



Université
de Toulouse

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Université Toulouse 1 Capitole (UT1 Capitole)

Présentée et soutenue par :

Hamdi Chaker

Le mardi 24 juillet 2012

Titre :

Une approche de gestion de contextes métiers pour l'accès à l'information

ED MITT : Image, Information, Hypermedia

Unité de recherche :

IRIT UMR 5505

Directeur(s) de Thèse :

Chantal Soulé-Dupuy - Professeur, Université Toulouse 1 Capitole

André Tricot - Professeur, Université Toulouse-le-Mirail

Rapporteurs :

Sylvie Calabretto - Professeur, INSA de Lyon

Ludovic Le Bigot - Professeur, Université de Poitiers

Autre(s) membre(s) du jury :

Thierry Nodenot - Professeur, IUT Bayonne, examinateur

Max Chevalier - Maître de conférences (HDR), Université de Toulouse III, co-encadrant

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Université Toulouse 1 Capitole (UT1 Capitole)

Titre :

**Une approche de gestion de contextes métiers pour
l'accès à l'information**

Auteur :

Hamdi Chaker

Directeur(s) de Thèse :

Chantal Soulé-Dupuy - Professeur, Université Toulouse 1 Capitole

André Tricot - Professeur, Université Toulouse-le-Mirail

Rapporteurs :

Sylvie Calabretto - Professeur, INSA de Lyon

Ludovic Le Bigot - Professeur, Université de Poitiers

Autres membres du jury :

Thierry Nodenot - Professeur, IUT Bayonne, examinateur

Max Chevalier - Maître de conférences (HDR), Université de Toulouse III, co-encadrant

Hamdi Chaker

Une approche de gestion de contextes métiers pour l'accès à l'information

Directeurs de thèse

Chantal Soulé-Dupuy, Professeur à l'Université Toulouse 1 Capitole
André Tricot, Professeur à l'Université Toulouse-le-Mirail

RESUME

L'accès à l'information dans un contexte métier (maintenance aéronautique), c'est-à-dire lorsque l'utilisateur doit retrouver et lire un document pour réaliser une tâche métier, est aujourd'hui devenu un réel défi dans la mesure où les moyens et dispositifs existants, en particulier les modèles et moteurs de recherche classiques, s'avèrent inadaptés par manque de pertinence.

Le travail que nous proposons se base sur la notion de contexte et plus précisément sur la gestion des informations de contexte afin de répondre aux besoins d'adaptation des systèmes d'accès à l'information et particulièrement dans des cadres métiers. L'objectif global est donc d'adosser une couche de contextualisation à ces systèmes, les rendant plus souples et adaptables.

La contribution de ce mémoire vise à proposer un gestionnaire de situations contextuelles générique et adapté à tout contexte métier. Le gestionnaire de situations contextuelles proposé prend en compte toutes informations contextuelles nécessaires au processus de contextualisation des systèmes d'accès à l'information métiers. Il est particulièrement efficace pour les domaines où l'enjeu de fiabilité est fort et où les tâches sont bien structurées. Pour cela, nous avons choisi la maintenance aéronautique comme domaine d'application parce qu'elle réunit toutes les conditions pour montrer l'efficacité de notre approche.

Nous avons étudié cette notion de contexte et nous proposons une définition générique du contexte et les concepts qui le composent. Cette définition n'exclut aucune dimension contextuelle jugée pertinente dans la littérature. Nous avons insisté sur la généralité de la définition afin que le gestionnaire de situations contextuelles puisse intégrer tout ce qui peut être considéré comme des informations contextuelles.

Nous exposons également dans ce travail une nouvelle définition de la situation comme étant une interprétation du contexte à un instant donné. Autrement dit, nous définissons la situation comme la photographie de l'activité à un instant t . Ainsi, le gestionnaire combine les informations issues du contexte pour créer des situations et ces dernières seront le point d'entrée des systèmes (maintenance aéronautique) qui se basent sur notre gestionnaire.

Pour valider les propositions présentées dans ce travail, nous avons implémenté un prototype. Ce dernier s'articule aux applications en quête de contextualisation (exemple : système d'accès à l'information dans la maintenance aéronautique). Le prototype permet de réaliser les engagements du gestionnaire de situations contextuelles envers les applications.

À mes parents,

« Le grand homme est comme l'aigle ; plus il s'élève, moins il est visible, et il est puni de sa grandeur par la solitude de l'âme. » [Stendhal]

Remerciements

Je tiens tout d'abord à remercier les membres extérieurs du jury. Ainsi, ma reconnaissance va à Madame Sylvie Calabretto, professeur à l'INSA de Lyon et à Monsieur Ludovic Le Bigot, professeur à l'université de Poitiers qui m'ont fait l'honneur d'être rapporteurs de ce mémoire. Je remercie également Monsieur Thierry Nodenot, professeur à l'IUT de Bayonne pour l'honneur qui m'a fait en acceptant d'être le président de mon jury de thèse.

Je tiens à remercier très sincèrement Messieurs Claude Chrisment et Gilles Zurfluh, responsables de l'équipe « Systèmes d'Informations Généralisées » (SIG), pour m'avoir accueilli au sein de leur équipe afin que je puisse mener au bout cette thèse.

Je tiens à remercier chaleureusement Madame Chantal Soule-Dupuy, Professeur à l'Université Toulouse I Capitole, pour avoir dirigé et encadré cette thèse. Je la remercie de la confiance qu'elle m'a faite, de ses critiques constructives ainsi que pour ses conseils. Je la remercie de m'avoir donné l'occasion de vivre cette expérience

Je tiens également à remercier vivement mon codirecteur de thèse, Monsieur André Tricot, Professeur à l'Université Toulouse-le-Mirail pour avoir accepté d'encadrer mes recherches. Je le remercie pour sa gentillesse, sa disponibilité et son expérience qui m'a beaucoup aidée dans ce travail.

Je tiens spécialement à remercier mon co-encadrant, Monsieur Max Chevalier, maître de conférences, HDR à l'université Toulouse 3. Sa disponibilité permanente et son sérieux m'ont beaucoup apporté. Il a su trouver les mots pour me faire avancer et m'a donné tous les moyens pour permettre à ce travail d'aboutir.

Je tiens à remercier du fond du cœur les membres de l'équipe SIG pour leur soutien. Je remercie en particulier tous les thésards passés qui ont partagé leur expérience avec moi : Karim, Ronan, Guillaume, Saad, Dana, Houssein. Merci à mes amis pour les pauses déjeuner passées ensemble et pour les moments de rire : Arlind, Fatma, Imen, Faten, Amjed, Cyril, Firas, Anas, Dana, Madalina et encore bien d'autres. Je les remercie d'autant plus pour leurs relations à la fois professionnelles et amicales.

Je remercie les membres de l'UFR d'informatique de l'Université Toulouse 1 Capitole pour leur collaboration dans les différentes unités d'enseignements où je suis intervenu.

J'adresse un remerciement à toutes les personnes qui m'ont entouré durant ces quelques dernières années. Je tiens à remercier tous mes amis en France et également tous ceux qui sont en Tunisie ou ailleurs.

J'ai une pensée bien particulière à Marion qui n'a cessé de m'encourager tout au long de ces trois dernières années.

Encore une fois, je n'oublie pas les sacrifices de mes parents qui m'ont permis d'être ici aujourd'hui. Un énorme merci à mon père et à ma mère pour leur indéfectible soutien.

Toulouse, le 24 juillet 2012

Hamdi Chaker

Introduction générale	14
Objectif.....	16
Contexte du travail et problématique	16
Contribution.....	19
Organisation de la thèse.....	21
Partie I : Etat de l'art	23
Chapitre I.1 : Représentation et gestion de l'information de contexte	25
I. Introduction	29
II. Pourquoi avons-nous besoin du contexte ?	30
II.1. Informatique contextuelle.....	30
II.2. Les objectifs de l'adaptation.....	31
II.3. Quelques utilisations du contexte.....	32
III. Définitions du contexte dans l'informatique ubiquitaire.....	34
IV. Représentation et modélisation du contexte.....	37
V. Les infrastructures de gestion de contexte	39
V.1. Qu'est-ce qu'un gestionnaire de contexte	39
V.2. Exemples de gestionnaires de contexte	42
VI. Synthèse des approches et modèles de contexte	45
VII. La situation contextuelle	47
VII.1. Besoin de la situation contextuelle	47
VII.2. Quelques définitions de la notion de situation contextuelle.....	48
VIII. Conclusion.....	50
Chapitre I.2 : L'accès contextuel à l'information	53
I. Introduction	57
II. Accès à l'information	58
II.1. Introduction	58
II.2. Recherche d'information	59
II.2.1. Phases de la RI.....	60
II.2.2. Modèles de RI.....	61

Modèle booléen	61
Modèle vectoriel	61
Modèle probabiliste	62
II.3. Filtrage / recommandation d'information.....	62
II.3.1. Filtrage basé sur le contenu	63
II.3.2. Filtrage collaboratif.....	64
II.3.3. Filtrage hybride	65
III. Vers l'accès contextuel à l'information.....	65
III.1. Problématiques de l'accès à l'information hors contexte.....	66
III.2. Les systèmes de filtrage (recommandation) contextuels	67
III.3. La recherche d'information contextuelle.....	68
III.4. Synthèse sur l'accès à l'information en contexte	69
IV. Définition et application du contexte dans la RI	69
IV.1. Qu'est-ce que le contexte en RI.....	70
IV.2. Les vues du contexte en RI.....	70
IV.3. Quelques applications du contexte en RI	75
IV.3.1. Niveau environnemental de l'information	75
IV.3.2. Niveau humain de RI	76
IV.3.3. Niveau interactionnel de la RI	77
IV.3.4. Niveau de la requête.....	79
IV.4. Synthèse.....	81
V. Les dimensions pertinentes du contexte dans l'accès à l'information	82
V.1. La tâche.....	82
V.1.1. La tâche comme motivation de la RI	83
V.1.2. Tâches informationnelles	83
Définition.....	83
Lien entre tâche informationnelle et tâche de travail	84
V.1.3. Principales caractéristiques des tâches.....	86
V.1.4. Modélisation des tâches	87
Introduction.....	87
Quelques formalismes de modélisation des tâches	89
Différences entre ces formalismes	94
V.2. L'utilisateur	96
V.2.1. Profil usager.....	96
V.2.2. Au-delà du profil usager classique.....	97
V.3. Environnement.....	98

VI.	Conclusion.....	99
	Conclusion état de l’art.....	101
	Partie II : Contribution.....	107
	Chapitre II.1 : Le gestionnaire de situations contextuelles métier.....	109
I.	Introduction	115
II.	Pourquoi un gestionnaire de situations contextuelles ?.....	115
III.	Le contexte	116
III.1.	La définition du contexte	117
III.2.	Les concepts du contexte	118
	La dimension contextuelle	119
	L’élément contextuel	119
	a) Nom	119
	b) Valeur.....	119
	c) Constance.....	119
	d) Fiabilité.....	120
IV.	Interprétation du contexte.....	120
IV.1.	Exemples d’interprétation du contexte	120
	a) Exemple de contexte initial.....	121
	b) Exemple d’instances de dimensions contextuelles.....	121
	c) Exemple 1 : conservation des dimensions initiales	122
	d) Exemple 2 : modification des modèles initiaux des dimensions contextuelles	122
	e) Exemple 3 : modification des valeurs des éléments contextuels initiaux.....	123
IV.2.	Rôles des interprétations du contexte	124
	IV.2.1. Validité de l’interprétation du contexte.....	125
	IV.2.2. Prise en compte de l’impact des éléments contextuels les uns sur les autres	125
	IV.2.3. Prise en compte des informations contextuelles manquantes	126
V.	La Situation	128
V.1.	Objectifs.....	128
V.2.	Définition de la situation	128
V.3.	Les concepts de la situation	129
V.4.	Exemple de situation	130
VI.	Gestionnaire de situations contextuelles	130
VI.1.	Architecture du gestionnaire de situations contextuelles	131

VI.2.	Le gestionnaire de données contextuelles	132
VI.2.1.	Présentation.....	132
VI.2.2.	La base des règles	133
a)	Les trois types de règles	133
b)	La définition des règles	134
VI.3.	Le processus de Mise En Situation.....	135
VI.4.	Le processus d'extraction de règles.....	137
VI.5.	Les engagements du GSC envers les applications.....	138
VI.5.1.	Vérification des situations valides	138
VI.5.2.	Comblir le manque d'information dans le contexte	139
VI.5.3.	Proactivité/anticipation	140
VI.5.4.	Fournir les éléments contextuels.....	140
VI.6.	Synthèse.....	140
VII.	Démarche de création de la situation	141
VII.1.	Les informations contextuelles nécessaires à la création des situations.....	142
VII.1.1.	Les informations contextuelles transitant par le GDC	142
VII.1.2.	Les informations contextuelles transitant par le MES	144
VII.2.	Fonctionnement du processus MES	146
VII.3.	Algorithme général du MES.....	146
VII.4.	Adaptation des dimensions contextuelles.....	147
VII.4.1.	Principe	147
VII.4.2.	Exemple illustratif.....	148
VII.5.	Stabilisation du contexte.....	150
VII.5.1.	Principe	150
VII.5.2.	L'algorithme de stabilisation du contexte	150
VII.5.3.	Illustration de l'algorithme de la stabilisation.....	152
VIII.	La démarche d'extraction de règles.....	154
VIII.1.	La fouille de règles d'association	155
VIII.1.1.	Les concepts de la fouille de règles d'association.....	155
VIII.1.2.	Exemple	157
VIII.2.	Fonctionnement du processus d'extraction des règles	158
VIII.2.1.	Motivation.....	158
VIII.2.2.	Les étapes de l'extraction des règles.....	160
VIII.2.3.	L'algorithme d'extraction de règles	161
VIII.2.4.	Exemple d'extraction de règles dans notre approche.....	162
IX.	Conclusion.....	164

Chapitre II.2 : Implantation et expérimentation - application à l'accès à l'information dans la maintenance aéronautique	167
I. Introduction	171
II. Spécificités du GSC pour l'accès à l'information	172
II.1. Objectif	172
II.2. Différentes composantes du GSC pour les SAI métier	173
II.3. Exemple de mise en situation pour un SAI métier	174
II.4. Exemple d'extraction de règles pertinentes pour un SAI métier	176
III. Architecture globale du prototype implanté	177
III.1. Serveur de situations	179
III.2. Serveur de données	179
III.3. Gestionnaire de données contextuelles	180
III.4. Le processus MES	181
III.5. Processus de fouille de règles	181
IV. Expérimentation basée sur les données de la maintenance aéronautique ..	182
IV.1. Etude du domaine de la maintenance	182
IV.2. Modélisation des trois dimensions pour la maintenance	183
IV.2.1. Dimension environnement	183
IV.2.2. Dimensions usager	184
IV.2.3. Dimension tâche	184
IV.3. Simulation de mises en situations	185
a) Transformation des tâches	185
b) Modélisation des règles	187
IV.4. Fouille de règles basée sur des situations réelles	188
IV.4.1. Méthodologie	188
IV.4.2. Résultat de la fouille de règles	189
V. Conclusion	190
Conclusion générale	193
Synthèse de nos propositions	195
Limites et perspectives de recherche	197
Bibliographie générale	201

Annexe	217
I.	Exemple de contenu d'une procédure de maintenance aéronautique	219
II.	Décomposition des tâches en deux groupes en se basant sur leur nature.....	220
III.	Tableaux des différentes valeurs des éléments composant les vingt situations récoltées avec l'aide des psycho-cogniticien	221

Introduction générale

Objectif

Le travail que nous proposons se base sur la notion de contexte et plus précisément sur la gestion des informations de contexte afin de répondre aux besoins d'adaptation des systèmes d'accès à l'information et particulièrement dans des cadres métiers. En effet, nous devons leur permettre d'utiliser ce type d'information pour la personnalisation des traitements et des données. L'objectif global est donc d'adosser une couche de contextualisation à ces systèmes, les rendant plus souples et adaptables.

Contexte du travail et problématique

La maintenance aéronautique consiste à réaliser des tâches ou des combinaisons de tâches de remplacement, modification et correction d'anomalie d'un aéronef ou d'un élément d'aéronef. La maintenance est un des moyens de maintenir les performances de l'avion et prévenir les incidents et les accidents aériens, parfois catastrophiques. L'objectif de la maintenance est aussi de minimiser la durée d'immobilisation de l'avion en cas de dysfonctionnement en respectant les conditions de **conformité** et d'**efficacité** de la réparation dans les **meilleurs délais**.

Dans ce domaine, la place de la maintenance est plus grande que dans d'autres, car l'avion coûte extrêmement cher, mais aussi parce que sa sûreté de fonctionnement est un enjeu principal, de par les risques de pertes humaines en cas d'accident. De ce fait, c'est une activité très réglementée, qui doit obéir à des normes et réglementations strictes définies par des organismes internationaux. Les procédures de mise en œuvre des tâches de maintenance aéronautique sont inscrites dans des documents que l'opérateur doit suivre et qui doivent garantir à l'opérateur qui l'utilise l'exécution d'une tâche appropriée, efficace et légale.

La documentation technique aéronautique a pour but la réalisation d'un ensemble d'actions structurées de manière procédurale pour guider l'opérateur dans la réalisation d'une tâche de maintenance aéronautique. En effet, le document technique aéronautique propose à ce dernier une liste prescrite d'instructions organisées définissant le déroulement détaillé du travail. Ainsi, la documentation technique aéronautique a une fonction de support à la tâche afin d'aider les utilisateurs à l'accomplir, en suivant des étapes obligatoires pour assurer la sécurité, la charge de travail et les conditions de performance.

L'utilisation de la documentation technique aéronautique a également une fonction légale. En effet, elle garantit aux supérieurs hiérarchiques et à l'autorité concernée que l'opérateur a suivi fidèlement les prescriptions du déroulement de la tâche de maintenance réalisée. La documentation technique aéronautique est donc gage de sécurité et de sûreté des avions par les procédures de maintenance qu'elle fournit. Pour cela et afin de s'assurer du déroulement conforme de la tâche de maintenance, la réglementation oblige l'opérateur à certifier (par signature) qu'il a bien utilisé le document et qu'il a respecté les étapes de la procédure.

Par ailleurs, la documentation technique aéronautique est décomposée en trois familles pour satisfaire trois besoins essentiels des compagnies aériennes : la formation, l'exploitation des appareils et l'entretien de la flotte. La documentation est livrée avec chaque avion et se focalise uniquement sur l'état technique de cet avion et de ses composantes. Sachant que les avions diffèrent d'une compagnie aérienne à une autre (cela peut être le cas dans une même compagnie), la documentation technique est très abondante et particulièrement précise. Ainsi, toute l'activité de maintenance est fondée sur un accès précis à l'information : chaque tâche, parmi des centaines de milliers, correspond à un ensemble strictement défini de documents non seulement nécessaires mais obligatoires. Ainsi, l'opérateur doit trouver le ou les « bons » documents correspondant à sa tâche et aux caractéristiques spécifiques au modèle de l'avion. Pour cela, il se base sur un système d'accès à l'information dédié au métier de la maintenance. Sans ce dernier, l'opérateur ne pourrait pas accéder au document dont il a besoin.

En somme, la maintenance aéronautique est un domaine où les enjeux de sûreté, de légalité, d'efficacité et d'efficience sont importants et connus de tous. C'est aussi un domaine bien défini et structuré : toutes les tâches sont connues et définies ; leur réalisation implique l'utilisation d'un ou plusieurs documents connus et définis. Pourtant, la maintenance aéronautique est un domaine où les erreurs humaines existent, souvent liées au non-respect d'une procédure ou à la non-utilisation d'un document, avec des conséquences dramatiques. Nous allons maintenant nous focaliser sur ces problèmes de défaillances humaines et informationnelles.

L'association de professionnels pour l'amélioration de la sécurité de l'aviation commerciale a conclu dans une étude (CAA, 2002) que la documentation de maintenance est la première source d'incidents dans la maintenance. En effet, des études ont montré que 54 % des erreurs dans la maintenance sont liées à l'exécution de la procédure par les opérateurs.

Ces erreurs sont très graves et peuvent aller de l'immobilisation de l'appareil, à l'accident et la perte de vies humaines. Les erreurs de maintenance sont la cause de 12 % des accidents d'avion (Hobbs, 2000). Le rôle de la documentation en général, et en particulier le rôle du processus d'accès à l'information est très important pour éviter de telles catastrophes. Ainsi, la même enquête a montré que la moitié des erreurs humaines dans la maintenance sont liées à la documentation technique. 28 % des erreurs des opérateurs sont dues à la non-utilisation de la procédure (**manque de contrôle avant de débiter la tâche**). Plus encore, 24 % des erreurs sont directement liées au processus de recherche d'information : dispersion des utilisateurs dans le processus informationnel ou bien non-utilisation du document approprié (**manque d'assistance et d'adaptation spécifique à l'opérateur**). Ces deux derniers points montrent qu'en maintenance aéronautique, les opérateurs censés être contraints de suivre leurs procédures, ne le sont pas effectivement. Une première hypothèse consisterait à envisager la malveillance, la négligence ou l'inattention du technicien qui fait une erreur, l'organisation du travail ou toute autre source « humaine » (cf. (Reason & Hobbs, 2003) pour une synthèse). Une seconde hypothèse, qui est à la source du présent travail, consiste à envisager cette défaillance en termes d'accès à l'information.

