bases ; secondly, throughout their career at this position, they continue

to learn through their experiences, those of their colleagues and through experience feed-back sessions organized by their managers.

This field experience is not a general rule : other systems are less restricting and more technical. Such is the case, namely, in Rio de Janeiro. The system is recent and not very dense. The operation is carried out with less trains and the regulator has more time to react. The interface of the regulation work station is more complex and the manager has chosen to employ young engineers for this activity. The regulators follow a substantial primary training, but the lack of field experience can cause some problems.

The RATP (Paris Public Transportation Authority) has been one of the leading company in the development of subway systems in many cities throughout the world. The expertise acquired during the construction of the different Paris subway lines has helped in the realization of many subway networks in very different conditions. These subway networks are thus more modern. Companies that operate them become competitive on the international scene and are even more efficient. To maintain its rank in the international market of design and realization of metropolitan railway networks, the RATP has to assert its centenary experience. Indeed, after a century of operation, the RATP has acquired a great experience on traffic regulation and incidents management. The quantity of incidents already encountered and the quality of applied solution are of great value. It is necessary for the company to assert this experience to maintain its position on the international scene.

In 1996, two executives of the Rio de Janeiro's subway central control room, undertook PhD theses in France. They permit thus to initiate a Franco-Brazilian project for the design of a Traffic Regulation Assistance System (SART, in French). The goal of this project is to realize a cooperative decision support system for operators in charge of traffic regulation and incidents management, for Rio as well as for Paris. However, even if the general principles are the same (RATP has participated in the design of Rio's subway), great differences exist. First, as previously mentioned, the rhythm is not so much sustained in Rio than in Paris. Secondly, operators don't have the same types of expertise in the sense of Prince [1996] : in Paris operators are issued from the field, they have a typical reflex expertise; in Rio operators are newly-graduated engineers, they have a bookish expertise. Thirdly, many installations are specific to the networks (air conditioning of stations in Rio, defrosting of rails in Paris...), as well as some actions, strategies or procedures.

These differences are more or less restricting. The difference in the type of expertise modifies the interface and the type of information that can be provided by the support system. These are major modifications in the functioning of a part of the system. The other differences are simpler to manage by conceiving a system where most of the knowledge is declarative.

All these differences in operation between Paris' and Rio's systems increase the importance of the context and its management by SART.

More generally, SART is a support system : it has to be conceived as an expert system, but its role is to support the operator in his/her daily activities, without taking the decisions in his/her place. SART is interactive : it has to be able to inform the operator on precise points by providing relevant and targeted explanations, it has also to be able to assimilate by questioning the operator. These interactions with the operator have to be realized in a language comprehensible by the system, SART must thus be able to take into account the situation to provide useful explanations to the operator.

The operators concerned by SART base their actions on the context and its richness. The operators construct strategies to manage incident starting from technical data on the operation of the line and from the procedures, but has to adapt them to the real situation. Procedures can not take into account all of the richness of the met situations, but the context of incidents to process influences greatly strategies of management. Thus two identical incidents are not processed in the same way if they occur during peak-hours or in off-peak-hours. The time pressure being lesser in off-peak-hours, the operator can conceive longer strategies. Elements of the context such as the type of hour (peak or slack) which intervene in the choice of strategies are numerous in the case of incidents management on a subway line.

The notion of context is important in operators' decision process. It is therefore necessary to integrate the context in the knowledge and the reasoning followed by SART. The design of knowledge-based systems necessitates the modeling of knowledge encountered in the application field and of the reasoning followed by the experts. We want to introduce the notion of context to the deepest of our system. This is why the modeling of knowledge of the application field around the notion of context has been realized by Saker [1999] and that we propose, in this thesis, to model the reasoning followed by the operator by integrating the context notion.

Approach

The task of the operators is complex. We propose to use a downward design. We begin with the observation and the analysis of the activity of operators. The results of this analysis give a structure of the reasoning followed and the organization of the sub-tasks and actions realized by operators. We have modeled the structure of their activity by graphs representing the different manners to reach a desired goal. We associated these graphs with a system that decomposes task in sub-tasks. Each used graph represents the different methods associated with a given task. These different methods are distinguished by the situation in which they

may be applied. Graphs that organize these different methods are therefore strongly linked to the notion of context and show the importance of the context in strategic choices realized by the operators. These graphs are therefore called *contextual graphs*.