Plus précisément, notre hypothèse de travail est que les systèmes de maintenance ne prennent pas en compte les informations contextuelles liées à aux procédures : l'environnement de la tâche (exemple : les ressources mises à la disposition pour le déroulement de la procédure), les caractéristiques de la tâche et enfin les caractéristiques des opérateurs ne sont pas considérées. Les systèmes actuels n'ont pas ces informations et ne peuvent donc ni **contrôler** le processus de travail pour assister les opérateurs, ni **adapter** les processus informationnels pour éviter d'éventuelles erreurs.

Prenons un exemple afin d'aborder la problématique liée à l'absence de contextualisation des systèmes d'accès à l'information dans le domaine de la maintenance. Une étude (Reason & Hobbs, 2006) s'est intéressée aux facteurs provoquant les erreurs de maintenance ; elle a montré qu'une procédure jugée « routinière » combinée à un opérateur « expérimenté » peut paradoxalement être un des facteurs les plus importants d'erreur. Évidemment, les systèmes classiques sont inadaptés dans ce cas et ne vont pas aider l'opérateur à éviter l'erreur. La cause soulevée est celle des systèmes actuels de maintenance qui n'ont pas cette photographie de ce qui se passe réellement durant une activité. En effet,

pour cet exemple et afin d'éviter ce type d'erreur, le système doit contrôler le déroulement des étapes de la tâche et l'assistance de *l'expert* effectuant une tâche *routinière* sera renforcée.

Le troisième et dernier niveau se focalise sur la gestion des activités passées et plus précisément sur **l'absence totale de traçabilité dans les systèmes actuels de maintenance aéronautique**. Ainsi, les experts du domaine n'ont pas de vision systématique des activités passées, ni sur ce qui s'est vraiment produit pendant le déroulement des procédures. Par conséquent, ils ne peuvent pas connaître les sources exactes des erreurs de maintenance survenues dans ces activités. Par ailleurs, les systèmes actuels ne peuvent ni améliorer leurs processus en basant sur les expériences passées des utilisateurs, ni éviter à nouveau les erreurs rencontrées parce qu'ils ne prennent en compte aucun feedback.

En résumé, les systèmes d'accès à l'information utilisés dans la maintenance aéronautique ne sont pas adaptés à la complexité de ce domaine. Nous catégorisons leurs faiblesses en trois niveaux suivant le déroulement des étapes de l'activité. En effet, le premier niveau correspond à une absence du **contrôle** nécessaire pour débiter la tâche de maintenance. Le deuxième problème réside ensuite dans le manque d'**adaptation** des processus informationnels et dans le manque d'**assistance** des opérateurs pendant l'exécution de la tâche. Enfin, les systèmes n'ont aucune traçabilité et aucun **feedback** sur ce qui s'est vraiment déroulé dans les activités passées. En conclusion, nous pouvons assurer que les systèmes actuels ne peuvent pas être efficaces, ni éviter des erreurs de maintenance parce qu'ils ne gèrent pas toutes les informations associées à l'activité : c'est ce que nous appelons les « **informations issues du contexte de l'activité** ».

Contribution

La contribution de ce mémoire vise à proposer un gestionnaire de situations contextuelles générique et adapté à tout contexte métier. Ainsi, notre approche permet de gérer les problèmes liés à la prise en compte des informations contextuelles pour améliorer les systèmes en quête de contextualisation. Le gestionnaire de situations contextuelles proposé prend en compte toutes informations contextuelles nécessaires au processus de contextualisation des systèmes métiers. Il est particulièrement efficace pour les domaines où l'enjeu de fiabilité est fort et où les tâches sont bien structurées. Pour cela, nous avons choisi la maintenance aéronautique comme domaine d'application parce qu'elle réunit toutes les conditions pour montrer l'efficacité de notre approche.

Le gestionnaire de situations contextuelles permet de traiter les données issues du contexte des activités de maintenance aéronautique. Ce gestionnaire résout les problèmes présentés plus haut. Il permet le contrôle de l'activité de maintenance en contrôlant la conjonction de toutes les informations contextuelles pour une activité donnée. Il propose ensuite au système de maintenance aéronautique la photographie réelle des activités. Ces dernières sont utilisées soit pour l'adaptation des procédures de maintenance ainsi que pour les processus informationnels qui y sont liés, soit pour proposer au système un feedback sur les activités passées des opérateurs.

Notre contribution peut être présentée en trois volets.

Le gestionnaire de situations contextuelles que nous proposons se base sur les informations de contexte. Pour cela, nous avons étudié cette notion et nous proposons une définition du contexte et les concepts qui le composent. Cela n'était pas une tâche facile en raison de la multitude de définitions de cette notion dans la littérature. Nous avons ainsi proposé une définition générique du contexte qui ne dépend pas d'un domaine spécifique et qui n'exclut aucune dimension contextuelle jugée pertinente dans la littérature. Nous avons insisté sur la généricité de la définition afin que le gestionnaire de situations contextuelles puisse intégrer tout ce qui peut être considéré comme des informations contextuelles.

Dans le deuxième volet, nous définissons la notion de situation en plus des concepts qui la composent. Comme pour la notion de contexte, nous avons été confrontés à une multitude de travaux qui définissent et utilisent cette notion. Cependant, nous exposons dans ce travail une nouvelle définition de la situation comme étant une interprétation du contexte à un instant donné. Autrement dit, nous définissons la situation comme la photographie de l'activité à un instant t . Ainsi, le gestionnaire combine les informations issues du contexte pour créer des situations et ces dernières seront le point d'entrée des systèmes (maintenance aéronautique) qui se basent sur notre gestionnaire.

Dans le dernier volet de notre contribution, nous présentons le gestionnaire de situations contextuelles. Ce dernier propose aux systèmes les données contextuelles nécessaires pour **adapter** le processus informationnel de ses activités. Le gestionnaire de situations contextuelles permet aussi de **contrôler** les données contextualisées avant la création de situation et ainsi, dans le cas des systèmes de maintenance aéronautique, appuyer l'aspect légal du respect de la procédure de maintenance.

De plus, pour donner une photographie la plus fidèle possible à la réalité, nous proposons une approche « originale » de création de situation dans laquelle le gestionnaire doit prendre en considération l'impact de tous les éléments contextuels en entrée les uns par rapport aux autres.

Enfin, le gestionnaire de situations contextuelles permet aussi l'extraction de nouvelles connaissances à partir des situations passées et permet ainsi de proposer un feedback aux systèmes. Cela leur permet d'utiliser les nouvelles informations issues des activités passées pour améliorer les procédures courantes ou bien futures. Ainsi, pour les systèmes de maintenance par exemple, la traçabilité que leur propose notre gestionnaire à travers les situations passées permet aux experts du domaine de mettre le doigt sur les causes réelles des erreurs de maintenance commises par les opérateurs.

Organisation de la thèse

Le présent mémoire est organisé en trois parties.

La **première partie** est constituée de deux chapitres consacrés à l'état de l'art. Elle traite des contextes de notre travail que ce soit dans le domaine de la gestion du contexte ou celui de la contextualisation de l'accès à l'information.

Le premier chapitre s'intéresse à la notion de contexte. Ce dernier a reçu un intérêt croissant et partagé dans beaucoup de domaines de l'informatique. En effet, le contexte est depuis quelques années une notion primordiale qu'il faut intégrer dans les approches classiques, quel que soit le domaine. Pour cela, nous exposons un panorama des définitions de la notion de contexte issues de la littérature ainsi que leurs utilisations. Nous nous focalisons spécialement sur les approches de la gestion de cette notion dans le domaine de l'informatique ubiquitaire dans lequel elle est particulièrement développée. Nous introduisons finalement la notion de situation comme amplement liée à celle de contexte. Nous en présentons alors quelques définitions issues de la littérature.

Le deuxième chapitre s'intéresse également à la notion de contexte, mais dans un domaine particulier celui de l'accès à l'information. Nous exposons dans un premier temps les notions et concepts du filtrage et de la recherche d'information qui représentent les deux approches principales de l'accès à l'information. Ensuite, nous montrons le besoin du contexte dans ce domaine et illustrons ainsi l'émergence de l'accès contextuel à l'information.

De ce fait, nous exposons les définitions du contexte les plus connues dans le domaine de la recherche d'information ainsi que les approches qui les exploitent. À la fin du chapitre, nous nous focalisons sur les travaux qui se sont intéressés aux trois dimensions jugées les plus importantes du contexte dans le cadre de l'accès contextuel à l'information.

La **deuxième partie** est constituée de deux chapitres consacrés à notre contribution.

Le premier chapitre vise au préalable à donner notre vision du contexte et de la notion de situation. En effet, nous proposons une nouvelle définition du contexte ainsi que des concepts associés. Cette définition est intuitive et générique car elle prend en considération tout ce qui peut être considéré comme des informations contextuelles ; et ce dans le but principal de proposer un gestionnaire de contexte se basant sur notre vision du contexte et permettant à tout système s'y reposant d'avoir une photographie réaliste du contexte à un instant donné. Le gestionnaire de situations contextuelles a un rôle primordial pour garantir la pertinence et la précision des informations contextuelles provenant des différentes sources du contexte.

Le deuxième chapitre présente un cadre destiné à la validation des approches proposées. En effet, dans ce chapitre nous sortons de la généralité du gestionnaire de situations contextuelles afin de le transposer dans le domaine d'application métier qui nous intéresse, celui de la maintenance aéronautique. Ainsi, nous détaillons l'architecture du prototype réalisé et son fonctionnement. Enfin, nous donnons des résultats expérimentaux montrant l'intérêt du gestionnaire de données contextuelles dans la contextualisation de la maintenance aéronautique et dans l'amélioration des processus informationnels indispensables à l'accomplissement de ces activités.

Enfin, nous clôturons ce manuscrit par une conclusion et des perspectives de recherche ouvertes par notre contribution.

Partie I : **Etat de l'art**

Chapitre I.1 : Représentation et gestion de l'information de contexte

I.	Introduction.....	29
II.	Pourquoi avons-nous besoin du contexte ?.....	30
	II.1. Informatique contextuelle.....	30
	II.2. Les objectifs de l'adaptation	31
	II.3. Quelques utilisations du contexte.....	32
III.	Définitions du contexte dans l'informatique ubiquitaire	34
IV.	Représentation et modélisation du contexte	37
V.	Les infrastructures de gestion de contexte	39
	V.1. Qu'est-ce qu'un gestionnaire de contexte	39
	V.2. Exemples de gestionnaires de contexte	42
VI.	Synthèse des approches et modèles de contexte	45
VII.	La situation contextuelle	47
	VII.1. Besoin de la situation contextuelle.....	47
	VII.2. Quelques définitions de la notion de situation contextuelle.....	48
VIII.	Conclusion.....	50

I. Introduction

Le travail que nous proposons dans ce manuscrit se base sur la notion de contexte et plus précisément sur la gestion des informations de contexte afin de répondre aux besoins d'adaptation des applications. En effet, nous devons leur permettre d'utiliser ce type d'information pour la personnalisation des traitements et des données. L'objectif global est donc d'adosser une couche de contextualisation aux systèmes, les rendant plus souples et adaptables. Cependant, en plus de l'hétérogénéité de ces applications, nous devons aussi faire face à la multitude et l'ouverture des sources de contexte. Par conséquent, généraliser la gestion de ces informations de contexte n'est guère une tâche facile.

Ainsi, un intérêt croissant pour la notion de contexte est partagé dans presque tous les domaines de l'informatique. Le contexte a été défini et utilisé dans plusieurs domaines de recherche depuis des années, que ce soit en psychologie cognitive, en linguistique ou récemment en informatique. Les premières utilisations du contexte dans les sciences informatiques ont été réalisées dans le domaine de l'intelligence artificielle avec les systèmes logiques du premier ordre (Weyhrauch, 1979). Nous notons qu'en intelligence artificielle, la notion de contexte est principalement présente dans deux axes de recherches : la représentation de connaissances et la logique.

Malgré cela, le contexte est longtemps resté un « problème » mal défini dans la communauté informatique jusqu'à la fin des années quatre-vingt-dix. Depuis, une quantité importante de travaux utilisant le contexte vise avant tout à définir et comprendre cette notion afin de mieux la gérer. En effet, il existe un grand nombre d'approches permettant la gestion du contexte, mais le problème reste le même : les définitions proposées du contexte sont spécifiques aux domaines d'application et ne considèrent pas la globalité des dimensions contextuelles disponibles. Ainsi, Bazire et Brézillon (Bazire & Brézillon, 2005) ont présenté et examiné 150 définitions différentes du contexte dans des domaines variés. Ils ont souligné la difficulté de trouver une définition pertinente et satisfaisante du contexte dans toutes les disciplines.

Nous observons aujourd'hui une évolution très importante dans les travaux se basant sur la notion de contexte. En effet, les travaux antérieurs sur les systèmes sensibles au contexte se sont concentrés sur une modélisation de bas niveau du contexte. Récemment, l'intérêt porte sur la façon d'abstraire, de représenter et d'identifier des situations d'après un

contexte brut constitué d'informations bas niveau provenant directement de capteurs (Ye, Dobson, & McKeever, 2011). Ainsi, plusieurs approches tentent de définir cette notion de situation et le moyen de générer des situations à partir de contextes initiaux.

Nous consacrons ce premier chapitre de l'état de l'art à la notion de **contexte**. Pour cela, nous allons dans un premier temps donner des définitions du contexte issues de la littérature. Nous allons surtout nous intéresser à cette notion dans le domaine de l'informatique ubiquitaire dans lequel elle est très développée. Dans un deuxième temps, nous nous focalisons sur les méthodes les plus pertinentes de modélisation et de représentation du contexte. En plus de cela, nous présentons quelques infrastructures permettant de gérer les informations contextuelles. Finalement, et après avoir présenté la notion de contexte et des approches de gestion contextes, nous nous focalisons dans ce travail sur la notion de **situation** qui est une représentation à haut niveau d'un contexte exploitable par un système dans un but d'adaptation.

II. Pourquoi avons-nous besoin du contexte ?

Le terme « contexte » est couramment utilisé pour décrire un environnement, une situation, un état, une tâche, etc. D'après le dictionnaire de la langue française¹, « contexte » est un nom masculin (latin *contextus*, ce qui veut dire assemblage) (1) : ensemble des conditions naturelles, sociales, culturelles dans lesquelles se situe un énoncé, un discours (2) : ensemble des circonstances dans lesquelles se produit un événement, se situe une action (par exemple, replacer un fait dans son contexte historique...).

Avant de présenter les définitions du contexte dans son domaine phare qui pour nous est celui de l'informatique ubiquitaire, nous allons nous intéresser à la naissance de l'informatique contextuelle. Nous allons donc brièvement définir, dans ce qui suit, le rôle de la contextualisation pour les applications dites sensibles au contexte. Cela pour nous familiariser avec les définitions de cette notion de contexte.

II.1. Informatique contextuelle

Le terme « *context awareness* » a été introduit par Schilit et al. en informatique ubiquitaire (Schilit, Adams, & Want, 1994). Ce qui veut dire que les systèmes informatiques

¹ <http://www.larousse.fr/dictionnaires/francais/contexte>

peuvent sentir et interagir avec l'environnement externe. « *Context-aware computing is the ability of a mobile user's applications to discover and react to changes in the environment they are situated in* ». Dey (Dey, 2001) a aussi défini les systèmes sensibles au contexte comme « *context aware system* » : un système qui utilise le contexte pour fournir une information pertinente et/ou des services à l'utilisateur, où la pertinence dépend de la tâche de l'utilisateur. Dey différencie le contexte de la situation. Celle-ci est définie comme une description des états des entités pertinentes (Dey, 2001). Pour l'auteur, la description de la situation nécessite moins d'effort que la définition des composantes du contexte. Dans d'autres travaux, nous avons constaté que des chercheurs définissent l'information contextuelle, « *contextual information* », comme étant des données pertinentes extraites en tenant compte du contexte (Anagnostopoulos, Tsounis, & Hadjiefthymiades, 2007).

Enfin, nous pouvons dire qu'à la différence des applications classiques, celles qui sont sensibles au contexte s'adaptent en se basant sur les informations contextuelles fournies par un gestionnaire de contexte. **Le contexte est donc une source d'adaptation pour les applications.**

II.2. Les objectifs de l'adaptation

Quelques travaux ont essayé de définir un cadre à l'adaptation des applications sensibles au contexte. Schmidt (Schmidt, 2000) par exemple décrit trois objectifs d'adaptation en utilisant le contexte. Le premier concerne *l'adaptation des modalités d'entrées et de sortie* en utilisant le contexte. Le deuxième objectif est de faire en sorte que *les applications utilisent implicitement les informations contenues dans le contexte*. Le troisième et dernier objectif présenté par Schmidt est *la détermination des moments favorables et des modes pertinents d'interaction de l'utilisateur*.

D'un autre côté, pour Dey et ses collaborateurs (Dey, Abowd, & Salber, 2001) l'objectif principal de la contextualisation des systèmes est de prédire ce que fait ou tente de faire l'utilisateur à un moment donné afin de l'assister dans son activité en lui fournissant les services adéquats. Ils spécifient trois objectifs pour la gestion et l'utilisation de l'information contextuelle. Le premier objectif concerne *l'exécution de services*. Autrement dit, les applications sensibles au contexte peuvent lancer automatiquement des services pour assister les utilisateurs en utilisant les informations contextuelles. Le deuxième objectif concerne *la présentation de l'information*. En effet, les applications sensibles au contexte adaptent la

présentation de l'information aux utilisateurs en fonction des informations contextuelles en leur possession. Plus encore, ces applications peuvent utiliser le contexte pour proposer aux utilisateurs les actions les plus pertinentes selon leurs conditions spécifiques d'utilisation. Enfin, le troisième objectif d'adaptation utilisant le contexte touche *le stockage des données*. Les applications peuvent ajouter en plus de leurs données générées d'autres informations issues du contexte. Un exemple très courant pour comprendre le troisième objectif est celui des nouvelles générations d'appareils photo qui sont sensibles au contexte. En effet, ils stockent en plus des photos prises par l'utilisateur les coordonnées du lieu géographiques. Ainsi, les photos sont automatiquement géo référencées et peuvent être classées plus facilement.

II.3. Quelques utilisations du contexte

Plusieurs domaines informatiques ont utilisé la notion de contexte pour différents objectifs. Nous allons dans ce qui suit présenter quelques utilisations du contexte pour montrer son apport aux systèmes des différents domaines. En effet, les exemples d'utilisation du contexte que nous allons présenter ne sont pas exhaustifs mais nous permettent d'introduire la notion de contexte et plus précisément la portée de son utilisation selon l'approche de Dey (Dey et al., 2001) présentée dans la section précédente (II.2), i.e. pour assister l'utilisateur.

Dans le domaine de la modélisation des utilisateurs, White et ses collègues (White, Bailey, & Chen, 2009) se sont concentrés sur un type d'information de contexte qui est extrait des données des utilisateurs (i.e. log de recherches Web). En utilisant cette information contextuelle spécifique, ils construisent des profils des utilisateurs qui serviront a posteriori à prédire les intérêts des utilisateurs.

Le contexte est utilisé aussi pour rendre les maisons « intelligentes ». Par exemple, Vildjiounaite et ses collaborateurs (Vildjiounaite, Kocsis, Kyllönen, & Kladis, 2009) ont présenté une architecture pour l'acquisition de contextes et de préférences qui servira de base à un ensemble de dispositifs permettant de donner cet aspect d'intelligence aux maisons. Des exemples d'applications concernent la domotique, le shopping, l'enregistrement proactif de films et d'actualités, etc. (Vildjiounaite et al., 2009). Dans le même domaine, nous pouvons aussi présenter le travail de Kavitha (Kavitha, 2009). L'auteur a montré, en utilisant certains scénarios, que le contexte apporte beaucoup pour les applications de la domotique. Il a

également montré dans ce travail que le contexte est utilisé pour personnaliser la réaction des systèmes de commande de la maison. La personnalisation concerne principalement le niveau de luminosité, le volume du téléviseur, la climatisation, etc.

Le contexte est aussi utilisé dans le domaine des services sensibles au contexte, il contient toutes les informations qui peuvent être utilisées par un service Web pour ajuster son exécution et sa sortie (Sheng & Benatallah, 2005). Par ailleurs, dans le domaine de l'accès et de la recherche d'information, l'enjeu de l'intégration du contexte dans un système d'accès à l'information réside dans le simple fait qu'un système ne peut pas afficher le même résultat pour deux usagers travaillant dans deux contextes différents, uniquement parce qu'ils ont exprimé le même besoin. En outre, selon le contexte, le processus d'utilisation de l'information de ces deux usagers sera différent (J. D. Johnson, 2003). Le contexte va donc servir à rendre l'accès à l'information sensible au contexte. Nous allons nous focaliser sur ce domaine dans le deuxième chapitre de l'état de l'art.

Enfin, le contexte et les informations de contexte sont utilisés dans le domaine qui nous intéresse dans ce chapitre, c'est-à-dire l'intelligence ambiante, et notamment à travers son utilisation pour adapter l'interaction entre l'homme et la machine. Par exemple, un des systèmes sensibles au contexte dans ce domaine est le guide touristique intelligent. En effet, ce système est une des applications dans laquelle la notion de contexte a été largement étudiée. Dans le travail présenté dans (Cheverst, Davies, Mitchell, Friday, & Efstratiou, 2000), les auteurs présentent un système qui aide les touristes à visiter une ville en utilisant les informations contextuelles. Le guide proposé répond aux besoins informationnels des visiteurs en utilisant des hypermédias adaptatifs. Dans ce but, ce système exploite des informations telles que la localisation des visiteurs, le temps, l'intérêt des visiteurs, etc. D'autres travaux s'inscrivant dans la même démarche sont présentés dans (Krüger, Baus, Heckmann, Kruppa, & Wasinger, 2007). Les auteurs abordent les questions de modélisation du contexte et les situations avant de donner un aperçu sur les systèmes existants provenant de trois classes importantes de guides mobiles : les guides des musées, les systèmes de navigation intelligents et les assistants commerciaux.

Dans ce qui suit, nous allons présenter les définitions du contexte dans le domaine de l'informatique ubiquitaire qui s'est le plus intéressé à la notion de contexte ces dernières années.

III. Définitions du contexte dans l'informatique ubiquitaire

Les contextes ont été soumis à de multiples définitions, certains d'entre eux se confondent souvent avec le concept de profil. Par exemple, une des définitions les plus larges et également avant-gardiste est celle de Schilit et ses collaborateurs : « *Le Contexte englobe d'avantage que simplement l'emplacement de l'utilisateur, parce que d'autres choses/points d'intérêt sont aussi mobiles et changeant(e)s : l'ensemble des gens et des objets qui se trouvent à proximité, aussi bien que les changements de ces objets dans le temps. Le contexte inclut l'éclairage, le niveau sonore, la connectivité du réseau, les coûts de communication, la largeur de bande de communication et même la situation sociale (par exemple, si vous êtes avec votre directeur ou avec un collaborateur)* » (Schilit et al., 1994). L'emplacement de l'utilisation est la partie la plus importante dans la compréhension du contexte que Schilit et ses collaborateurs ont développé. Pour eux, les aspects importants du contexte sont : où vous êtes, et avec qui, et quelles sont les ressources accessibles, donc disponibles. Pascoe définit lui le contexte comme *un sous-ensemble d'états physiques et conceptuels ayant un intérêt pour une entité particulière* (Pascoe, 1998).

D'après ces définitions, le contexte reste très spécifique aux conditions particulières d'utilisation. C'est-à-dire, le contexte est tout ce qui touche une situation appropriée à une application et son ensemble d'utilisateurs. Les deux définitions de Schilit et Pascoe sont plus proches dans l'esprit de la définition opérationnelle du contexte de Dey, mais ce dernier n'utilise pas ces définitions parce que selon lui nous ne pouvons pas énumérer tous les aspects importants de toutes les situations car ces aspects changent d'une situation à l'autre (Dey, 2001). Dey définit le contexte comme une source très importante d'informations dans l'informatique ubiquitaire. Il a donné une définition plus générale du contexte : « *n'importe quelle information qui peut être utilisée pour caractériser la situation d'une entité. Une entité est une personne, l'endroit, ou l'objet que l'on considère approprié à l'interaction entre un utilisateur et une application, y compris l'utilisateur et les applications eux-mêmes* ». De cette définition, nous pouvons dire que le contexte est un jeu de situations et d'actions. De telles situations qui changent dans le temps, agissent sur les comportements des usagers, l'application et les états environnementaux, chaque fois que des actions spécifiques sont appliquées. D'un autre côté, le travail de Yau et ses collaborateurs (Yau, Huang, Gong, & Yao, 2006) peut être considéré comme une spécialisation de la définition générique de Dey.

En effet, pour eux « *le contexte est composé du système de l'utilisateur ainsi que toutes les informations instantanées, détectables et pertinentes provenant de l'environnement* ».

D'autres travaux ont aussi essayé d'élargir la notion de contexte. La recherche dans l'informatique mobile et ubiquitaire a considérablement élargi cette dernière. Bien qu'il n'y ait pas vraiment d'accord sur ce qui devrait être inclus dans la zone de contexte, la plupart des travaux dans ce domaine se concentre sur une base commune qui inclut la dimension humaine et environnementale (Jameson, 2001; Schmidt, Beigl, & Gellersen, 1999). Dans ces travaux, la dimension environnementale comprend trois dimensions : l'emplacement des usagers, l'infrastructure (les ressources disponibles, les tâches entreprises, etc.) et les conditions physiques (lumière, température, accélération, pression, etc.). La dimension humaine se décompose en trois catégories : le contexte personnel (le pouls des utilisateurs, la pression sanguine, l'humeur, la charge cognitive, etc.), l'environnement social de l'utilisateur, et la tâche des utilisateurs.

Il est très important de rappeler que les travaux en modélisation de l'utilisateur considèrent que les tâches sont une partie du modèle de l'utilisateur, et non du modèle du contexte. Pour éviter toute confusion, il faut préciser que les recherches en cours dans la modélisation du contexte sont divisées en deux communautés de chercheurs qui abordent le contexte selon deux points de vue différents :

- dans la vue centrée sur l'utilisateur, employée dans le domaine de la modélisation de l'utilisateur, la tâche n'est pas une partie du contexte contrairement aux dispositifs matériels qui en font partie.
- dans la deuxième vue centrée matériels, dominante dans l'informatique mobile et ubiquitaire, une gamme de paramètres caractérisant l'état actuel de l'utilisateur est une partie du contexte.

Pour établir une frontière plus objective entre la modélisation de l'utilisateur et celle du contexte, il est utile de remarquer que la modélisation d'utilisateur est concentrée surtout sur les propriétés à plus long terme de l'utilisateur qui sont extraites des observations. D'un autre côté, les modèles de contexte essaient de représenter les caractéristiques actuelles de l'utilisateur et de son environnement qui sont extraites du contexte et des capteurs physiques.

Parmi la multitude de définitions du contexte, il est important d'introduire le travail de Dourish (Dourish, 2004) afin d'avoir une vision claire sur cette notion. Dourish distingue deux vues du contexte : la vue *représentative* et la vue *interactionnelle*.

Il suggère que la vue représentative, dominante dans le traitement de l'informatique ubiquitaire, fasse quatre suppositions clefs (Dourish, 2004). La première est que **le contexte est une forme d'information qui peut être connue et donc codée**. Ensuite, **le contexte est susceptible d'exister**, i.e. pour un ensemble d'applications données, on définit ce qui peut être considéré comme contexte et que l'application supporte. La troisième supposition de cette première vue du contexte proposée par Dourish est que **le contexte est stable**. En effet, il ne varie pas d'une instance à une autre de la même application. Enfin la quatrième, **le contexte est supposé séparable de l'activité**. Autrement dit, selon Dourish ce contexte est l'information qui peut être décrite par un *ensemble d'attributs « appropriés » qui peuvent être observés et collectés*. En outre, ces attributs ne changent pas et peuvent être clairement distingués des caractéristiques décrivant l'activité entreprise par l'utilisateur dans le contexte. Un des problèmes-clés avec cette vue est que la définition du contexte est limitée aux éléments observables. De fait, les travaux adoptant cette perspective ont été principalement confrontés aux difficultés liées à la manière de capturer et de représenter un contexte stable.

La deuxième vue du contexte proposée par Dourish est la vue interactionnelle. Elle adopte une position différente par rapport aux quatre suppositions faites pour la vue représentative. La vue interactionnelle considère le contexte comme « *un dispositif émergent de l'interaction, déterminé par le temps et le contenu* ». Dans cette vue, Dourish suggère que la contextualisation soit une propriété relationnelle, c'est-à-dire qu'une certaine information peut être ou ne pas être appropriée à une certaine tâche. Aussi, au lieu de délimiter et de définir le contexte par avance, cette vue alternative propose que les caractéristiques contextuelles soient définies dynamiquement plutôt que d'une manière statique. Ceci implique qu'un facteur peut être pris en compte dans la définition du contexte parce qu'il est considéré important pour la nouvelle tâche de l'utilisateur. Finalement, plutôt que de supposer que le contexte définit la situation dans laquelle une tâche arrive, Dourish suggère un rapport cyclique entre le contexte et l'activité, où la tâche engendre le contexte (Dourish, 2004).

Lieberman et Selker se rapportent aussi à ce rapport cyclique quand ils déclarent que le comportement des systèmes intelligents (conscients du contexte), auxquels on fournit des entrées explicites (données par un homme) et implicites (le contexte), peut être modifié pour

affecter aussi bien les sorties explicites (résultats de l'interaction) que le contexte lui-même. Ces systèmes doivent percevoir leur environnement, décider quels sont les aspects importants d'une situation et déduire l'intention de l'utilisateur de ses actions concrètes. Les actions du système peuvent être dépendantes de l'heure, de la place ou de l'historique des interactions, autrement dit, du contexte (Lieberman & Selker, 2000). Une différence entre les points de vue de Dourish (vue interactionnelle) et de Lieberman et al. (vue du contexte) concernant le contexte doit être soulignée : pour Dourish la question essentielle est « qu'est-ce que le contexte et comment il peut être représenté dans une application ? » ; Lieberman est concerné par la réalisation et le maintien d'une compréhension mutuelle du contexte avec les actions de l'utilisateur et du système, sans se soucier de la représentation du contexte.

IV. Représentation et modélisation du contexte

Comme nous l'avons présenté dans la section précédente, de nombreux travaux et applications se focalisent sur l'information contextuelle, le plus souvent appelée contexte. Cependant, ces travaux divergent, comme nous l'avons noté, sur la définition même de cette notion ou des éléments qui composent le contexte. Plus encore, cette divergence touche aussi la sémantique de la modélisation du contexte qui diffère d'un travail à un autre. Dans ce qui suit, nous allons donc présenter les différents modèles de représentation du contexte. Ces derniers sont classés en suivant les schémas des structures de données qui sont utilisés pour représenter les informations contextuelles dans leurs systèmes respectifs (Bettini et al., 2010).

- Les modèles attribut-valeur

Ces modèles constituent la structure de données la plus simple et élémentaire pour la modélisation de contexte. Schilit et ses collègues (Schilit et al., 1994) ont utilisé ce formalisme de modélisation du contexte en attribuant la valeur d'une information de contexte (par exemple des informations de localisation) à une application (par exemple une variable d'environnement). L'approche de modélisation attribut-valeur est fréquemment utilisée dans le cadre de services distribués (De Virgilio & Torlone, 2005).

- Les modèles de balisage

Tous les modèles basés sur le balisage utilisent une structure de données hiérarchique composée de balises avec des attributs associés à du contenu. Les profils usagers sont typiquement représentés en utilisant les modèles de balisage (par exemple le modèle CC/PP)

(Kiss, 2006). Cette approche de modélisation du contexte combine l'utilisation du langage XML associé à RDF. En effet, ces approches sont les premières approches de modélisation de contexte à utiliser RDF et incluent des contraintes élémentaires ainsi que des relations entre les types de contexte. Cependant, les types de raisonnement effectués sur ces contraintes élémentaires et leurs relations sont simples et à but spécifique. Ces modèles peuvent donc avoir plusieurs limites, surtout dans la prise en compte de la généralité du modèle de contexte (Strang & Linnhoff-Popien, 2004).

- Les modèles orientés objet

L'approche orientée objet a été introduite dans la modélisation du contexte pour profiter de ses principaux avantages, à savoir l'encapsulation et la réutilisation. En effet, ces dernières permettent de résoudre les problèmes découlant de la dynamique du contexte dans les environnements ubiquitaires. Dans ce type d'approche, les détails du traitement du contexte sont encapsulés au niveau des objets et donc cachés à d'autres composants. L'accès à l'information contextuelle est ainsi fourni uniquement à travers des interfaces spécifiées. Nous pouvons citer l'exemple de l'approche proposée par Hofer et ses collaborateurs pour rendre les appareils mobiles sensibles au contexte (Hofer et al., 2003).

- Les modèles graphiques

L'un des formalismes de modélisation graphique le plus connu est le langage de modélisation unifié (UML), qui a une forte composante graphique (diagrammes UML). Grâce à sa structure générique, UML est également approprié pour modéliser le contexte. Diverses approches existantes utilisent UML pour modéliser les aspects contextuels comme l'exemple du travail de Sheng et Benattallah (Sheng & Benattallah, 2005). Un autre exemple de la modélisation graphique du contexte est celui présenté par Henricksen et ses collaborateurs (Henricksen, Indulska, & Rakotonirainy, 2003). Dans leur travail, les auteurs enrichissent le modèle relationnel graphique classique par une extension pour prendre en compte le contexte en ajoutant une classification contextuelle et une description des propriétés.

- Les modèles basés sur la logique

Cette approche de modélisation du contexte possède un haut niveau de formalisme. En effet, une expression logique définit les conditions dans lesquelles une expression de conclusion ou de fait peut être dérivée (processus de raisonnement ou d'inférence) à partir

d'un ensemble d'autres expressions ou de faits. Pour décrire ces conditions dans un ensemble de règles, un système formel est appliqué. Pour les modèles de contexte basés sur la logique, le contexte est donc défini par des faits, des expressions et des règles. Généralement, les informations contextuelles permettent d'ajouter, de mettre à jour ou de supprimer de nouveaux faits. Le processus d'inférence (aussi appelé raisonnement) peut être utilisé pour dériver de nouveaux faits basés sur des règles existantes dans les systèmes. Une des premières approches a été publiée par McCarthy et Buvac (McCarthy & Buvac, 1998) qui modélise le contexte en utilisant la logique du premier ordre. Nous citons aussi le travail de Stefanidis et ses collaborateurs (Stefanidis & Pitoura, 2008) qui modélisent le contexte en utilisant un ensemble de paramètres contextuels prédéfinis. Dans leur approche, un paramètre correspond à un ou plusieurs prédicats.

- Les modèles basés sur l'ontologie

Les ontologies représentent une description de concepts et de relations entre concepts. Les ontologies ont une expressivité relevée et formelle (basée sur la logique) qui permet l'application de techniques de raisonnement. Elles constituent par conséquent un instrument prometteur et approprié à la modélisation de l'information contextuelle. Une multitude de travaux récents présentent des applications sensibles au contexte se basant sur les ontologies en tant que modèles de contexte. Un des premiers travaux qui s'inscrit dans cette catégorie est celui de (Öztürk & Aamodt, 1997). Nous pouvons citer comme exemple le travail récent d'Apolinarski et ses collaborateurs (Apolinarski, Handte, Le Phuoc, & Marrón, 2011) qui utilisent un modèle de contexte basé sur une ontologie pour implémenter leur gestionnaire de contexte adapté aux dispositifs utilisés dans deux cadres différents : la gestion du trafic et la e-santé. D'autres approches de modélisation du contexte qui se basent sur les ontologies peuvent être consultées dans (Baldauf, Dustdar, & Rosenberg, 2007).

V. Les infrastructures de gestion de contexte

V.1. Qu'est-ce qu'un gestionnaire de contexte

Les gestionnaires de contexte sont les garants de la collecte, de la gestion et de la présentation de l'information de contexte pour le bénéfice des applications. Ce service intermédiaire, ou « *middleware* », est essentiel aux applications afin de gérer l'accès aux sources de contexte en plus du traitement de l'inférence des informations de contexte (cf.

figure I.1). Les gestionnaires de contexte sont principalement utilisés dans le domaine de l'informatique ubiquitaire pour avoir des applications dites sensibles au contexte ou « *context-aware* ». Ce type de systèmes distribués opère dans de tels environnements et doit constamment gérer le contexte dans lequel ces applications sont exécutées afin de détecter les conditions de l'adaptation.

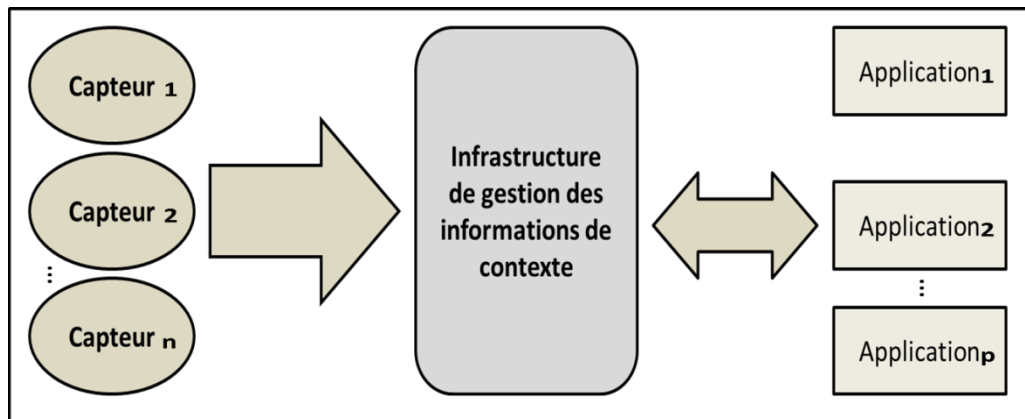


Figure I.1. La gestion des informations contextuelles pour les applications sensibles au contexte

Les travaux liés aux gestionnaires de contexte peuvent être classés selon deux approches distinctes. La première catégorie est centrée sur l'utilisateur et inclut le terminal des utilisateurs, tels que les petits appareils accessibles via un réseau (Coutaz, Crowley, Dobson, & Garlan, 2005). Nous remarquons que tous les travaux récents s'inscrivent dans cette première catégorie. Cependant, la deuxième catégorie a beaucoup été étudiée dans le passé, elle est uniquement basée sur la supervision « système » et plus particulièrement sur l'observation des ressources systèmes distantes (Schroeder, 1995).

Avant de présenter quelques exemples de systèmes proposant ce service de gestion du contexte, nous devons présenter les bases de toutes ces infrastructures. En effet, tous les gestionnaires de contexte ont une architecture commune dans les applications sensibles au contexte : **architecture conceptuelle en couches**. Ainsi, comme nous l'avons mentionné ci-dessus le rôle de ces infrastructures est de collecter puis de gérer et enfin de proposer les informations contextuelles aux applications. Pour cela, une architecture en couches permet une séparation entre la détection et l'utilisation du contexte afin de garantir l'extensibilité et la réutilisabilité des systèmes. L'architecture conceptuelle en couches (cf. Figure I.2) souligne en passant de bas niveau (capteurs) vers le haut niveau (application) les fonctionnalités de raisonnement et d'interprétation des informations contextuelles. C'est-à-dire que nous passons

de l'information brute provenant des capteurs à l'adaptation engendrée par cette même information sur l'interface ou le traitement de l'utilisateur (Ailisto, Alahuhta, Haataja, Kyllönen, & Lindholm, 2002).

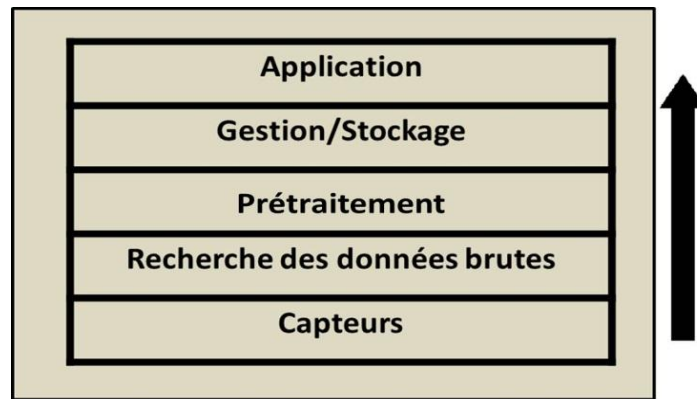


Figure I.2. L'architecture conceptuelle en couches des infrastructures de gestion de contexte

La première couche, de bas niveau, de cette architecture concerne les « *Capteurs* ». Ils sont utilisés pour collecter directement l'information contextuelle brute. Les capteurs ont été classés en trois groupes (Indulska & Sutton, 2003) :

- les capteurs physiques qui vont chercher l'information contextuelle brute (lumière, audio, température ...)
- les capteurs virtuels qui utilisent des services ou les logiciels systèmes, par exemple se baser sur des mouvements de la souris de l'utilisateur pour en déduire sa localisation ;
- les capteurs logiques. Cette dernière catégorie de capteurs est une combinaison des deux premiers.

La seconde couche, de « *Recherche des données brutes* » est responsable de la récupération des données brutes du contexte. Pour cela, elle utilise des pilotes appropriés pour communiquer avec les capteurs physiques en plus des API pour les capteurs virtuels et logiques.

La troisième couche de « *Prétraitement* » n'existe pas toujours dans les systèmes sensibles au contexte. Cependant, elle peut proposer des informations supplémentaires dans le cas où les données brutes seraient trop complexes pour être directement utilisées par les applications. La couche de prétraitement est donc responsable du raisonnement et de l'interprétation des informations contextuelles. En effet, elle permet de faciliter la tâche aux concepteurs des applications en leur proposant une information compréhensible et utilisable

(selon le domaine d'application) au lieu des informations brutes très techniques provenant des deux couches inférieures. Par exemple, la position GPS exacte d'un étudiant pourrait ne pas être utile à une application de campus intelligent, mais le nom de la salle de classe où il se trouve peut l'être. Nous insistons sur le fait que cette couche contient également les fonctionnalités d'agrégation (composition) des différentes informations contextuelles. Évidemment, une valeur unique provenant du capteur n'est souvent pas importante pour une application, alors que l'information combinée pourrait être plus précieuse et pertinente.

La quatrième couche de « *Gestion et de stockage* » organise les données recueillies et les offre, via une interface publique, au client (l'application). Cette couche permet de stocker les informations contextuelles passées, ce que nous appelons l'historique de contexte. Cette couche permet également le stockage temporaire des informations contextuelles courantes, ce que nous appelons « les situations » et qui sont présentées et définies plus loin dans ce même chapitre.