The contextual graph associated with the task/sub-task decomposition gives a representation formalism of the activity quite close to what can be observed besides the operators. Moreover we obtain, in the same formalism, different levels of detail for the activity and its organization. This formalism is well accepted by the operators because it is based on a reasoning mode that they follow in practice. This has two main advantages : firstly the assistance and explanations provided by the system are directly assimilated by the operator and given at the right level wished by the former ; secondly the insertion, by the operator, of new methods to reach a goal or a sub-goal is facilitated.

This approach can appear not really original. Generic tasks defined by Chandrasekaran [1986], the componential approach of Steels [1992] or the heuristic classification of Clancey [1979; 1985] (among others) propose similar decomposition mechanisms. The originality of our approach lies in the taking into consideration the dynamic aspect of the decisions taken by the operators. Contrarily to generic tasks where methods of decomposition are fixed once and for all during the programming of the system, we opt for a dynamic selection of the decompositions in sub-tasks during the elaboration of the strategy, and this, so as to increase the sensitivity of the system face to the context. Thus our goal is not to conceive a "turnkey" system that would claim to know the solution for each incident to manage in the different possible contexts. We prefer to propose a flexible system constructing possible strategies for each given incident in a given context. The construction of strategies by the system then rests on the model of tasks/sub-tasks decomposition according to the context. The flexibility of the system is strengthened by the possibility of the new practices integration realized by operators during the phase of incremental knowledge acquisition in incidents management. This integration follows immediately an incident management by the operator.

Concerning our approach in terms of the way we design the system, starting from the observation of operators' activities, we obtain both the possible elementary actions and the reasoning followed for the elaboration of incidents management strategies. Starting from a rather cognitive approach, we end up to a formalism of reasoning representation authorizing a rising phase of design. This second phase consists in bringing to the fore the elementary actions (facilitated by the first phase of design) then to organize these actions in structures representing the different explanatory levels of activities for incident management. Our approach is therefore intermediate to the two main approach for the design of knowledge-based systems. It is situated, thus, at the frontier between cognitive sciences and the necessary knowledge engineering for the development of an operational system.

Plan

In first chapter, we make a point on the expert system evolution : from first generation systems based on rules to second generation systems based on knowledge, of which we analyze the limits. The context plays a capital role in many human expertise, it is therefore necessary to take it into account during the realization of expert systems, decision support systems or autonomous systems evolving in a complex environment. In the framework of a decision support system the context plays in addition a particular role for explanations provided to the user. We show the role of the context as much for decision-making than for explanations. We also propose an expert system model oriented toward to the user : Intelligent Assistance Systems in Context (SAIC, in French) [Brézillon, 1999a] to make up for the failings of current expert systems.

We propose thereafter to apply the model of SAIC to a particularly sensitive application field : the incident management on a subway line. The subway of Paris is a complex system where the main word is "security". Indeed the subway is an underground railway system for passenger transportation. This complicates incident management and forces to have a particular vigilance concerning the power supply of installations and the management of passengers. We present, in second chapter, the Paris subway and the incident management on a subway line. We first present the history of the subway and its structure. Then, we present the task of incident management, the staff in charge of this task, their activity... We put a particular accent on the importance of the context in these activities.

Third chapter is dedicated to our work on a model of the reasoning held in context. We present the contextual graph model : their constitutive elements, their structure, how they allow to model and to represent various reasoning held in context. We also show the capacities of evolution of the models obtained and how new reasoning are taken into account thanks to a system of incremental knowledge acquisition for the reasoning. We show finally how to apply this model to the reasoning observed besides the operators concerned by the support system that we conceive and establish thus a bridge between two important disciplines : Artificial Intelligence and Cognitive Ergonomics.

In the fourth chapter, the contextual graph model and its association to a mechanism of tasks/sub-tasks decomposition is compared to the various works already undertaken or under way. We analyze particularly works related at the knowledge level and those related at the programming level. Among others, we particularly discuss about Frames, Scripts, Generic Tasks, Diagrams, Scenarios, Dynamic Planning, Schemes of Action... We study for each of them the common points and differences with contextual graphs, and what our model brings. Our approach, not really original at first view, appears innovative on several important points.

This chapter shows principal advances compared to related works, whatever their areas are, and shows the federative aspect of our model.

The fifth chapter details, at a semantic level, the use of contextual graphs to represent human activities. We apply the model to the reasoning observed among the operators concerned with the support system we design. We thus establish a bridge between two important research fields: *Artificial Intelligence* and *Cognitive Ergonomics*. This chapter points the context-based reasoning: We essentially discuss about the link between tasks and activities and then about contextual strategic reasoning. Then, we detail how contextual graphs might be used to represent activities. We also come back on the incremental knowledge acquisition and precise the consequences of activity representation on the algorithm described in the third chapter. Next, we compare more precisely the model we propose (activities representation through contextual graphs) with the schemes of action exhibited by cognitive sciences. We end this chapter with the consequences of such a representation on the design of a SAIC.