Enfin, la couche « *Application* » est spécifique aux utilisateurs. elle permet l'adaptation réelle en utilisant toutes les données contextuelles remontées par les différentes couches adjacentes. Tous les événements déclenchés par les informations contextuelles spécifiques sont mis en œuvre dans cette couche. Parfois, les couches applications sont séparées des gestionnaires de contexte. Dans ce cas précis, des processus automatiques (exemple : des agents) communiquent avec le gestionnaire de contexte et agissent comme une couche supplémentaire entre le prétraitement et la couche application.

V.2. Exemples de gestionnaires de contexte

Comme nous l'avons présenté dans une section précédente, Schilit et ses collègues ont introduit la notion d'applications sensibles au contexte (Schilit et al., 1994). Le contexte inclut la localisation et les profils des personnes et des objets qui sont dans leurs proximités ainsi que les modifications pouvant intervenir sur ces objets. Pour réaliser ce type de système, ils se sont basés sur leur gestionnaire PARCTAB (Schilit, Adams, Gold, Tso, & Want, 1993). Ce dernier est un dispositif portable muni d'un écran tactile et de trois boutons en interface d'entrée, ainsi que d'un écran et d'un haut-parleur en interface de sortie. PARCTAB ne possède aucune ressource de calcul. Les seules informations contextuelles prises en compte sont la date du jour, l'heure et l'information de localisation des utilisateurs. Ainsi, nous pouvons noter que le contexte de Schilit et ses collègues n'est constitué en réalité que

d'informations sur l'environnement de l'utilisateur afin de lui proposer le service par proximité.

Le travail de Dey et ses collaborateurs est considéré comme l'un des premiers travaux sur la gestion de contexte et ces chercheurs sont en réalité les premiers à avoir généralisé la notion de contexte. En effet, Context Toolkit (Dey et al., 2001) est un canevas logiciel orienté objet empruntant aux IHM les concepts de programmation événementielle et de *widget* pour la collecte du contexte des ressources. Il constitue une boîte à outils capable de proposer aux applications des informations contextuelles avec le niveau d'abstraction souhaité. Context Toolkit rassemble d'autres fonctionnalités comme : un interpréteur afin de composer et abstraire les informations de contexte, un agrégateur pour la médiation avec l'application, un service pour contrôler les actions de l'application sur le contexte et un serveur de noms ou registre appelé *discoverer*.

Le travail de Rey est une autre infrastructure de gestion de contexte décentralisée à base de composants appelés contexteurs (Rey, 2005). Le CONTEXTEUR est construit comme un réseau de ce que l'auteur appelle, « *contexteurs* ». Chaque contexteur est une abstraction logicielle qui va mettre à la disposition du système une information contextuelle détectée à partir des capteurs environnementaux. Cette architecture abstraite modélise la transformation d'observations de phénomènes physiques à partir de l'environnement. En utilisant toutes les informations transmises par les contexteurs, elle permet d'identifier les entités, les relations, les rôles et les attributs pertinents afin d'établir des réseaux de contextes.

Une autre infrastructure dans le même courant est appelée COSMOS. Elle a été introduite par Conan et ses collaborateurs (Conan, Rouvoy, & Seinturier, 2007). Il s'agit d'une approche pour la composition et le traitement des informations de contexte (cf. figure I.3). COSMOS comprend la collecte, l'interprétation et l'identification de situations. Nous notons qu'une situation dans COSMOS est un changement d'état de l'information contextuelle détectée par un des capteurs. Le contexte de COSMOS est statique, ce qui signifie que tous les éléments contextuels doivent être a priori définis et modélisés. Cependant, COSMOS possède davantage de possibilités par rapport au CONTEXTEUR de Rey en rendant les observations et les notifications symétriquement paramétrables.

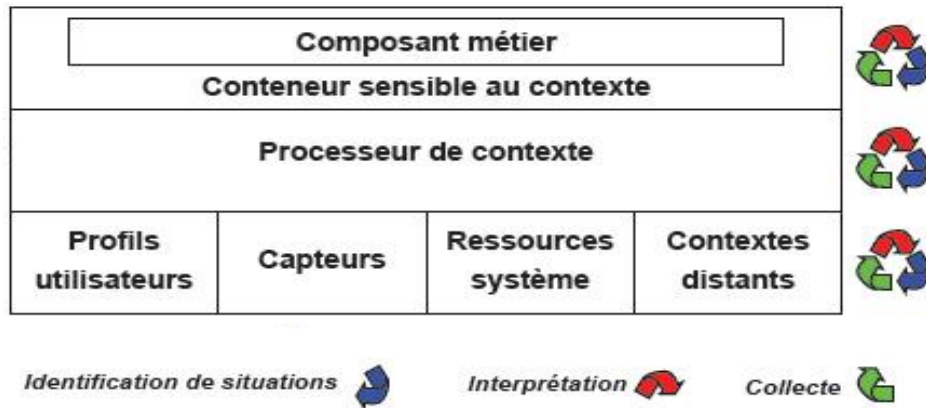


Figure I.3. L'architecture du gestionnaire de contexte COSMOS (Conan et al., 2007)

CASS est un autre exemple de gestionnaire de contexte introduit par Fahy et Clarke (Fahy & Clarke, 2004). Il s'agit d'une structure de gestion de contexte basée sur un serveur centré sur l'utilisateur. Autrement dit, elle est dédiée à la contextualisation des petits appareils portables, offrant une abstraction de haut niveau des informations contextuelles détectées à travers des capteurs distribués. CASS gère à la fois le temps et l'espace, en tenant compte de l'historique du contexte, et fournit un raisonnement sur les informations collectées du contexte. Cependant, cette approche ne tient pas compte des capacités de profilage des utilisateurs ; c'est-à-dire que les informations contextuelles se limitent uniquement aux informations environnementales.

Context Broker Architecture (CoBrA) (Chen, Perich, Finin, & Joshi, 2004) est un gestionnaire spécifique aux applications intelligentes qui gèrent des réunions ou des manifestations. Dans cette approche, le contexte est représenté comme une base de connaissances sur le contexte utilisant les ontologies. CoBrA se base sur un serveur central destiné à partager les informations de contexte entre les différents dispositifs dans un environnement intelligent. Elle possède une architecture basée sur les agents pour assister les applications sensibles au contexte. La spécificité de cette approche est la présence du *Courtier de contexte intelligent* (Brokers) qui entretient et gère un modèle partagé de contexte en faveur d'une communauté d'agents. Ces derniers peuvent être des applications embarquées, par exemple, dans les appareils mobiles des utilisateurs. Pour assurer l'interopérabilité des différents dispositifs et le partage de l'information de contexte, les dispositifs qui utilisent l'architecture CoBrA utilisent la même ontologie, CoBrA ONT, pour décrire leurs informations de contexte. Cette ontologie introduit un vocabulaire spécifique au domaine.

VI. Synthèse des approches et modèles de contexte

Nous avons étudié de nombreux travaux s'intéressant à la gestion du contexte. Ces travaux proposent chacun une solution à la contextualisation des applications pour leur donner une dimension intelligente. Bon nombre de ces travaux ont étudié et comparé les gestionnaires de contexte selon plusieurs critères et différents angles (Bettini et al., 2010; Bolchini, Curino, Quintarelli, Schreiber, & Tanca, 2007) comme par exemple le modèle de contexte ou l'architecture du gestionnaire... Les autres ont présenté les gestionnaires de contexte selon les domaines spécifiques d'application (Marques, Barenholtz, & Charvillat, 2011; Truong & Dustdar, 2009) comme par exemple le domaine médical ou industriel...

Dans la section précédente, nous avons présenté quelques gestionnaires de contexte pour montrer la richesse des approches dans la littérature. Nous résumons ces approches dans le tableau suivant (cf. Tableau I.1) dans le but de catégoriser les gestionnaires de contexte présentés précédemment afin de mieux les appréhender.

Les critères qui sont pris en comptes dans ce tableau sont l'année de la publication, les chercheurs à l'origine du travail. En plus, nous avons ajouté les critères en liaison avec les dimensions prises en comptes par le contexte : l'espace, le temps et le sujet. Cette dernière caractéristique se réfère au point de vue utilisé pour décrire le contexte lui-même; certains modèles décrivent le contexte tel qu'il est perçu par l'utilisateur, tandis que d'autres assument le point de vue applicatif, en considérant l'utilisateur lui-même comme partie du contexte. Dans le même esprit, nous avons mentionné la caractéristique du profil utilisateur (en termes de préférences et caractéristiques personnelles). C'est-à-dire, est-ce que le profil utilisateur est représenté dans le modèle de contexte? Enfin, les autres caractéristiques sont l'architecture du gestionnaire, le formalisme du modèle du contexte, les outils de prétraitement du gestionnaire et l'existence d'un historique du contexte.

Tableau I.1 Comparaison des gestionnaires de contexte

Systèmes	Année	Chercheurs	Espace	Temps	Sujet principal du contexte	Profil utilisateur	Architecture	Modèle de contexte	Prétraitement	Historique de contexte
PARCTAB	1994	Schilit et al.	Oui	Oui	Utilisateur	Non	Décentralisée	Attribut-valeur	Non disponible	Non disponible
Context Toolkit	2001	Dey et al.	Oui	Oui	Application	Non	Basée sur les <i>widgets</i>	Tuples attribut-valeur	Le Service reconnaissance de contexte	Disponible
CoBrA	2004	Chen et al.	Oui	Oui	Application	Oui	Basée sur les agents	Ontologie	Moteur d'inférence et base de connaissances	Disponible
CASS	2004	Fahy et al.	Oui	Oui	Utilisateur	Non	Intergiciel centralisé	Logique	Moteur d'inférence et base de connaissances	Disponible
Le contexteur	2005	Rey	Oui	Oui	Application	Non	Décentralisée à base de composants	Tuples attribut-valeur	Service de reconnaissance utilisant des métadonnées	Disponible
COSMOS	2007	Conan et al.	Oui	Oui	Application	Non	Décentralisée à base de composants	Tuples attribut-valeur	Service de reconnaissance utilisant des métadonnées	Disponible

Depuis ces dix dernières années les travaux qui proposent des approches pour la gestion de contexte ont connu un changement significatif. Les gestionnaires s'avèrent de plus en plus efficaces dans la façon de collecter les informations contextuelles ainsi que la multitude de dimensions prises en compte. Nous pensons que le travail qui reste à faire concerne le contexte à haut niveau. Autrement dit, nous nous demandons comment toutes ces informations issues du contexte vont être réunies dans un même cadre qui sera proposé aux applications sensibles au contexte. Cette agrégation intelligente des informations contextuelles que nous appelons « **situation** » est le cœur de notre intérêt dans ce travail. Pour cela, nous allons présenter dans ce qui suit cette notion de contexte à haut niveau ainsi que les travaux qui se sont intéressés aux situations comme interprétation du contexte exploitable par l'application.

VII. La situation contextuelle

VII.1. Besoin de la situation contextuelle

Les systèmes sensibles aux contextes ont tendance à se transformer de plus en plus en des systèmes sensibles aux situations. En effet, leur difficulté à construire un système qui s'adapte selon les informations contextuelles de bas niveau en est la cause. Ces informations provenant des capteurs physiques ont tendance à changer continuellement et sont parfois dénuées de sens ou incertaines. Ainsi, les informations contextuelles brutes ne permettent pas une prise en compte de l'utilisateur dans son contexte, ni l'adaptation des applications sensibles aux contextes.

Une façon d'aborder ce problème est d'ajouter une couche d'abstraction du bas niveau du contexte en créant une nouvelle couche dans le modèle du gestionnaire de contexte qui permet de créer des situations. Ainsi, le gestionnaire sur la base des données brutes provenant des différents capteurs génèrera des situations en utilisant un raisonnement d'interprétation de contexte. Ces situations peuvent être dans certains travaux similaires à des interprétations sémantiques externes du contexte (Ye et al., 2011). Comparées au contexte, les situations sont significatives, certaines et relativement stables. C'est-à-dire qu'elles ne changent pas lors de modification « mineurs » des informations contextuelles. En plus de cela, faire abstraction des changements des sources contextuelles dans les situations épargne aux systèmes de gestion du contexte les difficultés liées à la résolution des imperfections des informations contextuelles et les changements dans le contexte initial (Bettini et al., 2010). Plus précisément, les

situations ajoutent du « sens » destiné aux applications sensibles au contexte et sont plus faciles à définir et à entretenir que les informations contextuelles brutes. Ainsi, l'adaptation des applications sensibles au contexte est alors déclenchée par le passage à une nouvelle situation. Autrement dit, si le changement d'une valeur de contexte déclenche un changement de situation alors le système pourra déclencher l'adaptation adéquate.

VII.2. Quelques définitions de la notion de situation contextuelle

Dey a défini la situation contextuelle comme « la description des états des entités pertinentes » (Dey, 2001). Nous rappelons que pour Dey une entité peut être une personne, un endroit, un objet pertinent pour l'utilisateur ou pour l'application. Une situation est donc un état à l'instant t d'un contexte.

Jameson (Jameson, 2001) définit également la situation contextuelle dans laquelle un usager effectue sa tâche et selon laquelle le système va personnaliser les informations proposées. Jameson donne l'exemple de l'emplacement de l'utilisateur à l'instant t à partir duquel le système personnalisera les informations. Jameson ne s'arrête pas là, il va ajouter d'autres dimensions au contexte : l'état de l'utilisateur, que ce soit cognitif et/ou psychologique, ou le profil à long terme de l'utilisateur. La situation définie par Jameson ne concerne que la dimension environnementale du contexte, l'emplacement de l'utilisateur au moment de faire sa tâche ou quand il interagit avec le système.

Schmidt (Schmidt, Aidoo, et al., 1999; Schmidt, Beigl, et al., 1999) a lui aussi introduit la situation dans la contextualisation en considérant que le contexte décrit une situation dans laquelle on trouve un usager, le matériel utilisé, la tâche à réaliser et l'environnement. La situation définie par Schmidt n'est pas du tout celle de Jameson qui ne prend pas en compte l'emplacement du processus. Schmidt définit la sensibilité au contexte (*Context awareness*) comme « la connaissance de l'utilisateur et l'état du dispositif informatique, y compris l'environnement, la situation et, à une mesure moindre, l'emplacement ».

D'un autre point de vue, pour Loke (Loke, 2004) la notion de contexte est liée à la notion de situation. Il propose l'agrégation des informations de contexte, afin de déterminer la situation des entités. Ainsi, la situation est considérée comme étant à un niveau plus élevé que le contexte. Nous remarquons que Loke différencie l'activité et la situation. Il considère une activité comme un type d'information contextuelle pour caractériser une situation. Pour lui,

l'activité se réfère généralement à des actions ou des opérations menées par les êtres humains tels que « cuisiner », « courir », « écrire un article ». Loke s'appuie sur les données issues directement des capteurs pour déterminer l'action courante et l'état du périphérique. Il propose une nouvelle approche pour la représentation des situations en découplant les procédures d'inférence du raisonnement sur les données de contexte. En effet, Loke applique une approche de programmation logique pour caractériser les situations, ce qui aide le concepteur des systèmes sensibles au contexte à identifier naturellement les situations d'une application.

Une autre approche dans le même courant (activités, situations) est celui des travaux (Li & Landay, 2008). Ces derniers proposent un paradigme d'interaction pour le système « Ubicomp » basé sur l'activité (l'informatique ubiquitaire basée sur les activités). Dans leur modèle, la relation entre l'activité et la situation est définie comme suit : une activité évolue chaque fois qu'elle est réalisée dans une situation particulière. Une situation est un ensemble d'actions ou de tâches effectuées dans certaines « circonstances ». Dans leur approche, les circonstances sont les informations issues des différentes sources de contexte. D'autres travaux récents se basent sur l'approche de Li et Landay pour définir les situations. Par exemple le travail de (Bruegger, Lalanne, Lisowska, & Hirsbrunner, 2009) dans lequel les auteurs proposent un modèle de situation défini comme suit : une situation est toute activité réalisée dans des contextes.

D'autre part, Thomson et ses collaborateurs (Thomson, Terzis, & Nixon, 2006) ont proposé une approche de détermination automatique des situations à partir des informations contextuelles. Ils ont fourni une bibliothèque réutilisable des caractéristiques des situations. En effet, ils ont exprimé différents niveaux de granularité d'une situation à travers l'héritage des caractéristiques provenant des sources d'informations de contexte. Ainsi, les nouvelles caractéristiques sont créées à partir de caractéristiques existantes afin que la même situation puisse être interprétée à différents niveaux d'abstraction.

Le contexte pour Yau et ses collaborateurs est composé du système de l'utilisateur ainsi que toutes les informations instantanées, détectables et pertinentes provenant de l'environnement. Quant à la situation, dans leur approche (Yau et al., 2006), elle est définie par un ensemble de contextes collectés sur une période de temps et qui est pertinent pour les actions futures des applications. Yau et ses collaborateurs ont analysé la sémantique des situations et leur ont donné des représentations formelles. Une situation peut être atomique ou composée. La situation atomique est une composition de contextes basée sur les opérateurs de

contexte en plus des opérateurs arithmétiques, des opérateurs de comparaisons ainsi que des contraintes de temps. La situation composée, elle est une composition de situations atomiques ou composées en utilisant de la même manière les opérateurs logiques et les contraintes de temps. L'approche de Yau et de ses collaborateurs permet aux concepteurs des applications sensibles au contexte d'utiliser les expressions de la logique formelle pour la spécification des situations.

Costa et ses collaborateurs (Costa, Almeida, Pires, & Van Sinderen, 2007) se sont appuyés sur la composition des informations de contexte dans les situations afin de définir et étudier leurs classifications. Dans leur approche, une situation peut être dérivée d'une seule partie du contexte initial. Dans ce cas, elle est appelée une situation contextuelle intrinsèque. La situation peut être aussi une situation contextuelle relationnelle. C'est-à-dire, elle est le résultat d'une association de plusieurs parties du contexte initial en respectant une certaine logique relationnelle. Dans leur approche, la situation peut être aussi une situation de relation formelle. En d'autres termes, une situation peut être définie directement en appliquant des relations formelles entre deux parties du contexte initial, comme supérieure à, sous-ensemble de, etc. Enfin, une situation peut être aussi une combinaison de plusieurs situations.

La plupart des travaux que nous avons présentés dans cette section se concentrent sur la composition des situations et sur des représentations formelles basées sur ces compositions. Le but de ces travaux est d'inférer des situations, permettant une spécification de haut niveau du comportement humain dans un environnement sensible au contexte, afin de lui proposer par la suite les services les plus adaptés. L'intérêt de l'ensemble des travaux présentés porte sur la façon d'abstraire, de représenter et d'identifier les situations d'après les informations brutes provenant du contexte.

VIII. Conclusion

Nous avons présenté au cours de ce chapitre le cadre général de notre contribution ainsi que le concept sur lequel se base notre travail : le contexte. Nous avons dans un premier temps montré la nécessité du contexte dans des travaux relevant de différents domaines. Néanmoins, nous avons focalisé notre présentation sur des travaux se basant sur cette notion de contexte dans le domaine de l'informatique ubiquitaire qui reste le domaine phare de la modélisation et de l'utilisation du contexte pour la richesse et la diversité de ses approches.

Plusieurs approches de gestion du contexte dans le domaine de l'informatique ubiquitaire ont été présentées dans ce chapitre. Tous les travaux présentés sont différents les uns des autres, mais nous pouvons les regrouper selon trois critères de base. Premièrement, la définition même du contexte est un point de divergence entre les travaux de la littérature. En effet, nous avons montré que plusieurs définitions existent dans la littérature et que la définition du contexte ainsi que les parties qui le composent sont toujours un sujet de controverse entre les chercheurs. Ensuite, le deuxième point sur lequel nous nous basons pour différencier ces approches est la sémantique de la modélisation du contexte. Effectivement, nous avons présenté la majorité des formalismes de modélisation du contexte utilisés dans le domaine de l'informatique ubiquitaire. Enfin, le troisième et dernier point de divergence est la structure des infrastructures de gestion de contexte. En effet, la richesse des travaux de gestion du contexte nous a contraints à en détailler quelques-uns pour souligner les aspects les plus communs entre ces gestionnaires ainsi que les critères de divergence.

Dans notre approche, le concept de situation a un rôle primordial. Comme nous l'avons présenté dans la dernière section de ce chapitre, la situation n'est que le contexte « à haut niveau ». Pour cela nous avons décrit quelques approches se basant sur la situation et proposant également des définitions de ce concept de situation. Nous nous intéressons à la situation car elle constitue un appui stable pour les applications contrairement aux informations contextuelles brutes récoltées directement des différents capteurs. Cependant, ces approches présentent un inconvénient important. En effet, ces travaux doivent inférer des activités des utilisateurs qu'ils ne connaissent pas a priori. Ce traitement nécessite une période d'apprentissage très importante au cours de laquelle plusieurs exemples de chaque situation ainsi que des concepts connexes sont recueillis et analysés. Par ailleurs, dans un contexte métier nous ne rencontrons pas cet inconvénient parce que les utilisateurs effectuent des tâches bien spécifiques et connues au préalable.

Dans le chapitre suivant, nous allons présenter les mêmes notions de contexte et de situation, mais dans le cadre plus précis de notre travail qui est celui de l'accès et la recherche d'information. Le contexte a pris une importance capitale dans ce domaine, et tout particulièrement dans des environnements ubiquitaires.

Chapitre 1.2 : L'accès contextuel à l'information

I.	Introduction.....	57
II.	Accès à l'information.....	58
	II.1. Introduction	58
	II.2. Recherche d'information.....	59
	II.2.1. Phases de la RI.....	60
	II.2.2. Modèles de RI.....	61
	II.3. Filtrage / recommandation d'information	62
	II.3.1. Filtrage basé sur le contenu	63
	II.3.2. Filtrage collaboratif	64
	II.3.3. Filtrage hybride.....	65
III.	Vers l'accès contextuel à l'information	65
	III.1. Problématiques de l'accès à l'information hors contexte.....	66
	III.2. Les systèmes de filtrage (recommandation) contextuels.....	67
	III.3. La recherche d'information contextuelle	68
	III.4. Synthèse sur l'accès à l'information en contexte	69
IV.	Définition et application du contexte dans la RI.....	69
	IV.1. Qu'est-ce que le contexte en RI	70
	IV.2. Les vues du contexte en RI	70
	IV.3. Quelques applications du contexte en RI.....	75
	IV.3.1. Niveau environnemental de l'information.....	75
	IV.3.2. Niveau humain de RI.....	76
	IV.3.3. Niveau interactionnel de la RI.....	77
	IV.3.4. Niveau de la requête	79
	IV.4. Synthèse	81
V.	Les dimensions pertinentes du contexte dans l'accès à l'information.....	82
	V.1. La tâche	82
	V.1.1. La tâche comme motivation de la RI.....	83
	V.1.2. Tâches informationnelles.....	83

V.1.3.	Principales caractéristiques des tâches	86
V.1.4.	Modélisation des tâches	87
V.2.	L'utilisateur	96
V.2.1.	Profil usager	96
V.2.2.	Au-delà du profil usager classique	97
V.3.	Environnement	98
VI.	Conclusion	99

I. Introduction

Dans le chapitre précédent, nous nous sommes intéressés à la notion de contexte ainsi qu'à sa gestion dans le domaine de l'informatique ubiquitaire. Dans ce chapitre, nous nous focalisons sur l'application du contexte dans le domaine qui nous concerne plus particulièrement c'est-à-dire l'accès à l'information.

L'information sur le contexte fournit aux systèmes une base importante afin d'identifier et de comprendre les besoins en information des utilisateurs. Ainsi, en utilisant la notion de contexte, les systèmes d'accès à l'information deviennent de plus en plus efficaces en proposant des résultats plus adaptés aux utilisateurs. Le contexte permet aux systèmes de mieux cerner le besoin informationnel des utilisateurs.

Les approches focalisées sur la contextualisation de l'accès à l'information et plus spécifiquement la recherche d'information sont nombreuses. Néanmoins, ces travaux n'ont pas tous le même point de vue sur l'information de contexte. En effet, ils divergent sur la définition du contexte et également sur les éléments contextuels devant être intégrés dans la contextualisation du processus informationnel. Parmi les éléments les plus importants, nous avons identifié les tâches des utilisateurs, les caractéristiques personnelles de ces derniers, comme leurs centres d'intérêt et leurs préférences, et enfin l'environnement de recherche des utilisateurs qui peut être spatio-temporel, social, etc.

Dans ce chapitre, nous allons dans un premier temps présenter l'accès à l'information qui regroupe le filtrage et la recherche d'information. Par la suite, nous abordons les problématiques majeures de l'accès à l'information hors contexte. Puis nous présentons d'une manière générale la recherche d'information contextuelle ainsi que le filtrage contextuel d'information. Dans un deuxième temps, nous exposons les définitions du contexte les plus connues dans le domaine de la recherche d'information ainsi que les approches qui les exploitent. Enfin, nous nous intéressons aux trois dimensions les plus importantes du contexte de l'accès contextuel à l'information : l'utilisateur, la tâche et l'environnement.

II. Accès à l'information

II.1. Introduction

L'accès à l'information est un terme général qui englobe plusieurs domaines de recherche liés aux sciences de l'information. L'objectif des efforts des chercheurs dans les différents domaines qui s'intéressent à l'accès à l'information est quasiment le même : décrire et comprendre les besoins informationnels des utilisateurs des systèmes informatiques afin de mieux les servir (Belkin, 2000; Gerard Salton & McGill, 1986). Autrement dit, un Système d'Accès à l'Information (SAI) doit fournir les documents et/ou les informations pertinentes aux usagers qui les ont demandées de façon explicite ou implicite. De manière basique, la pertinence de ces informations est déterminée selon des mesures de comparaison entre l'information et la modélisation des besoins des usagers que le système possède.

Dans la littérature nous avons repéré plusieurs techniques d'accès à l'information que nous pouvons classer principalement selon l'interaction de l'utilisateur avec le SAI. Ainsi, deux approches d'accès à l'information sont disponibles : l'approche *push* et l'approche *pull*.

Dans la première approche concernant les systèmes *push*, l'utilisateur exprime implicitement ses besoins en information une seule fois. C'est-à-dire que le système trouve automatiquement et régulièrement de nouvelles informations pertinentes. L'utilisateur est ainsi averti de l'information pertinente à chaque fois que le système la trouve. Les systèmes *push* sont aussi appelés systèmes de filtrage d'information et de recommandation.

La deuxième approche, *pull*, diffère de la première dans le sens où l'utilisateur interroge le système à chaque fois qu'il veut trouver des informations. Dans les systèmes *pull*, l'utilisateur exprime explicitement son besoin en information par la forme de requête composée de termes de recherche en langage naturel. Par conséquent, le système lui fournit une liste de documents jugés pertinents pour cette requête. Les systèmes *pull* sont aussi appelés systèmes de recherche d'information et le représentant le plus connu est sans doute le moteur de recherche.

Dans ce qui suit, nous détaillons les principes sous-jacents à ces deux approches d'accès à l'information.

II.2. Recherche d'information

Les travaux sur la recherche d'information (RI) sont très nombreux et ont débuté il y a longtemps par les travaux de Salton (Salton, 1971) dans les années soixante puis les travaux de Rijsbergen (Rijsbergen, 1979) dans les années soixante-dix et enfin par les travaux de Baeza-Yates dans les années quatre-vingt-dix (Baeza-Yates & Ribeiro, 1999). La recherche d'information est devenue la pierre angulaire de l'accès à l'information que ce soit sur internet ou sur les autres supports électroniques par lesquels les usagers ont accès à de grandes bases d'informations stockées. Ainsi, l'objectif des travaux de recherche en RI est de développer des concepts, des méthodes, des systèmes et des algorithmes qui proposent aux personnes en quête d'information les réponses pertinentes et adéquates quels que soient leurs formats ou leurs emplacements. C'est-à-dire qu'elles doivent être à la portée de tous les usagers en plus d'être dans un format qui est accessible pour chacun.

Selon Peter Ingwersen (Ingwersen, 1992), les travaux en RI se préoccupent surtout des processus impliqués dans la représentation, le stockage, la recherche pour proposer des informations qui sont pertinentes par rapport à un besoin informationnel souhaité par un utilisateur. D'une manière simplifiée, cette pertinence est déterminée en comparant les mots représentant les besoins avec les mots présents dans les documents préalablement indexés. Les systèmes de RI ont été initialement conçus pour être utilisés par des opérateurs qualifiés. Ils sont formés aux langages des requêtes et aux systèmes spécifiques de RI, aux vocabulaires du domaine du corpus des documents recherchés ainsi qu'aux types des documents indexés. Aujourd'hui, l'efficacité du domaine de la RI est visible partout, dans internet à travers les moteurs de recherche, sur nos bureaux informatiques et dans beaucoup d'autres aspects de vie quotidienne, sont à la disponibilité de tous les types d'utilisateurs.

Les notions de base de la RI sont les documents et les requêtes. Les documents représentent toute donnée pouvant pallier le manque informationnel des utilisateurs. Un document peut être sous plusieurs formats possibles comme du texte, une photo, une vidéo, etc. En ce qui concerne les requêtes, elles constituent une formulation du besoin informationnel de l'utilisateur : c'est l'interface entre le système de RI et l'utilisateur. Ce dernier soumet un ensemble de mots clés au système, composant ainsi la requête et décrivant au mieux les thématiques qui décrivent son besoin d'information.

II.2.1. Phases de la RI

Le principal objectif du processus de RI est de chercher les informations qui ressemblent le plus au besoin informationnel de l'utilisateur. Pour atteindre ce but, le système de RI se base sur un ensemble de méthodes pour gérer les documents indexés et préparer une représentation intermédiaire de leurs contenus sémantiques. Le processus mis en œuvre par les systèmes de RI est appelé « processus en U » et consiste en deux phases : l'indexation et l'appariement (illustré dans la figure II.1).

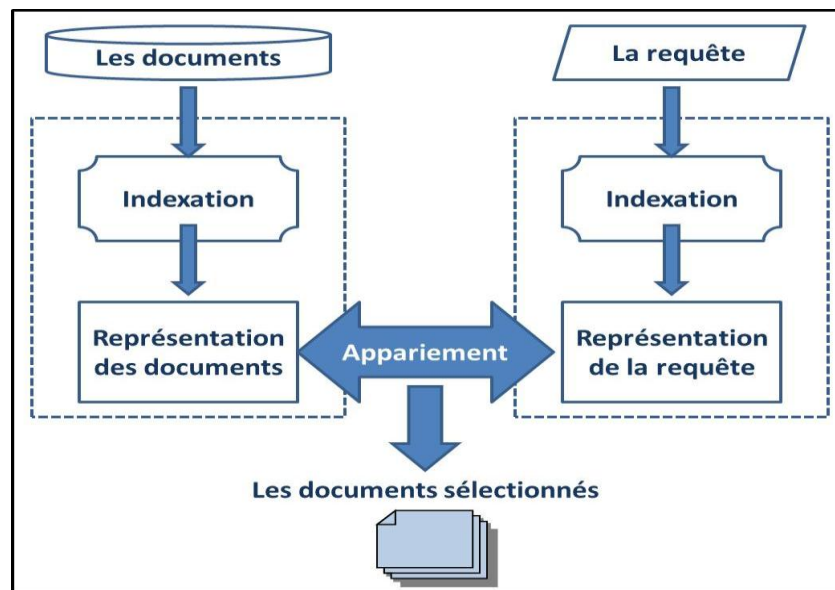


Figure II.1. Le processus en U de la RI

L'indexation est une étape primordiale dans le processus de RI. Elle concerne à la fois les documents et la requête pour en extraire les termes significatifs. L'importance de cette étape réside dans la qualité du résultat de l'indexation qui influence la qualité de la RI. Les représentations de la requête et des documents sont appelées descripteur de la requête et des documents. Le descripteur est le plus souvent la liste des termes assortis du poids représentant leur degré de représentativité du contenu textuel du document qu'ils décrivent (Salton, 1971).

Chaque fois que l'utilisateur exprime son besoin en information à travers sa requête, le système crée un modèle d'index compatible avec celui des documents. Par la suite, le système réalise la deuxième phase de la RI : l'appariement. Le système de RI calcule alors un score de similarité entre la requête et les documents par exploitation des index respectifs et affiche les résultats dits pertinents. Ce score de similarité entre la requête indexée Q et le descripteur d'un document D est généralement noté $RSV(Q,D)$ pour *Relevance Status Value*. Nous notons

ici que par définition la pertinence dans le domaine de la RI est évaluée par une mesure de correspondance entre un document et une requête ou encore par un degré de relation (chevauchement, etc.) entre le document et la requête ; etc.

Deux méthodes d'appariement existent dans la littérature. La première est appelée « *exact match retrieval* », donne comme résultat une liste de documents non triée qui respecte fidèlement la requête donnée (Salton, 1971). La deuxième méthode dite « *best match retrieval* » envoie comme résultat une liste de documents triée selon un score de pertinence vis-à-vis de la requête (Robertson & Jones, 1976).

D'une manière plus précise, le processus de RI ainsi que la pertinence des documents proposés aux utilisateurs dépend profondément du modèle de RI utilisé par les systèmes. Nous allons présenter brièvement dans ce qui suit les trois modèles les plus connus de la RI. Ces modèles ont été étudiés et détaillés dans de nombreux ouvrages (Baeza-Yates & Ribeiro, 1999; Soule-Dupuy, 2001)

II.2.2. Modèles de RI

Modèle booléen

Ce modèle représente la requête sous forme d'une expression logique (Salton, 1971). Autrement dit, les termes d'indexation sont reliés par les connecteurs logiques ET, OU et NON. Ainsi, le rôle principal du processus de recherche d'un système basé sur le modèle booléen est de procéder à des opérations sur les ensembles de documents définis par la présence et l'absence de termes d'indexation, afin de réaliser une similarité exacte avec la requête.

Modèle vectoriel

Ce modèle proposé par Salton (Salton, 1971; Salton & McGill, 1986) repose sur les bases mathématiques des espaces vectoriels. Les termes d'indexation, des requêtes et des documents sont ainsi représentés dans l'espace vectoriel engendré par ces termes. Sous l'angle de ce modèle, le mécanisme de recherche consiste à retrouver les vecteurs documents qui se rapprochent le plus du vecteur requête. La pertinence d'un document relativement à une requête est donc directement liée à la mesure de similarité des vecteurs associés minimisant une distance vectorielle. Sachant que plus un document ressemble à la requête, plus il est susceptible d'être pertinent pour l'utilisateur.

Modèle probabiliste

Ce modèle utilise un modèle mathématique fondé sur la théorie de la probabilité. Robertson et ses collaborateurs (Robertson & Jones, 1976) introduisent le problème de la RI dans un cadre probabiliste. La pertinence document-requête est traduite par le calcul de la probabilité de pertinence d'un document par rapport à une requête. La similarité entre un document et une requête est mesurée par le rapport entre la probabilité qu'un document D donné soit pertinent pour une requête Q , notée $p(R=D)$, et la probabilité qu'il soit non pertinent, notée $p(\bar{R}=D)$, où R est l'évènement de pertinence et \bar{R} de non pertinence. Ces probabilités sont estimées par les probabilités conditionnelles selon qu'un terme de la requête est présent, dans un document pertinent ou dans un document non pertinent.

Nous avons présenté dans cette section les notions relatives à l'accès à l'information classique. C'est-à-dire que les approches d'accès à l'information présentées précédemment sont « non intelligentes » parce qu'elles ne prennent pas en considération l'importance du **contexte**. En effet, le domaine de l'accès à l'information a évolué en tentant d'intégrer des facteurs contextuels. Pour cela, nous présentons dans la section suivante cette notion de contexte dans le domaine de l'accès à l'information ainsi que sa légitimité dans les processus de recherche d'information.

II.3. Filtrage / recommandation d'information

Dans le filtrage d'information, l'utilisateur ne formule pas explicitement son besoin en information. Les préférences de l'utilisateur sont prises en compte afin de lui proposer des informations pertinentes (Belkin & Croft, 1992). Autrement dit, le but principal d'un système de filtrage d'information est de filtrer un flux entrant dynamique d'informations et cela d'une façon personnalisée, pour chaque représentation d'utilisateur, tout en s'adaptant en permanence à son besoin en information. Ainsi, le système de filtrage suggère aux usagers les informations ressemblant le plus à ses attentes à long terme après des phases de collecte, de sélection et de classification de toutes les données disponibles. L'utilisation des systèmes de filtrage (recommandation) est grandissante dans plusieurs domaines comme le commerce électronique...

Les approches de filtrage peuvent être classées en trois grandes catégories (Montaner, López, & De La Rosa, 2003) :

- une première approche, basée sur le contenu, utilise le profil de chaque usager par rapport aux flux d'information et ainsi lui recommande les documents qui sont le plus proches ;
- la deuxième approche dite collaborative génère les recommandations en tenant compte de l'opinion des membres de l'application classifiés en « communautés ». C'est-à-dire que ce type de système regroupe les utilisateurs selon des critères de ressemblance liés à leurs « jugements » passés pour ensuite envoyer des recommandations jugées pertinentes par communauté ;
- enfin, la troisième approche est dite hybride parce qu'elle combine les deux premières approches pour tirer profit au maximum des avantages de chacune.

Ces trois approches sont détaillées dans ce qui suit.

II.3.1. Filtrage basé sur le contenu

Le filtrage basé sur le contenu (*Content-based Filtering*) recommande des éléments qui sont similaires à ceux privilégiés par les utilisateurs dans le passé. En effet, le système définit les items (documents, produits, services ou personnes) par leurs caractéristiques. Les préférences de l'utilisateur, stockées dans le profil utilisateur, sont ainsi liées à ces caractéristiques associées aux items déjà notés par les utilisateurs (Lang, 1995; Pazzani & Billsus, 1997). Autrement dit, chaque profil utilisateur qui décrit ses centres d'intérêt est construit de façon manuelle ou automatique. Selon Schafer et ses collaborateurs (Schafer, Konstan, & Riedi, 1999) la technique de filtrage basée sur le contenu est aussi appelée technique de « corrélation item à item ».

Nous pouvons citer quelques travaux classiques basés sur l'approche de filtrage basée sur le contenu. *NewsWeeder* (Lang, 1995) est un système de filtrage pour les groupes d'actualités. Il recommande les actualités non lues pour les utilisateurs en fonction des notes dans les articles qui ont déjà été lus. Dans un autre champ d'application, Pazzani et ses collaborateurs proposent un système qui recommande des sites internet sur des thèmes qui pourraient intéresser chaque utilisateur (Pazzani & Billsus, 1997).

Les avantages de ce type d'approches est qu'elles répondent efficacement aux intérêts à long terme des utilisateurs en utilisant des techniques d'apprentissage pour la mise à jour des profils et leurs liens avec les contenus des documents disponibles (Montaner et al., 2003). Cependant, le filtrage basé sur le contenu possède quelques inconvénients comme, à titre

d'exemple, celui du démarrage à froid. En effet, à l'initialisation du système, le profil des utilisateurs peut être différent de la réalité car ces derniers ont des difficultés à définir les thèmes qui pourraient les intéresser.

II.3.2. Filtrage collaboratif

Le filtrage collaboratif (*Collaborative Filtering*) propose aux utilisateurs des items que d'autres utilisateurs, ayant les mêmes goûts et préférences, ont aimés dans le passé. Le profil utilisateur est constitué d'informations et de notes de leurs utilisateurs respectifs. Selon (Schafer et al., 1999) la technique de filtrage collaboratif est également appelé « corrélation personne à personne ». Cette approche est l'approche de filtrage la plus fréquemment utilisée et mise en œuvre dans les systèmes de recommandations.

Les travaux qui se sont intéressés aux systèmes de recommandations classiques se basent sur la notion de notes comme un moyen de capturer les préférences des utilisateurs pour différents items (articles, morceaux de musique, film, etc.). Ces notes sont soit explicitement fournies par les utilisateurs ou implicitement déduites de leurs actions. Une fois ces évaluations initiales définies, le système de recommandation tente d'estimer la fonction de prédiction R ($Utilisateurs \times Items \rightarrow Note$) pour les paires (utilisateur, item) qui n'ont pas été initialement notées par les utilisateurs. Nous remarquons que *Note* est un ensemble ordonné d'entiers non négatifs ou réels dans une certaine fourchette, et *Utilisateurs* et *Items* sont les domaines respectifs des utilisateurs et des items. Enfin, le système de recommandation peut recommander les éléments prédits (les mieux notés) pour un utilisateur. C'est ce que l'on appelle « *systèmes de recommandation* » traditionnels ou aussi à « *deux dimensions* » (Ricci, Rokach, Shapira, & Kantor, 2011).

Nous pouvons donner quelques travaux classiques basés sur l'approche de filtrage collaboratif. *Ringo* (Shardanand & Maes, 1995) recommande des albums musicaux et des artistes en se basant sur les similitudes entre les goûts de l'utilisateur et ceux des autres utilisateurs. Nous pouvons également citer le travail de Goldberg et ses collaborateurs (Goldberg, Nichols, Oki, & Terry, 1992) qui exploitent les traces d'usage des utilisateurs. Ainsi, les auteurs se basent sur quelles ressources consultées par quel utilisateur afin de filtrer les documents électroniques.

Dans cette approche, les utilisateurs profitent des évaluations de tous les autres utilisateurs du système et ainsi des recommandations pertinentes basées sur ces ensembles de

communautés. Le succès de cette approche est principalement dû au fait qu'elle ne requiert pas d'information a priori sur les ressources, ni sur les utilisateurs. Par conséquent, toutes les ressources de n'importe quelle nature que ce soit ou médias (texte, audio, vidéo) peuvent être exploitées. Cependant, l'un des principaux inconvénients de cette approche est l'expression limitée du besoin informationnel des utilisateurs. En effet, un changement dans le besoin de l'utilisateur ne peut être exprimé que par des évaluations des documents qui lui sont proposés par le système. Ainsi, le principal problème est lié au fait que ces mêmes documents ne lui donnent pas la possibilité d'exprimer ses nouveaux besoins.

II.3.3. Filtrage hybride

Les systèmes de filtrage dits « hybrides » sont définis par beaucoup de chercheurs, tels que (Adomavicius & Tuzhilin, 2005; Burke, 2002), comme une technologie qui applique plusieurs techniques de filtrage. Généralement, la technique de filtrage collaboratif est combinée avec d'autres techniques. Par exemple, le système hybride peut gérer des profils d'utilisateurs orientés contenus, en plus de la comparaison des profils individuels qui donne lieu à la formation de communautés d'utilisateurs permettant le filtrage collaboratif.

Parmi les travaux classiques utilisant ce type d'approche hybride, le système de recommandation *FAB* (Balabanović & Shoham, 1997) recommande des pages Web pour les utilisateurs en fonction des 100 mots les plus importants sur la page Web. Le profil utilisateur est composé de pages aimées par l'utilisateur, leur poids respectif selon les termes qui en sont extraits et comparés à ceux figurant dans ce même profil. Dans ce système de filtrage hybride, l'approche de filtrage collaboratif est combinée avec l'approche basée sur le contenu.