The last chapter presents the prototype build on these researches and the results obtained during the validation stage in real situation. This system is a SAIC built around the hybridization of a contextual graph based reasoning and a Case-Based Reasoning. We precise here the consequences of the introduction of the context at a deep level into the system for the implemented reasoning and for the knowledge manipulated. After a first section placing our system into the SART project, this chapter presents the functionalities and the structure of our system. We develop thereafter the choices realized while programming for the two modes of reasoning hybridized. We explain in particular the great flexibility obtained thanks to the massive usage of declarative knowledge. We end up with the analysis of the results obtained in real situation during the validation stage.

In the conclusion, we discuss possible evolution of SART. The design anticipates these evolution notably concerning the application to Rio de Janeiro. In Paris, a new type of control room for the subway has been put into operation for line number 4. The Incident Manager of SART has been tested on this new control room, with the modifications of some knowledge declaration files. Currently the RATP works on the renovation of the control room of line number 13 that could become another test place. All the more interesting is that line number 13 is distinguished from the other lines on several points : the line has a junction, the two branches have very different length, it has several intermediary terminal stations... Another extension of the system is currently studied concerning the Regional Express Network (RER, in French). The RER is distinguished from the subway by the presence of many rail junctions, the trains are not always omnibus, the frequency of trains is weaker, there are two companies operating the system (the RATP and the SNCF)...

Conclusion

A metropolitan train line is a highly complex dynamical system. Lot of operation incidents occur every day. The quantity of incidents treated since the beginning of the Parisian subway operation is the foundation of the great experience of incident management acquired by the company. This acquired experience of incident management is translated into procedures which form a frame to diagnosis and actions that must be done to manage an incident or a particular state of the system (formed by the line and the trains). These procedures are compiled forms of the knowledge implied in incident management but are too general to be directly applicable to any particular situation. For each incident, the regulators adapt these procedures to the real situation to take the particularities of the context into account. The practices they apply in this way are different, for a same incident, from one situation to another and from one regulator to another.

The adaptation of the procedures by the regulators is made possible by their great field experience and the experience acquired as regulator. The latter comes from the situations directly treated by the regulator or from discussions with his/her colleagues about particular situations. Since the regulators of all the Parisian subway lines work in the same room, information about particular situations or original strategies are rapidly shared by all of them. Thus, the regulators form a community of practice. At this time, central control rooms tend to be decentralized, so that this community is in danger. The exchanges between regulators of different lines will not be as efficient as nowadays. The risk is the appearance of as much community of practice as subway lines. Nevertheless a great part of their experiences are useful to the whole community, although each line have its specificities. The development of knowledge management tools becomes necessary.

SART project is a tool which can establish a connection between several control rooms and permit thus to maintain the unity of the community of practice. This project aims the development of a support system for traffic regulation. We especially worked on the incident manager, which is one of its components, also called SAGIM (French acronym of support system for incident management in the subway). SAGIM is a Context-sensitive Intelligent Assisting System (SAIC in French). Its goal is to gather knowledge about incident management as it is applied by the regulators, to organize it, and then to use it to help a regulator when an incident occurs or to propose experience feedback sessions.

The architecture of a SAIC is intermediate between Interactive Decision Support Systems and Intelligent Assisting Systems. They inherit the design based on Knowledge-Based Systems (or second generation Expert Systems) from the former. The latter brings a global overview of the interactions between the system and the user, defining an assisting role for the

system. Thus the system have in charge several tasks that can be automated, such as surveillance or advice. Both underlying systems need user models to correctly target the support or the assistance provided. Thus SAIC also need such models.

Moreover, SAICs place the context in the center of their functionalities. Context plays an important role in numerous human reasoning. It is thus necessary to introduce it in support systems for highly situation-dependent tasks. SAIC architecture take up a great part of the one of the Knowledge-Based Systems: knowledge about reasoning allow the system to manipulate knowledge about application field to propose solutions to an occurring problem. The introduction of context intervene at each level: application field knowledge, reasoning knowledge, user models... Earlier works introduced the context into application field knowledge. This thesis relates the introduction of context into reasoning models.

SAGIM is thus based on two context-based reasoning modes. First we use a computerized model for analogy reasoning: le Case-Based Reasoning. This reasoning take naturally context and its consequences on the proposed solutions into account. The work done by Mignot [1997] about context-dependent similarity measures is included in SAGIM. Nevertheless, we introduce an originality in our application of this reasoning: The number of incidental situations encountered in the subway is significant and most of them differ on details consequence-free on their management; others, rare, are original and some incidents are even exceptional (indeed unique). Lot of occurring incidents do not produce any new knowledge (the great majority of cases) and sometimes original incidents do so. We thus deal with two types of incident bases in SAGIM: first an historic incident base which contains every occurrence (for statistical reasons) and second a typical incident base which contains every occurrence producing new knowledge (which is the basis of the Case-Based Reasoning module).

The second reasoning mode used by SAGIM is based on the modeling of regulators' activities using contextual graphs. Activities are the expression of the reasoning put into practice for the planning of diagnosis and actions for the realization of a particular task. Numerous studies (in cognitive psychology and artificial intelligence) propose a task/sub-task decomposition and the organization of several possible methods for the realization of each task. This lead to a hierarchy of tasks/sub-tasks more or less adapted to the encountered situations. Nevertheless, these decompositions do not address operators' activity linked to each task. This activity is not, as supposed by the classical decomposition, composed of one diagnosis followed by action plans, but present an intertwining of both components. Classical models do not allow the representation of such diagnosis/action structures.

We thus propose the contextual graphs model to represent such structures where decision and actions are deeply linked. These graphs are made of decision nodes (where a contextual

element is analyzed to choose between several alternatives) called contextual nodes, action elements (which refer to an action to do) and activity elements (which refer to a secondary activity). These graph elements are linked by oriented arcs representing a time dependence: a graph element precede another when the first have to be done before the realization of the second. Temporal branchings are available to precise when diagnosis/action sub-structures are timely independent and might be managed in parallel. Thus contextual graphs represent the evolution of the situation and of the knowledge about the situation, function of the realization advancement of the corresponding task and of the choices previously made.

Contextual graphs have exactly one root representing the starting point before any diagnosis. As long as no diagnosis is made, the situations are not differentiated and the treatment is identical whatever the situation is. Contextual graphs have exactly one exit point because every different action paths contained into the graph represent a different manner to achieve a common goal and thus to attain a wished situation. Thus, contextual graph structure present series of spindles. Each spindle represent a local contextual reasoning where contextual pieces of knowledge are pointed to select an adapted action path. The identified contextual knowledge represents the diagnosis on the situation; and depending on this diagnosis, choices of specific diagnosis/action plans are done.

Contextual graphs are thus a model allowing the representation of diagnosis/action cognitive structures, as the one discussed above, and might represent such activities observed among humans. Those activities, expressing the realization of tasks, are also hierarchically organized. We have the same type of decomposition as with tasks and sub-tasks. However, we do not address the tasks but the activities induced by those tasks. We propose a model representing the interactions between diagnosis and actions and the dynamical link between these interactions and the context.

Moreover, the context is explicitly represented in our formalism. Most of the represented strategic choices are based on contextual elements. Thus contextual pieces of knowledge are incorporated into the diagnosis and influence directly action plans. Our representation formalism is based on the context and its dynamism during the construction of the diagnosis/action plan.

We also propose an incremental strategic knowledge acquisition algorithm which allow the introduction of new practices corresponding to a task into the contextual graph organising the corresponding activity. This algorithm is based on high interactions with the operator. The system guides him/her in this operation and make emerge deep knowledge of which the operator was not conscious. This guided introspection is beneficial for the operator because it structures his/her own knowledge on the reasoning involved in his/her activity. Operators' expertise is thus acquired step by step by the system and organized for a future reuse.

The obtained model is very close to action schemes, which are cognitive structures observed among subway regulators (and among lot of other human activities). These structures are activity organizers. They contain information about conditions and means necessary to the realization of the activity, the expected results, reasoning that can be applied and decision/action plans. These cognitive structures evolves in respect of past experiences to integrate the particularities of each situation to increase their efficiency. This proximity between the cognitive structures observed among the regulators and the artificial reasoning model we propose explains the rapid assimilation of the model and of its functioning by the operators.

The application, in the prototype of SAGIM, of this modeling of regulators' activities is successful. The regulators did rapidly understand the functioning of the assisting system; the system was perfectly able to deal with the activity models and with their evolution according to the one of operators' practices.