III. Vers l'accès contextuel à l'information

L'intégration du contexte dans le processus de RI est très importante. Le système ne peut pas afficher le même résultat pour deux usagers appartenant chacun à un domaine complètement différent, uniquement parce qu'ils ont saisi la même requête. De plus, comme cela fut souligné par Johnson (Johnson, 2003), le contexte est un facteur crucial dans la formation du processus d'utilisation de l'information : selon le contexte, les processus d'utilisation de l'information peuvent être très différents. Cela implique que le contexte devrait être pris en compte dans les études sur l'utilisation de l'information.

III.1. Problématiques de l'accès à l'information hors contexte

Dans la plupart des cas, les utilisateurs en quête d'information spécifient leurs besoins en information en donnant quelques mots qui composent leur requête. En moyenne, la longueur d'une requête se situe entre 2 et 3 mots (Spink, Jansen, Wolfram, & Saracevic, 2002). Partant de ce constat, les systèmes classiques d'accès à l'information doivent en plus faire face au volume d'information (sous format électronique) qui ne cesse de croître avec l'explosion de l'utilisation d'internet. En effet, les données s'accumulent partout, comme sur le World Wide Web, dans des domaines spécialisés comme des bibliothèques numériques, des organisations privées et des collections personnelles (au bureau, à la maison, etc.). Ces systèmes doivent proposer des documents relativement pertinents parmi la multitude de documents potentiellement intéressants pour une requête donnée et cela indépendamment du contexte de recherche de l'utilisateur. Ainsi, l'une des premières problématiques de l'accès à l'information hors contexte est la surcharge informationnelle qui frappe les utilisateurs en quête d'information, rendant le résultat envoyé par les SAI quasiment non exploitable. Autrement dit, les utilisateurs sont souvent perdus devant des listes de résultats plus longues les unes que les autres et passent généralement à côté des documents les plus pertinents. D'un autre côté, les SAI classiques ne considèrent que la requête comme expression du besoin informationnel de l'utilisateur. Cependant, il est prouvé que les utilisateurs ont souvent des difficultés à articuler leurs besoins d'information (Belkin, 2000). Ainsi, les SAI classiques ne peuvent pas satisfaire les usagers parce qu'ils ne prennent pas en compte leurs préférences de recherche ou centres d'intérêt et leur contexte. Ils doivent donc évoluer prendre en compte la dimension utilisateur (orientés utilisateurs) et intégrer le contexte de la recherche.

Budzik et Hammond déterminent trois types de problèmes issus de l'interprétation de la requête hors contexte (Budzik & Hammond, 2000) :

- pertinence du but actif : les buts actifs de l'utilisateur contribuent significativement à l'interprétation de la requête et aux critères pour juger une ressource appropriée à celle-ci ;
- ambiguïté du sens du mot : le sens d'un mot peut être différent d'un scénario à un autre. Par exemple, le mot « Python » peut causer une ambiguïté selon son utilisation dans un contexte informatique ou le langage animalier (serpent). Le contexte de la requête participe à choisir le sens de mot ;

- convenance de l'utilisateur final : l'utilisateur final contraint le choix de résultats. Les sources appropriées à un vétérinaire ne seront probablement pas appropriées à un écolier du primaire.

Ajoutant à ce qui a été dit plus haut, d'autres problématiques ont vu le jour ces dernières années. Un des nouveaux challenges de l'accès à l'information est l'apparition de l'accès mobile à l'information qui rend encore plus difficile le travail des SAI classiques. En effet, ces derniers doivent interpréter les besoins de l'utilisateur qui sont encore plus impactés par le contexte de l'utilisateur (exemple : recherche un restaurant selon le lieu). Puisque les informations recherchées par l'utilisateur dans ces environnements mobiles sont souvent reliées à leur contexte actuel ou à un contexte futur anticipé, il est primordial pour les SAI d'incorporer des méthodes par lesquelles ces données pourraient être intégrées dans le processus de RI (Jones, 2005).

III.2. Les systèmes de filtrage (recommandation) contextuels

La contextualisation a touché également les systèmes de recommandation. En effet, ce domaine a connu une nouvelle génération de systèmes de recommandation qui prennent en considération en plus du couple (utilisateur, item) des informations contextuelles supplémentaires tels que le temps, le lieu, etc. C'est ce qu'on appelle des systèmes de recommandation sensibles au contexte. Ces derniers ont pour but principal de prédire les goûts et les préférences des utilisateurs en intégrant les informations contextuelles dans le processus de recommandation. En effet, la nouveauté de ces systèmes est d'ajouter une dimension de données supplémentaire à la fonction d'évaluation R qui passe de deux à trois dimensions : $Utilisateurs \times Items \times Contexte \rightarrow Note$ (Adomavicius & Tuzhilin, 2008). Cependant, l'originalité des travaux qui proposent des systèmes de recommandation sensibles au contexte réside dans le choix d'informations contextuelles et dans leurs combinaisons avec les préférences à long terme des utilisateurs pour des items donnés. Ainsi, les informations contextuelles *Contexte* peuvent être de différents types, chaque type définissant un certain aspect du contexte. De plus, chaque type de contexte peut avoir une structure complexe qui reflète la nature complexe de l'information contextuelle. Une des structures de l'information contextuelle la plus utilisée est la structure hiérarchique qui peut être représentée par des arbres, comme cela se fait dans la plupart des systèmes de recommandation sensibles au contexte (Adomavicius & Tuzhilin, 2008).

III.3. La recherche d'information contextuelle

Les approches de RI s'orientent de plus en plus vers la contextualisation des Systèmes de RI pour pallier les limitations des systèmes de RI classiques présentés dans la section III.1. Ces nouveaux moteurs de recherche sont *orientés usager* en particulier et *contexte de l'utilisateur* en général. Ils ont comme objectif principal de combler le besoin en information des utilisateurs tout en prenant en compte leur contexte de recherche dans les processus de RI. Dans (Crestani & Ruthven, 2007), Crestani et Ruthven indiquent que le contexte de la recherche influence le processus de la RI. Selon eux, le contexte des utilisateurs cherchant l'information affecte l'interaction avec le système RI, le type de réponse attendu et l'usage des objets informationnels extraits par le système.

Pour présenter de manière générale la RI contextuelle, étant donné qu'elle fera l'objet de la section IV de ce chapitre, nous pouvons dire de la notion de contexte dans ce domaine couvre une multitude de dimensions très larges. En effet, les travaux ne convergent pas dans la définition même de ce contexte de la RI. Ainsi, en plus du profil de l'utilisateur, nous pouvons trouver le contexte de la requête, les préférences de recherche, le contexte d'interactions avec le système ainsi que le contexte temporel et géographique de la recherche...

De plus, les modèles classiques de la RI ont évolué pour intégrer ces nouvelles dimensions contextuelles dans le processus de recherche. C'est le cas du modèle probabiliste qui a connu une extension pour intégrer les informations de contexte (Wen, Lao, & Ma, 2004). Ainsi, pour adapter les données du contexte dans le cadre de la RI textuelle, des travaux de RI contextuelle ont représenté les requêtes, le contexte et le document par des termes (mots). De cette manière, le contexte est simplement combiné avec la requête pour étendre la requête. Néanmoins, cette extension du modèle probabiliste est limitée par les informations de contexte qui peuvent être non compatibles avec les requêtes. Pareillement, le modèle vectoriel a également connu une extension pour intégrer les informations de contexte dans la RI. Parmi les travaux qui s'inscrivent dans cette direction, nous pouvons citer ceux de Melucci (Melucci, 2005) qui propose de modéliser le contexte par un vecteur. Autrement dit, le contexte est modélisé dans un espace vectoriel et son évolution est modélisée par des transformations linéaires. Ainsi, pour mettre en relation chaque document ou requête avec le contexte, nous utilisons une des bases de vecteurs correspondant aux différents contextes.

III.4. Synthèse sur l'accès à l'information en contexte

Il existe une relation étroite entre le contexte, l'information (contenu des documents) et les utilisateurs. Le contexte peut être lié à la structuration des documents ainsi qu'aux contenus des documents. Le contexte concerne d'une façon directe l'accès (précédent et courant) aux documents. Le contexte peut être soit le contexte passé de l'utilisateur, soit le contexte actuel de l'utilisateur et, éventuellement, une prédiction sur le contexte futur de l'utilisateur (Brown & Jones, 2002). Le contexte de recherche de l'utilisateur est devenu indispensable dans le processus d'accès à l'information.

La plupart des travaux actuels en RI se sont mis d'accord sur le fait que le contexte doit être intégré dans les processus informationnels afin de rendre les résultats plus pertinents. Effectivement, cette notion subjective est indiscutablement la question fondamentale posée dans les systèmes d'accès à l'information en général et particulièrement dans la RI. La pertinence dépend surtout du point de vue de l'utilisateur et de son jugement sur les informations proposées par les systèmes sollicités. Les Systèmes de RI contextuels, se focalisent essentiellement sur deux types de pertinence. La première pertinence dite « *situationnelle* » compare l'information proposée par le système au but de recherche de l'utilisateur. La deuxième dite « *cognitive* », est la pertinence liée au thème de la requête mais en même temps liée aux connaissances de l'utilisateur sur ce thème (Borlund, 2003).

Nous avons présenté l'intégration du contexte dans le processus d'accès à l'information ainsi que les avantages d'une telle pratique. Cependant, la notion même de contexte dans le processus informationnel reste à définir. En effet, les approches qui se focalisent sur le contexte et son utilisation dans le processus d'accès à l'information ont chacune leur propre vision du contexte ainsi que de ses composantes. C'est ce qui nous a motivé à présenter, dans la section suivante, un récapitulatif des définitions du contexte dans la littérature du domaine de la RI.

IV. Définition et application du contexte dans la RI

Nous avons présenté la notion de contexte d'une manière globale dans le chapitre I.1. Ainsi, nous nous sommes avant tout intéressés aux travaux définissant le contexte dans le domaine de l'informatique ubiquitaire. Dans cette section, nous allons particulièrement nous

intéresser au contexte de la RI. Par conséquent, nous allons exposer les différentes visions des études traitant de cette notion issues de la littérature.

IV.1. Qu'est-ce que le contexte en RI

Le processus de recherche peut être amélioré en comprenant et modélisant le contexte. Le contexte peut être donc exploité pour rendre plus flexible la formulation du besoin informationnel et la consultation des résultats par les usagers. Des variables contextuelles importantes sont incluses comme par exemple (Budzik, Hammond, & Birnbaum, 2001) :

- l'emplacement ;
- les contextes de l'utilisateur lui-même (son intérêt à court et long terme, son état mental et son comportement lors de la recherche...) ;
- la qualité de l'information (sa source, son format, etc.) ;
- la tâche ;
- les contextes sociaux dans lesquels des besoins en l'information surgissent...

Au-delà de ces variables contextuelles, la dimension temporelle du contexte a aussi une importance cruciale. En effet, la durée du contexte de RI peut ainsi varier d'un laps de temps à une très longue durée et redéfinir ainsi les variables contextuelles les plus pertinentes (Ingwersen & Järvelin, 2005a).

En plus de cela, le type de contexte dépend de l'entité ou de l'acteur auquel il est destiné. Ainsi, une question fondamentale se pose : nous traitons **le contexte de quoi ou de qui pour contextualiser le processus de RI**. En effet, cette question est très importante puisque les informations contextuelles qui doivent être prises en compte et intégrées dans le processus de RI en dépendent (Göker, Myrhaug, & Bierig, 2009).

Dans ce qui suit, nous allons présenter dans un premier temps les différents travaux qui s'intéressent au contexte et à ses dimensions dans le domaine de la RI. Dans un deuxième temps, nous allons exposer les applications les plus représentatives du contexte dans des approches de RI.

IV.2. Les vues du contexte en RI

Les contextes dans lesquels une personne cherche l'information sont composés de facteurs cognitifs, sociaux et d'autres facteurs liés aux tâches, aux buts et aux intentions, qui

précèdent l'(es) épisode(s) de RI. Le domaine de la RI, lui, forme selon Ruthven, un terrain de test idéal pour expérimenter la contextualisation d'un point de vue informatique (Ruthven, 2008). Il n'existe pas comme dans le cadre général de définition unique du contexte ou de ses composants pour le processus de RI. Dans cette partie, nous présentons les différentes définitions du contexte dans le domaine de la RI.

Allen (Allen, 1997) a proposé une structure pour comprendre la situation d'un individu au moment de la quête de l'information. L'auteur propose une matrice qui unit deux types de variables, les variables individuelles et les variables sociales. Il souligne les interdépendances de ces deux types de variables :

- des variables individuelles incluent des structures de connaissance, des styles cognitifs et des traits de personnalité ;
- tandis que des variables situationnelles incluent, par exemple, l'environnement spécifique de la tâche.

Plus tard, des études ont repris le travail d'Allen afin d'essayer de définir un contexte pour la RI. Ainsi, Cool (Cool, 2001) décrit le travail d'Allen et qualifie sa matrice des facteurs sociaux et individuels de support pour expliquer les comportements de recherche et les besoins en information. Ce qui est à retenir du travail d'Allen dans la conceptualisation du contexte, est la métaphore de la matrice. En effet, le fait de considérer les éléments du contexte comme matrice permet de comprendre la large variété des combinaisons d'éléments contextuels et d'identifier la difficulté face à toutes les combinaisons possibles de ces éléments contextuels. Cela permet essentiellement a posteriori de faire un lien entre les combinaisons diverses de ces éléments et le comportement de RI (Kelly, 2006).

Quant à Johnson, il a exploré plus en détail trois sens du contexte utilisés dans le domaine de la RI (Johnson, 2003) :

- le contexte comme l'équivalent de la situation dans laquelle un processus est immergé. Les situations sont considérées plus importantes pour déterminer le comportement en RI que les traits individuels ou les dispositions. Ainsi, la **définition situationnelle** du contexte est tout simplement une représentation plus complexe de l'environnement dans lequel une personne débute un processus informationnel ;
- les approches imprévues au contexte détaillent les facteurs situationnels actifs produisant les états prévisibles en RI. Ces approches supposent que l'efficacité des

entités (telle que l'individu) est décidée par l'appariement convenable entre leurs caractéristiques, particulièrement structurelles, et l'environnement dans lequel ils se trouvent. Par exemple, pour accomplir un processus de RI, un usager doit avoir des outils informationnels disponibles lui permettant d'accomplir sa tâche. Ces outils, par exemple, sont les facteurs situationnels actifs spécifiés par les approches imprévues et qui ont un effet sur le comportement de recherche ;

- le contexte vu comme les différents cadres dans lesquels des échanges informationnels tels que des débats et des dialogues arrivent au sein d'organisations. Ces cadres du contexte sont les discussions, le cadre organisationnel, l'état émotionnel, les facteurs culturels.

Johnson a aussi divisé le contexte pour l'accès à l'information en trois niveaux : macro, local et individuel (Johnson, 2003) :

- le *macro-niveau* est celui dans lequel il met les informations sociétales, les informations technologiques, l'architecture et les tendances institutionnelles ;
- le *niveau local* contient le contenu, les contraintes de recherche, et le domaine d'information ;
- le *niveau individuel* contient les responsabilités individuelles, les programmes de décision, les opportunités individuelles, les procédures de recherche, le niveau de motivation de l'usager, l'étape de recherche, la logique et enfin l'expérience des individus.

Ingwersen et Järvelin ont une autre vision du contexte. Cette vision est orientée vers l'usager dans l'accomplissement de sa tâche de travail ayant un besoin informationnel, i.e. elle englobe une(des) tâche(s) informationnelle(s) (Ingwersen & Järvelin, 2005b). Pour les deux auteurs, le contexte se décompose en quatre couches imbriquées :

- la première couche du contexte est le *contexte socio-organisationnel et culturel* ;
- la deuxième couche est le *contexte de la tâche métier* ;
- ensuite la couche du *contexte du seeking* (accès à tous types de support informationnel) ;
- et enfin le *contexte de RI (searching)*.

Ces quatre niveaux permettent d'arranger les variables contextuelles pertinentes pour le processus de travail et le processus informationnel qui lui est associé. Pour chacun de ces

niveaux, les auteurs mettent en évidence l'interaction de l'utilisateur avec un épisode du processus de travail, du processus *seeking* ou du processus de recherche (*searching*).

Ingwersen et Järvelin développent une infrastructure contextuelle cognitive dans le but d'étudier les dimensions du contexte qui ont un impact sur le processus de RI. Cette infrastructure inclut de larges classes contextuelles comme la structure des objets, la session de recherche, le contexte techno-économique et sociétal, le contexte social, systémique, la tâche de travail et des contextes émotionnels. Ingwersen et Järvelin proposent une dimension « **historique du contexte** », qui représente la position des usagers dans ces différentes couches contextuelles (Ingwersen & Järvelin, 2005b). De cette manière, les auteurs admettent que le contexte d'un usager peut changer à chaque fois qu'une nouvelle recherche est lancée, qu'une nouvelle série de résultats est examinée, et qu'un nouveau document est visionné.

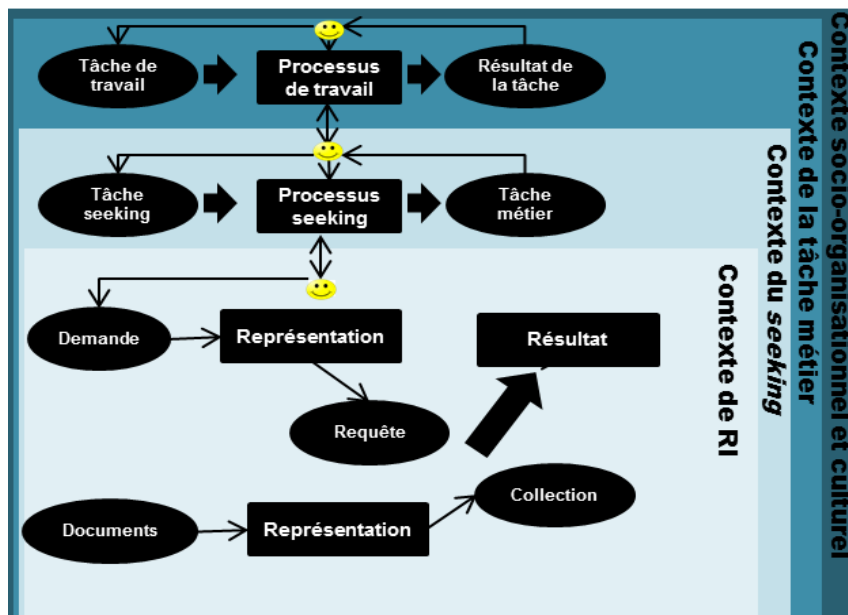


Figure II.2. Les quatre couches imbriquées du contexte (Ingwersen & Järvelin, 2005b)

Une autre taxonomie multidimensionnelle du contexte a été élaborée récemment dans (Tamime-Lechani, Boughanem, & Daoud, 2010). Cette taxonomie décompose le contexte de la RI en quatre dimensions :

- la dimension du *contexte mobile* qui regroupe l'environnement matériel de recherche ainsi que la position géographique de l'utilisateur ;
- la dimension du *contexte de l'utilisateur* qui réunit les deux sous dimensions liées au contexte personnel de l'utilisateur et son environnement social ;

- la dimension du *contexte de la tâche des utilisateurs* qui comprend les aspects en relation avec l'intention de recherche de l'utilisateur ou le but à accomplir par cette recherche ;
- la dimension du **contexte du document** qui regroupe les aspects contextuels liés soit au contexte qualitatif du document (structurels, la forme, etc.), soit aux caractéristiques des sources qui les proposent (crédibilité, sa fiabilité, etc.).

Dans une autre définition plus élaborée du contexte en RI, Cool et Spink identifient les quatre axes principaux (Cool & Spink, 2002) :

- le *niveau environnemental de l'information* concerne l'environnement de recherche. Ce dernier est en étroite liaison avec tous les facteurs cognitifs, sociaux ou professionnels qui influencent le comportement de recherche de l'utilisateur et donc son jugement de la pertinence du résultat ;
- le *niveau humain* de la RI englobe tous les facteurs contextuels liés aux connaissances de l'utilisateur en plus de ses intentions de recherche etc. ;
- le *niveau interactionnel* de la RI concerne l'interaction utilisateur avec le système de RI et souligne l'impact des variables contextuelles sur la rétroaction ou les jugements de pertinence de l'utilisateur ;
- le *niveau de la requête* concerne le niveau de requête ou le niveau linguistique du contexte. Dans ce dernier niveau, les auteurs s'intéressent aux travaux qui se focalisent sur l'interprétation et la reformulation des requêtes par les systèmes de RI en utilisant des variables contextuelles.

D'autres travaux ont proposé une définition plus spécifique du contexte en RI. Ce dernier y est alors décortiqué et ses facteurs les plus pertinents sont choisis selon le contexte métier des usagers. Par exemple, Freund et ses collaborateurs (Freund, Toms, & Waterhouse, 2005) présentent une vue du contexte pour un comportement de RI spécifique, celui des ingénieurs logiciels. Leur modèle final tient compte de quatre axes principaux qui influencent le comportement de RI :

- *consultant* (l'utilisateur) : son expertise, ses critères, son savoir et son rôle ;
- *engagement* : dans cette partie très spécifique aux ingénieurs informatique, les caractéristiques prises en compte par les auteurs sont : le système, l'étape du projet, le client et la durée de l'engagement ;

- *tâche de travail* ;
- *problème* : les buts d'information qui permettent de résoudre les problèmes. Un problème engendre un but d'information à atteindre pour le résoudre.

Cette décennie a donc connu un élargissement considérable de la définition du contexte en RI et de ses facteurs les plus pertinents. Le contexte qui se limitait à la plateforme utilisateur et à sa localisation prend une autre dimension. La plupart des travaux se concentrent sur un cœur commun qui inclut l'environnement cognitif et les dimensions humaines, le besoin en information et l'interaction liée à l'activité de la RI. Cependant, tous ces travaux, ne se sont pas mis d'accord définitivement sur les éléments qui doivent être inclus dans le contexte pour la RI (Brusilovsky & Millán, 2007).

IV.3. Quelques applications du contexte en RI

Les quatre dimensions du contexte proposées par Cool et Spink ont été largement utilisées dans la littérature. Ces quatre niveaux peuvent constituer une structure pour classer tous les travaux de la littérature qui s'inscrivent dans la contextualisation du processus de recherche et d'accès à l'information. Dans cette section, nous allons synthétiser les travaux de recherche liés au contexte pour l'accès à l'information. Ces travaux seront classés dans le cadre du contexte proposée par Cool et Spink : le niveau environnemental de information, le niveau humain de RI, le niveau interactionnel de la RI et le niveau de la requête (Cool & Spink, 2002).

IV.3.1. Niveau environnemental de l'information

Le contexte peut être perçu comme l'environnement de l'information dans lequel l'information est exploitée. Quelques exemples concrets du cadre environnemental du contexte pourraient être : l'organisationnel, l'institutionnel ou des paramètres de la tâche de travail.

Les travaux dans ce cadre explorent la dimension sociale ainsi que des facteurs extérieurs au système qui influencent les comportements humains, y compris les comportements de RI (exemple : les interactions avec les systèmes de RI).

Taylor (Taylor, 1991) a été le premier à employer le terme d'« environnements d'utilisation de l'information » pour décrire le contexte de travail dans lequel les individus sont

confrontés au choix des documents pertinents. Pour Taylor, l'environnement d'utilisation de l'information se définit comme : « Ces éléments qui : (a) affectent le flux et l'utilisation de messages de l'information à l'intérieur et à l'extérieur de n'importe quelle entité définissable; et (b) déterminent les critères selon lesquels on jugera la valeur des messages de l'information. ». En fin de compte, le contexte est décrit comme ayant une incidence sur les informations sélectionnées pour un traitement, ainsi que sur les informations jugées pertinentes ou utiles pour un utilisateur à un moment donné. Ainsi, nous pouvons dire que les approches méthodologiques qui explorent le contexte de l'information au niveau environnemental sont longitudinales et/ou liées à des facteurs naturels. Dans ces études, les enquêteurs explorent les besoins réels en information des utilisateurs dans le contexte particulier au sein duquel ces besoins surgissent et à l'endroit où le processus de RI prend place.

IV.3.2. Niveau humain de RI

Le niveau humain de RI inclut le(s) but(s) qu'une personne essaye d'atteindre, ou les tâches de résolution du problème qui influencent le niveau d'interaction avec l'outil permettant de réaliser la RI.

Ce niveau est le plus proche de la RI. À ce niveau, les travaux se préoccupent de la place de l'utilisateur qui réalise une procédure de RI, en respectant ses problèmes courants et son comportement informationnel. Autrement dit, le contexte pour ce niveau se traduit par le comportement informationnel des personnes dans une situation problématique quelconque, ou dans l'environnement du couple tâche/but.

Pour Belkin (Belkin, 2000), et depuis plus de 10 ans, les systèmes de RI doivent être un support continu à l'accomplissement des tâches de l'utilisateur. Pour cela, l'intelligence des systèmes de RI réside dans la bonne attribution des rôles et des responsabilités aux trois composantes principales dans le processus de recherche, qui sont :

- *l'utilisateur dans le système;*
- *la source de connaissance* (exemple : documents) à laquelle l'utilisateur a accès et avec laquelle il interagit;
- *les médiateurs* certaine(s) personne(s) et/ou périphérique(s) qui supporte(nt) les interactions de l'utilisateur avec les ressources de connaissances.

Byström dans (Byström, 2002) a montré que la complexité de la tâche de RI influence largement le choix de l'information et de la source de cette dernière. L'auteur déclare que la complexité de la tâche est une caractéristique qui influence le comportement des données de l'utilisateur : plus la tâche de recherche est complexe plus le nombre de sources augmente. Un autre point à retenir est que, quand la tâche est complexe, l'utilisateur va davantage tendre vers des sources humaines comme les collègues et les experts. Byström précise aussi que la complexité de la tâche en elle-même dépend de l'utilisateur, de sa base de connaissances et de son niveau d'expertise de la tâche à accomplir.

D'autres travaux s'inscrivent dans la continuité et séparent la complexité subjective et objective de la tâche. Gwizdka et Lopatovska (Gwizdka & Lopatovska, 2009) déclarent que la perception par l'utilisateur de la difficulté de la tâche dépend des facteurs individuels comme l'expérience du chercheur, la connaissance du domaine, la capacité verbale, ainsi que d'autres capacités cognitives et aussi la motivation pour effectuer la tâche. Les individus qui ont un savoir basique percevront des tâches simples comme étant difficiles, tandis que les utilisateurs qui ont des capacités développées percevront des tâches complexes comme faciles.

Par ailleurs l'utilisateur, un autre élément important de la RI, est étudié sous plusieurs angles dans le processus de RI. Tous les facteurs contextuels peuvent être intéressants et utiles à l'optimisation de l'accès et la RI. Les effets du niveau de connaissance du domaine sur le processus de RI sont étudiés par Zhang et ses collaborateurs. Ils ont un effet sur le comportement de la recherche et sur l'efficacité de celle-ci (Zhang et al, 2005). D'autres travaux s'intéressent aux traits de personnalité des utilisateurs (Halder, Roy, & Chakraborty, 2010; Hyldegård, 2009) en quête d'informations ainsi qu'à leurs capacités à contrôler leurs émotions (Kim, 2008) pendant le processus de RI. Les auteurs montrent l'importance de ces facteurs émotionnels dans le processus de RI.

IV.3.3. Niveau interactionnel de la RI

Le contexte a été examiné aussi en termes d'espace interactif lui-même. D'un point de vue contextuel, le niveau interactionnel de RI explore les interactions des utilisateurs avec le système dans des sessions de recherche.

L'importance de ce niveau dans la contextualisation de la RI est abordée par Belkin en 1993 d'un point de vue fondamental (Belkin, 1993). Le rôle primordial de l'interaction fut repris quelques années plus tard dans une tentative de modélisation d'un système de RI

intelligent. Belkin schématise un épisode de RI dont la composante la plus importante est l'interaction, celle-ci est en relation bilatérale avec l'utilisateur (buts, tâches, connaissance, etc.) et l'information (type, mode, niveau, etc.). Des processus, comme par exemple la visualisation et la navigation, sont intégrés à ce modèle (Belkin, 1996).

Pour comprendre les interactions de l'utilisateur, des études se sont appuyées sur des résultats obtenus en psychologie cognitive. C'est le cas de (Anand & Mobasher, 2007), qui présente une approche qui intègre le contexte de l'utilisateur dans le processus informationnel. La dimension du contexte qui se focalise sur la modélisation de l'utilisateur est faite à l'image de la mémoire humaine (les travaux de psychologie). Ainsi, les modèles de préférences des interactions antérieures des usagers sont sauvegardés dans un emplacement mémoire spécifique qu'ils appellent « mémoire à long terme ». Quant aux modèles de préférences des usagers courants sont sauvegardés dans un autre emplacement mémoire appelé « mémoire à court terme ». Les traces contextuelles générées par les données sauvegardées dans la mémoire à court terme sont utilisés pour chercher des informations pertinentes pour l'utilisateur à partir de « sa » mémoire à long terme. Les auteurs ne considèrent pas le contexte comme un ensemble fixe d'attributs, mais pour eux, l'activité façonne le contexte (cf. la vue interactionnelle de (Dourish, 2004) chapitre I.1 section III), i.e. le comportement de l'utilisateur est observé et celui-ci est produit selon son contexte spécifique. Les auteurs proposent que le contexte puisse être modélisé comme un processus stochastique inclus dans un des états définis par l'ensemble représentant tous les contextes distincts dans lesquels un utilisateur interagit avec le système.

Précédemment, Xie (Xie, 2000) s'est aussi intéressé aux interactions des utilisateurs dans le cadre de la recherche documentaire. L'auteur a fait une enquête sur quarante cas d'utilisations de bibliothèques classées en quatre types : une bibliothèque universitaire de sciences sociales et humaines, une bibliothèque universitaire de science et médecine, une bibliothèque municipale et une bibliothèque particulière. L'intérêt de cette enquête est le décalage qu'il peut y avoir entre le but de l'intention interactive de l'utilisateur et le but de la stratégie qu'il adopte pour sa RI. L'analyse qualitative et quantitative des données souligne quatre types de décalages d'intentions interactives et trois types de stratégies de RI. Les résultats de l'étude sont discutés pour comprendre la nature du processus RI interactif, et pour suggérer par la suite leurs implications pour la conception des Systèmes de RI adaptatifs (H. I. Xie, 2000). Plus encore, l'auteur montre dans un travail postérieur que les études de

l'interaction des utilisateurs avec les systèmes de RI est très importante dans l'amélioration du processus informationnel des utilisateurs et en particulier dans leurs environnements de travail (Xie, 2006).

D'autres travaux se sont concentrés sur des facteurs contextuels spécifiques et leurs impacts sur les interactions avec les systèmes de RI et ainsi sur l'efficacité du processus informationnel. À titre d'exemple, nous citons l'étude de Vilar et Zumer (Vilar & Žumer, 2008) qui montrent que même la discipline académique des utilisateurs a un effet considérable sur leurs interactions avec les interfaces des systèmes de RI. Dans la même direction, une étude présentée par (Tenopir, Wang, Zhang, Simmons, & Pollard, 2008) explore le facteur contextuel « état émotionnel » (heureux, triste, stressé, etc.) et les relations qu'il peut avoir avec les comportements cognitifs des utilisateurs lors de l'interaction avec les systèmes de RI.

Nous allons exposer dans ce qui suit le quatrième et dernier niveau des travaux sur la contextualisation de la RI : le niveau de la requête.

IV.3.4. Niveau de la requête

Ce niveau du contexte s'intéresse essentiellement aux aspects linguistiques des requêtes d'utilisateurs. En effet, l'objectif des travaux qui s'inscrivent dans ce niveau est de rendre les systèmes de RI plus performants en proposant des approches de désambiguïsation de requêtes afin que les termes soient interprétés correctement par le système selon le contexte assigné. Un problème répandu dans la RI classique (cf. section III.1) est la restitution de documents contenant pourtant les termes de la requête utilisateur, mais interprétés selon un contexte linguistique erroné. Ce problème est de plus en plus présent dans l'environnement Web, où les requêtes utilisateurs sont typiquement courtes et les modèles utilisateurs sont insuffisants pour les désambiguïser (Cool & Spink, 2002).

La reformulation de requête a été et est toujours largement étudiée dans la littérature. En effet, la reformulation peut engendrer un enrichissement des requêtes, et cela en se basant sur une multitude de facteurs contextuels comme l'utilisation des termes qui se rapportent à l'utilisateur (son profil, ses préférences, etc.). La formulation peut également à l'inverse entraîner un allègement de la requête initiale pour n'en garder que les mots les plus discriminants et éliminer ceux dont la présence risque d'altérer le résultat de la recherche. Nous avons néanmoins observé des travaux qui utilisent la reformulation de requête sans utiliser le contexte de la recherche dans leur approche. Nous pouvons citer à titre d'exemple le

travail de Kumaran et Allan (Kumaran & Allan, 2008). Dans ces travaux, les auteurs se basent sur des outils de traitement du langage naturel et sur les statistiques. Ainsi, pour chacune des possibilités de sous-ensembles de termes, ils calculent un score en se basant sur la cooccurrence des termes. Le sous-ensemble sélectionné est celui qui obtient le meilleur score.

Concernant les dimensions contextuelles utilisées pour les reformulations des requêtes, les travaux sont très variés. Beaucoup d'entre eux utilisent la dimension contextuelle utilisateur pour l'expansion des requêtes. Parmi ces travaux, nous citons (Kostadinov, Bouzeghoub, & Lopes, 2007). Les auteurs proposent une nouvelle approche de reformulation des requêtes alternant des étapes d'enrichissement et de réécriture afin de mieux tenir compte du profil utilisateur. Elle se compose de quatre étapes : expansion de la requête, identification des sources pertinentes, combinaison des sources pertinentes et enrichissement. Une autre méthode intéressante qui utilise le profil usager pour la reformulation des requêtes est celle de (Koutrika & Ioannidis, 2005). Les auteurs définissent le profil des usagers comme une liste de prédicats disjonctifs. Les prédicats sont associés à des poids compris entre 0 et 1 qui représentent leur importance relative par rapport aux préférences de l'utilisateur. Ainsi, leur processus d'enrichissement de requête consiste à reformuler la requête initiale de l'utilisateur en ajoutant des prédicats de ce profil.

D'autres travaux encore se sont tournés vers d'autres facteurs contextuels pour la reformulation des requêtes de RI. En effet, Bai et Nie (Bai & Nie, 2008) proposent une approche de modélisation du langage afin d'intégrer un certain nombre de facteurs contextuels : le domaine de la requête, les caractéristiques de la collection des documents, le contexte des termes de la requête. Ce dernier facteur est qualifié de « *contexte interne à la requête* », tandis que les deux autres constituent le « *contexte externe à la requête* ». Stojanovic, quant à lui, a proposé une approche complète d'amélioration de requête qui permet de cerner la nature du besoin d'un utilisateur en ayant un processus progressif, étape par étape, tout en prenant en compte sa tâche et son espace de connaissance. Cette approche d'enrichissement de la requête se base sur trois dimensions : la **dimension cognitive**, la **dimension affective** et enfin la **dimension situationnelle de l'utilisateur** (Stojanovic, 2005). Enfin, d'autres travaux plus récents proposent des méthodes de reformulation des requêtes en intégrant la dimension temporelle afin d'améliorer l'efficacité de la RI (Amodeo, Amati, & Gambosi, 2011). Ils utilisent le facteur contextuel lié au temps des documents (date de

publication, date de création, etc.) et des requêtes dans leur méthode de reformulation des requêtes.

IV.4. Synthèse

Les utilisations du contexte varient beaucoup selon le domaine et l'orientation des systèmes. En ce qui concerne le contexte dans le domaine informationnel, les facteurs qui peuvent être pris en compte varient d'une étude à l'autre et d'un système à l'autre. En effet, la multitude des approches que nous venons de présenter et les facteurs contextuels variés pris en compte dans celles-ci rend impossible une définition unique du contexte dans ce domaine. Pour cela, les travaux se concentrent généralement sur des aspects particuliers du contexte pour les intégrer aux systèmes de RI. En 2005, lors du workshop IRiX (Information Retrieval in conteXt) les chercheurs ont spécifié les quatre dimensions principales du contexte pouvant être retenues : **les granules d'information, la tâche, l'utilisateur, le système**. Nous remarquons que le contexte dans notre travail contient également les deux dimensions du contexte : tâche et utilisateur.

Cependant, ce que nous avons remarqué dans les études relativement récentes dans ce domaine, est que la tâche prend une place primordiale dans la littérature en tant que facteur contextuel le plus important (Kelly, 2006). De plus, Belkin (Belkin, 2008) a présenté les challenges actuels pour la RI comme devant être de plus en plus centrés autour de l'utilisateur. Pour lui, toutes les perspectives doivent tendre vers **l'utilisateur dans sa tâche de RI pour accomplir son but**. Les axes importants selon Belkin sont : en premier lieu la détermination de la tâche qui a poussé l'utilisateur à entamer sa recherche, le comportement de la recherche, le contexte, les émotions des utilisateurs, etc. Belkin, insiste sur le fait d'intégrer les systèmes d'accès ou de RI dans l'environnement même de la tâche de l'utilisateur.

Partant de ce constat, nous avons défini **les trois dimensions contextuelles les plus importantes à intégrer dans le processus informationnel**. En effet, les deux premières dimensions sont indiscutablement présentes dans la majorité des approches récentes de contextualisation de la RI. Ces deux dimensions sont **l'utilisateur** et **la tâche**. Concernant la tâche, nous incluons « les tâches de travail » et « les tâches informationnelles » qui sont liées. Enfin, la troisième dimension contextuelle qui va être présentée dans la section suivante et qui regroupe tous les facteurs contextuels n'appartenant pas aux deux premières dimensions est **l'environnement**.

V. Les dimensions pertinentes du contexte dans l'accès à l'information

V.1. La tâche

Les utilisateurs éprouvent toujours de grandes difficultés dans les activités de RI décisives, tout particulièrement dans des situations problématiques. Cette difficulté est encore plus grande dans un contexte métier, c'est-à-dire lorsque l'utilisateur doit retrouver et lire un document pour réaliser une tâche métier. Les travaux de (Hirsh, 2000) montrent dans une évaluation que les personnes interrogées déclarent passer approximativement 68% de leur temps de travail hebdomadaire sur des activités liées à la documentation (documents scientifique et technique). La tâche informationnelle est donc toujours liée à sa tâche de travail qui la plupart du temps motive l'accès à l'information. Li (Li, 2009) propose une étude empirique sur la relation entre la tâche de travail et la tâche informationnelle, ainsi qu'entre les différentes facettes entre les tâches informationnelles et la tâche de travail. Pour cela, il se base sur les travaux antérieurs de (Li & Belkin, 2008) pour décrire les facettes de la tâche, et sur une enquête qu'il a menée sur un public entièrement issu de la communauté universitaire (étudiants, enseignants, etc.). Cette enquête a permis de calculer analytiquement les différentes relations entre les facettes des tâches de travail et les tâches informationnelles qui lui sont associées.

La tâche métier peut être élémentaire ou décomposable en plusieurs sous-tâches. Sur la figure II.3, nous illustrons un exemple de tâche métier à deux niveaux. La méthode de modélisation utilisée pour cet exemple est la « *Hierarchical task analysis* » (Annett, 2003). Pour être réalisée, une tâche métier nécessite plusieurs éléments qui sont modélisés dans cette composante. Un catalogue des tâches/buts des usagers peut être modélisé sous une forme hiérarchique. Les tâches sont décomposées en sous-tâches généralement moins complexes que la tâche principale et dont l'objectif est nécessaire par la suite pour atteindre l'objectif de la tâche principale (tâche mère). Nous donnons une formalisation de l'objectif d'une tâche métier principale noté O_{Tp} , composition (symbole « \circ ») de l'ensemble des objectifs des tâches nécessaires à sa réalisation :

$$O_{Tp} = O_{T(n,i)} \circ O_{T(n-1,i)} \dots \circ O_{T(1,i)}.$$

Nous précisons par exemple que $O_{T(n,i)}$ est l'objectif d'une tâche secondaire numéro i du niveau n . Ce dernier correspondant au nombre maximum de niveaux qui n'a pas un seuil maximum prédéfini.

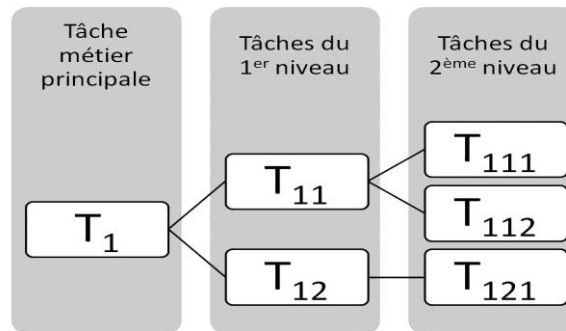


Figure II.3. Exemple de hiérarchisation d'une tâche métier

V.1.1. La tâche comme motivation de la RI

La tâche de travail comme motivation de RI a été définie selon plusieurs perspectives. Ingwersen (Ingwersen & Järvelin, 2005b), la définit comme un problème sous-jacent du travail courant d'une personne en se plaçant d'un point de vue cognitif. Cette définition a été étendue plus tard pour inclure tous les travaux journaliers des personnes. Pour Byström et Hansen (Byström & Hansen, 2005), les tâches de travail sont vues comme des parties séparées des devoirs d'une personne envers son employeur. La tâche de travail est considérée aussi comme un élément important du contexte (Belkin, 2008; Ingwersen & Järvelin, 2005b; Kelly, 2006; Spink & Cole, 2001). Pour Byström et Hansen (Byström & Hansen, 2005), les tâches qui nécessitent un besoin en information sont fortement liées à des facteurs situationnels et contextuels. Beaucoup de travaux dans les domaines de la documentation et de la RI s'appuient sur la modélisation de la tâche afin de mieux personnaliser l'accès au document pertinent pour l'utilisateur. Une meilleure prise en compte du modèle de la tâche de RI est de nature à augmenter l'efficacité des systèmes de RI (Stojanovic, 2005).

V.1.2. Tâches informationnelles

Définition

Pendant sa quête d'information, l'utilisateur peut interroger différentes sources d'informations. La variation de ces sources (quantité, qualité, etc.) a fait l'objet de plusieurs travaux. Son importance dans le processus de la tâche informationnelle n'est pas négligeable. Pour cela la définition de ces quelques notions est primordiale à la compréhension des différentes approches de la littérature. Nous allons utiliser les définitions de Byström et Hansen (Byström & Hansen, 2005) parce qu'elles sont les plus génériques et par ailleurs

parce que les autres approches sont souvent une spécialisation des définitions de ces deux auteurs.