Our approach is intermediary to several domains at different levels. First, considering SAGIM architecture, we propose an hybridizing of an analogical reasoning model (the Case-Based Reasoning) and of an more structured reasoning mode (based on operators' activities modeling via contextual graphs). This hybridizing is advocated by several works about reasoning modeling [Shrobe and Szolovits, 1993]. In fact each modeling introduce omissions with respects to the object modeled. Thus the combination of different models allow to have several point of view and thus to have a more global vision on the object. In our case the modeled object corresponds to incident management strategies. The consequences of omissions might be dramatic and it is safer to have several point of view. The modeling based on Case-Based Reasoning propose a fuzzy reasoning leading to the emergence of unexpected similarities between situations considered as very different. The solutions obtained by such a reasoning might seem ill-chosen but point specificities of the situations that might be of great importance. The modeling of operators' activities via contextual graphs support a highly structured reasoning. This structuring operates at two levels: first it structures the set of the activities (indicating which activities are implies which ones), and second, for each activity, this model structures the diagnosis and actions. The results obtained by this modeling are classical and no unexpected result can be produced. This reasoning allow thus to structure the knowledge involved in these activities and to explain the reasoning applied.

The contextual graphs model is itself intermediary to research fields studying rule-based reasoning (such as the ones involved in the first expert systems), the ones studying decision theory and the Context-Aware Systems. Rule-based reasoning propose the principles of the applied reasoning, linking the diagnosis results to the undertaken actions. Decision theory

propose a structuring of the diagnosis steps and allow a first structured representation of the diagnosis and the actions. However these diagnosis/action structures separate both components and ignore the consequences of the actions (and their results) over the evolution of the situation. The introduction of context show that diagnosis and actions are undissociable and that the reasoning is highly dynamical. Each undertaken action modifies the situation and thus influences the further diagnosis. Operators' reasoning is based on the situation and its evolution during he treatment. Context and its evolution are thus the core of operators' reasoning and of the model we propose. Moreover, Context Aware Systems propose a dynamical view on the environment of the user and adapt their behavior to the one of the user. Our context-based approach might be enhanced by these works which are a essential complement to it.

Finally our approach is intermediary between artificial intelligence and cognitive psychology. The activity modeling based on contextual graphs is highly linked to works realized by the cognitive psychology community (about schemes, tasks decomposition, the role of context in human reasoning...), by artificial intelligence researchers (Expert Systems, task/method paradigm, dynamical planning...) and to works already situated between both research fields (Frames, Scripts, MOPs...).

The work we realized on the modeling of context-situated reasoning, which lead to activity modeling through contextual graphs, were put into operation into SAGIM prototype. This Context-sensitive Intelligent Assisting System organizes the knowledge about incident management and deals with the subway regulators' activities. Two development main lines are now in operation. First a operational release of SAGIM is currently developed. This one respects the specifications given in the sixth chapter (sections 2 and 3) and take up the functioning principles of the prototype described in this thesis. The release enlarge also the application field by increasing the generality of the system. Second a new prototype is currently developed to adapt SAGIM to the Parisian region express rail network (RER in French). This prototype namely introduce a software interface with the existing incident databases for the RER. This portage is interesting twofold: first it extends the application field of SAGIM to the RER, which is a system quite different of the subway (this permit to test and validate the ideas developed in this thesis in other application conditions); second, incident databases, which are inspired from the RER one, are currently adapted to the subway, the portage of SAGIM prototype to the RER realizes thus a first step to the integration of SAGIM first release into the new working environment of subway regulators. Moreover this prototype will interact with a regulation agent of SART which will be developed on that occasion.

Besides the traditional statistical studies, the incident bases maintained by SAGIM allow the study of typical incidents and of the several practices applied for their management. The typical incident base might thus be the support for a case study realized by the executives managing the subway regulators. These studies might be eased by a new SART agent specialized in this task.

Another interesting extension of SART would be the development of an agent dedicated to support the procedures' enhancement. This agent would start from the structuring of operators' activity to propose new diagnosis/action plans. This agent would be suited for procedures editors and would guide them in a more complete integration of the context and its consequences into procedures. The cycle of procedures adaptation to real practice would thus be accelerated and facilitated by SART.

Finally, the application of the contextual graphs is not restricted to the modeling of regulators' activities. Other activities might be modeled through these graphs, namely all activities which are based on the duality between procedures and practices: incident management, supervision (check-lists), medicine... More generally contextual graphs might be applied when an interpretation or an adaptation of general rules is necessary to take into account the richness of the real application context.