Plusieurs travaux ont étudié la tâche informationnelle pour comprendre et améliorer le comportement des utilisateurs. Des travaux divergent sur la décomposition même de la tâche informationnelle. Avant de parler de chaque approche nous définissons les principaux termes utilisés par la suite :

- *information seeking* : la recherche et l'accès à l'information en se basant sur toutes sortes de sources possibles et disponibles est appelé *information seeking*. Notamment en utilisant l'outil informatique ou d'autres ressources : un livre, un article, demander à un collègue, etc.
- *information searching* : la recherche et l'accès à l'information en se basant sur l'outil informatique est appelé *information searching*, i.e. interaction avec un système d'information.
- *information retrieval* : la recherche et l'accès à l'information en se basant sur les systèmes spécifiques de RI (comme des moteurs de recherche) est appelé *retrieval*.

Lien entre tâche informationnelle et tâche de travail

Parmi les éléments contextuels composant la dimension tâche, il y a les connaissances informationnelles requises pour la réalisation de la tâche. Elles représentent les informations sans lesquelles l'utilisateur ne peut effectuer sa tâche et qui seront appelées « connaissance de la tâche », noté « C_T » connaissance d'une tâche T . Du point de vue des connaissances, l'objectif informationnel d'une tâche, noté « $O_{\text{informationnel}}$ » peut donc être vu comme l'union des ensembles des connaissances nécessaires pour chaque sous-tâche :

$$O_{T, \text{informationnel}} = C_{T(n,i)} \cup C_{T(n-1,i)} \cup \dots \cup C_I$$

Cette décomposition des objectifs de la tâche métier en objectif informationnel à son tour décomposable en un ensemble de connaissances des sous-tâches, nous permet par la suite de cibler le manque informationnel de l'utilisateur effectuant sa tâche métier.

Pour mieux comprendre la relation entre les tâches métier et les processus informationnels, nous montrons dans la figure II.4 un exemple de liaison possible entre une tâche métier, nécessitant de l'information, et deux processus informationnels permettant de

fournir les données manquantes nécessaires à son traitement. Comme nous pouvons le voir sur cette figure, le processus informationnel n'est pas unique pour une tâche donnée. Cependant, même si leur composition de tâches informationnelles diffère, tous ces processus ont le même objectif informationnel nécessaire pour accomplir la tâche T_{11} .

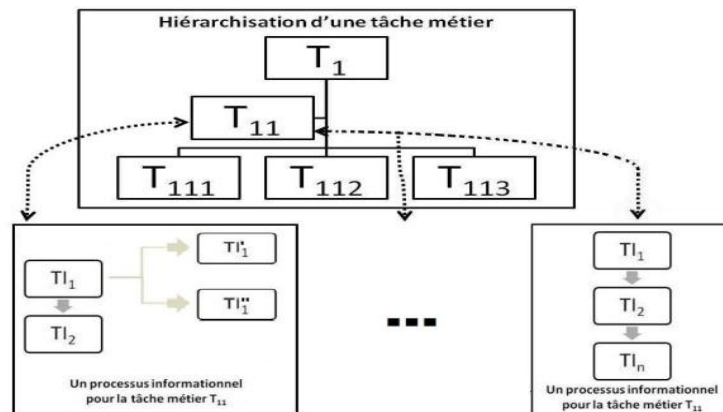


Figure II.4. Dépendance entre la tâche métier et les processus informationnels

De nombreux travaux se sont intéressés aux caractéristiques de la tâche d'accès ou de recherche d'information pour la rendre plus efficace. Selon Tricot et Nanard, « *la description d'une tâche de recherche d'information devrait prendre en compte la représentation mentale du but d'information qui a une composante conceptuelle et procédurale, voire informationnelle. Elle dépend de la crédibilité des sources et elle est donc influencée par les connaissances de l'individu et son besoin en information. Cette description de la tâche devrait aussi prendre en compte l'implémentation du but, c'est-à-dire l'adresse et le nombre de cibles dans le système, les procédures à utiliser pour atteindre ces cibles, la structure générale du système et de l'interface. Enfin, elle devrait décrire le contexte de l'activité et la raison pour laquelle un individu choisit d'utiliser un tel système pour chercher telle information dans le cadre de la réalisation d'une tâche principale* » (Tricot & Nanard, 1998).

Un aspect important afin de caractériser les buts des chercheurs de l'information, des contextes et des problèmes d'accès à l'information est de considérer les tâches qui les ont menés à s'engager dans une quête d'information et les tâches qu'ils doivent accomplir dans les processus d'information *seeking* et l'information *searching*. Nous avons remarqué une différence considérable entre les approches des différents travaux qui se concentrent sur la tâche informationnelle comme un moyen d'améliorer le processus de recherche et d'accès à l'information. Cette divergence réside dans la structure conceptuelle des tâches

informationnelles. Les auteurs (Byström, 2002; Byström & Hansen, 2005; Ingwersen & Järvelin, 2005b) classent les tâches en ensembles imbriqués.

Pour Byström et Hansen (Byström & Hansen, 2005) les tâches *seeking* et *searching* sont les tâches secondaires des tâches de travail. Les tâches informationnelles *searching* sont des tâches secondaires des tâches informationnelles *seeking*. Les tâches informationnelles *searching* englobent aussi les tâches informationnelles *retrieval*.

Ingwersen et Järvelin (Ingwersen & Järvelin, 2005b) classent les tâches informationnelles en deux dimensions imbriquées. *Seeking tasks* et *retrieval tasks* dépendent du système interrogé durant l'accès à l'information. Autrement dit, les *retrieval tasks* sont associées avec les systèmes de RI et les *seeking tasks* sont associées à tous les autres systèmes, par exemple les e-mails ou les réseaux sociaux. Ingwersen et Järvelin partagent la même approche que Byström et Hansen en définissant la tâche informationnelle comme un sous ensemble de la tâche de travail.

V.1.3. Principales caractéristiques des tâches

Plusieurs travaux ont étudié des caractéristiques spécifiques des tâches pour améliorer le processus informationnel. Ainsi, deux caractéristiques principales ont suscité beaucoup d'intérêt par la communauté de recherche :

- les *types des tâches* ;
- les *étapes des tâches*.

Concernant le **type de la tâche**, de nombreuses études ont proposé des classifications de la tâche informationnelle selon le type de tâche de travail qui l'a déclenché (Freund, 2008; Li & Belkin, 2008). Par exemple, Li et Belkin (Li & Belkin, 2008) ont proposé une approche à facettes pour la conceptualisation de tâche afin d'explorer les rapports entre les tâches et le comportement interactif de RI. Ils ont démontré que les comportements d'accès à l'information et les résultats de recherche désirés sont différents selon les types de tâches à chacun de ces trois niveaux, c'est-à-dire les tâches de travail, les tâches informationnelles *seeking* et les tâches informationnelles *searching*. Li et Belkin (Li & Belkin, 2008) montrent que les tâches partagent des aspects communs selon leur type. Ces aspects communs de tâche sont appelés « des facettes ». Ils ont examiné les modèles des tâches existants dans différents domaines et proposent un nouveau modèle à facettes qui aspire à faciliter la recherche aussi

bien dans les communautés de science de l'information, ou de recherche d'information. Li et Belkin se sont basés sur une étude complète de la littérature, et montrent que des modèles existants ont eu tendance à ne s'intéresser qu'à des aspects partiels des tâches et qu'une approche à facettes basée sur le type des tâches peut fournir un modèle plus holistique. Le modèle qu'ils proposent offre huit facettes principales : la source de la tâche, l'utilisateur, le temps, le produit, le processus, le but, des caractéristiques de la tâche et la perception de l'utilisateur de la tâche. Ces facettes sont regroupées en deux classes principales :

- les *facettes génériques* de la tâche décrivent les caractéristiques externes à celle-ci (exemple : temps d'exécution, but, source de la tâche, etc.);
- les *attributs communs de la tâche* (exemple : type, complexité, etc.).

D'un autre côté, les **étapes de la tâche** de recherche ont elles aussi reçu une attention importante. En effet, plusieurs études se sont penchées sur l'influence des étapes de la tâche sur le comportement de recherche des utilisateurs et leurs performances (J. Liu & Belkin, 2010a). Des travaux ont montré que les critères de pertinence des usagers dépendent de l'étape du processus d'exécution de la tâche. Ainsi, Taylor et ses collaborateurs (Taylor, Cool, Belkin, & Amadio, 2007) ont trouvé une corrélation statistiquement significative entre les critères de pertinence des usagers et leurs choix informationnels pendant les différentes étapes de la tâche.

Les études présentées plus haut ont montré la prise en compte de plusieurs caractéristiques de la tâche dans le processus informationnel. Parmi elles, nous avons présenté les **étapes** de la tâche sur lesquelles la communauté scientifique porte beaucoup d'intérêt. En effet, elle s'efforce d'améliorer le processus d'accès à l'information notamment quand il est lié à des tâches métier. Cependant, pour distinguer les étapes des tâches, les chercheurs doivent au préalable les modéliser en utilisant des formalismes de modélisation de tâches. Nous en avons étudié plusieurs que nous exposons dans la section suivante.

V.1.4. Modélisation des tâches

Introduction

L'analyse traditionnelle des tâches est aujourd'hui suffisamment maîtrisée (Diaper & Stanton, 2004). Plusieurs formalismes de modélisation des tâches sont disponibles dans la littérature. Un modèle de tâche est souvent défini comme une description d'une tâche interactive exécutée par l'utilisateur d'une application à travers l'interface utilisateur de

l'application. Les différents éléments dans un modèle de tâche représentent les actions spécifiques que l'utilisateur peut entreprendre. L'information sur le déroulement de tâches secondaires aussi bien que sur les conditions d'exécution de la tâche principale est également incluse dans le modèle.

Les méthodes d'analyse des tâches ont été élaborées dans plusieurs disciplines ayant chacune des centres d'intérêt spécifiques. Par exemple dans le domaine de la psychologie cognitive, les modèles de tâche sont utilisés pour mieux comprendre l'interaction d'un usager avec un système interactif pour accomplir une tâche. Dans d'autres domaines, des modèles de tâche sont employés pour évaluer la charge de travail, prévoir et assigner des tâches aux utilisateurs dans une organisation particulière, etc.

Les formalismes de modélisation des tâches présentent de nombreuses différences, que ce soit sur le fond ou sur la forme. Ces différences touchent aussi l'objectif final de chacun des modèles de tâche. Ces objectifs peuvent être :

- informer des concepteurs au sujet des problèmes potentiels de rentabilité, comme dans *Hierarchical task analysis* (HTA).
- évaluer l'activité humaine, comme dans *Goal operators methods selectors* (GOMS).
- soutenir la conception en fournissant un modèle de tâche détaillée qui décrit la hiérarchie des tâches, les objets utilisés et la structure de connaissance, comme dans *ConcurrentTaskTree* (CTT) ou dans *Task knowledge structure* (TKS).

Ces différents objectifs introduisent des concepts intéressants. Par exemple, les méthodes d'analyse des tâches employées dans l'analyse cognitive valorisent les objectifs, pour arriver à analyser la charge de travail, le temps d'exécution, ou la connaissance cognitive exigée pour effectuer un ensemble de tâches. À cet égard, elles sont semblables à la modélisation de l'utilisateur. D'autre part, les méthodes prévues pour soutenir le travail coopératif ont développé des formalismes pour représenter la manière dont les tâches sont assignées à différents rôles, élargissant la portée de l'analyse des tâches avec des concepts d'organisation. Mais cette diversité présente cependant des points faibles comme : le manque de compréhension du contenu de chacun des modèles de tâche, y compris le raisonnement derrière la méthode d'analyse, les concepts, leurs rapports, leurs vocabulaires, et les opérations cognitives impliquées. L'hétérogénéité de contenu de différents modèles est due à la variété des méthodes utilisées par les différentes disciplines en plus du manque d'interopérabilité des

systèmes. Les outils de modélisation des tâches ne partagent pas un format commun, ils sont limités à ces modèles de tâche exprimés selon leurs formats prédéfinis.

Quelques formalismes de modélisation des tâches

a) Hierarchical task analysis

C'est une méthode pionnière d'analyse des tâches (Annett, 2003; Annett & Duncan, 1967). Elle vise principalement la formation des utilisateurs afin d'effectuer des tâches particulières. Sur la base d'entrevues, d'observations d'utilisateurs, et d'analyse des documents existants (par exemple, manuels, documentation), HTA décrit des tâches en termes de trois concepts principaux : tâches, hiérarchie de tâche et plans.

Des *tâches* sont périodiquement décomposées en tâches secondaires à un point où les tâches secondaires sont assignées à l'utilisateur ou à l'interface utilisateurs, devenant de ce fait observables.

La *hiérarchie de tâches* représente statiquement cette décomposition des tâches. Le critère d'arrêt de la décomposition est un principe de base qui se rapporte à la règle suivante : le coût d'une exécution non satisfaisante \times la probabilité d'une exécution non satisfaisante. Par le coût d'une exécution non satisfaisante les auteurs entendent les conséquences qu'elle pourrait produire sur l'exécution de la tâche.

Bien que le *plan* soit une description informelle des rapports temporels entre les tâches, il est l'un des dispositifs les plus attrayants de HTA, car il est simple et expressif. Les plans sont très proches de la description textuelle ou de la liste d'activités d'analyse des tâches traditionnelles. Puisque la hiérarchie de tâche ne contient aucune tâche donnant une action, le plan spécifie les actions à effectuer et cela précisément pour chacune des tâches secondaires d'une tâche principale donnée. Le plan agit comme une contrainte sur l'exécution de la tâche. Par ailleurs, un plan est donné pour chaque niveau hiérarchique.

N'importe quelle tâche peut être exprimée en termes d'objectifs, dits atteints quand la tâche correspondante est accomplie. Chaque objectif a un statut (i.e., latent ou actif) et des conditions à satisfaire. L'avantage ici dans HTA est que les objectifs sont indépendants des moyens concrets mis en œuvre pour les atteindre. Par conséquent, pour chaque objectif à n'importe quel niveau de décomposition plusieurs opérations différentes peuvent être imaginées et spécifiées pour l'atteindre. Chaque opération est par conséquent liée à un

objectif (ou à plusieurs objectifs) et elle est spécifiée par les circonstances dans lesquelles l'objectif est activé (l'entrée), les activités (actions) qui contribuent à l'atteinte de l'objectif, et les conditions indiquant que l'objectif a été atteint (rétroaction).

b) Goal operators methods selectors (GOMS)

GOMS est un modèle d'ingénierie développé par (Card, Moran, & Newell, 1983). Ce modèle permet les prédictions quantitatives des performances des usagers en intégrant des tableaux de valeurs de paramètres qui reposent sur une architecture cognitive. Dans GOMS le concept de méthode est crucial. Celle-ci est une séquence d'opérateurs pour décrire la performance de la tâche. Par ailleurs, les tâches sont déclenchées par les objectifs et sont décomposées en sous-tâches.

GOMS décrit non seulement la structure de l'objectif et les actions élémentaires nécessaires à la réalisation de la tâche, mais aussi les procédures ou « méthodes » de réalisation de l'objectif et les règles de sélection de ces méthodes. Le modèle prédit aussi le temps d'exécution de chaque action.

Ce formalisme censé prendre en compte l'opérateur a été abondamment critiqué : il montre l'efficacité maximale de l'opérateur sans être capables de rendre compte d'erreurs. On lui reproche, en quelque sorte, d'être "inhumains". Le modèle suppose que l'utilisateur ne fait jamais d'erreur et qu'il sait à n'importe quel moment ce qu'il faut exactement faire.

c) Méthode analytique de description des tâches (MAD)

Cette méthode permet de décrire des tâches et de les comparer entre elles. Chaque tâche est représentée par plusieurs caractéristiques : l'identification, les éléments et les attributs de la tâche. Pour identifier la tâche, on lui donne un numéro et un nom (Scapin & Pierret-Golbreich, 1990). Les éléments de la tâche sont :

- le but : ce qui est recherché dans l'exécution de la tâche ;
- l'état initial : liste d'objets permettant de représenter, au temps t_0 , une partie du monde dans laquelle la tâche va s'effectuer ;
- les pré-conditions : ensemble de prédicats exprimant des contraintes sur les objets de l'état initial ;

- le corps de la tâche : expression de la façon dont la tâche est exécutée. Il peut être soit une action élémentaire, soit une structure de sous-tâches. Cette dernière est définie par un constructeur et un ensemble de sous-tâches. Elle peut être séquentielle, alternative, parallèle ou simultanée ;
- les post-conditions : contraintes portant sur les objets de l'état final ;
- l'état final : liste des objets permettant de représenter la partie du monde qui a été modifiée par la tâche ;
- les attributs de la tâche : caractéristiques particulières de certaines sous-tâches (facultatifs, itératifs, prioritaires).

Le formalisme MAD propose de décrire formellement la description du contexte de la tâche avant et après exécution. Que ce soit lorsque les objets contextuels influencent ou sont modifiés par la tâche. Par ailleurs, on reproche au formalisme MAD sa focalisation centrée sur l'activité individuelle de l'utilisateur, son incapacité à formaliser certaines caractéristiques des tâches et son manque d'outils informatiques adaptés.

d) Groupware task analysis (GTA)

Le formalisme GTA fut développé comme solution pour modéliser la complexité des tâches dans des environnements coopératifs. En effet, GTA est appliqué dans la conception des systèmes coopératifs et se base sur les activités. Par ailleurs, GTA distingue clairement les tâches des actions (Van Der Veer, Lenting, & Bergevoet, 1996). Ainsi, pour extraire des données, GTA emploie des techniques telles que des entrevues, l'analyse de l'interaction, et l'observation structurée. La modélisation de tâche est alors exécutée comme activité cyclique où des modèles sont créés, évalués, modifiés. GTA utilise principalement une représentation hiérarchique.

Un dispositif attrayant de GTA réside dans ses possibilités de représenter des tâches coopératives. La représentation se fait en intégrant le concept de rôle dans le modèle global de la tâche et en permettant la représentation des ensembles de tâches pour lesquelles un rôle est responsable des aspects organisationnels, tels que la manière dont un rôle est attribué aux divers agents.

Bien que l'ontologie, ajoutée par les auteurs et décrit les concepts et les relations entre méthodes, améliore la conceptualisation du monde de la tâche. Mais la représentation n'est

pas basée sur un formalisme satisfaisant. En effet, les buts et les actions sont représentés en tant qu'attributs de la tâche et non comme concepts. Ceci est quelque peu contradictoire avec le fait que GTA donne la possibilité aux buts d'être atteints de plusieurs manières.

e) Diane+

Diane+ modélise une tâche en se basant sur deux aspects importants (Tarby & Barthet, 1996) :

- les procédures décrivent seulement les caractéristiques propres à une application et incluent les actions standards que partagent toutes les applications. Ceci présume que les actions supposées standard, précédemment définies, s'appliquent vraiment à l'application d'intérêt. Si une action standard ne s'applique pas, ceci serait indiqué ;
- les procédures décrites ne sont pas obligatoires ; ce qui n'est pas interdit est donc par définition autorisé.

Nous notons que Diane+ peut représenter toutes les contraintes des caractéristiques ci-dessus. Toutes les structures algorithmiques existent dans Diane+, telles que séquence ordonnée, séquence non ordonnée, boucle, choix exigé, libre choix, parallélisme, opérations par défaut, et ainsi de suite.

f) Task knowledge structure (TKS)

TKS (H. Johnson & Johnson, 1991) est une représentation conceptuelle de la connaissance qu'une personne garde en mémoire pour effectuer une tâche particulière. Un TKS est associé à chaque tâche qu'un agent (utilisateur) effectue. Le rôle que l'agent est présumé assumer détermine les tâches dont il est responsable. Un agent peut prendre plusieurs rôles, et un rôle peut être pris par plusieurs agents. Même si les tâches ou la TKS peuvent sembler similaires à travers différents rôles (par exemple, écrire une lettre pour une secrétaire et pour un manager), elles seront considérées différentes. Un rapport de « similitude » est visé pour représenter cette situation.

La TKS d'une tâche détient des informations sur l'objectif de la tâche, qui est la finalité que la tâche est censée produire. Un objectif particulier est accompli par une tâche particulière. Un objectif est décomposé en sous-structures d'objectif, qui contiennent tous les sous-objectifs intermédiaires requis pour le réaliser. Chaque sous-objectif dans la structure des objectifs a une sous-tâche correspondante dans la structure de tâche, et vice versa.

La méthode TKS propose un système de règles de production afin de choisir les procédures appropriées pour chaque contexte. Des actions sont directement liées à un arbre de tâche et constituent les feuilles. Les actions et les objets ont des propriétés d'intérêt, qui peuvent être centrales à l'exécution d'une tâche et avoir des exemples typiques. Par exemple, la centralité d'un objet ou d'une action est toujours exprimée en ce qui concerne une tâche ou un procédé qui met en service l'objet ou l'action. Nous notons que les objets sont structurés dans une hiérarchie de décomposition.

g) ConcurrentTaskTree (CTT)

CTT (Paterno, 2000) est basé sur cinq concepts : tâches, objets, actions, opérateurs, et rôles. Des constructeurs de CTT, nommés opérateurs, sont utilisés pour lier les tâches filles de mêmes parents au même niveau de décomposition. À cet égard, CTT diffère des modèles précédemment décrits où les opérateurs agissent sur des rapports parents-enfants. Il est également important de noter que CTT possède une définition formelle pour ses opérateurs temporels. En outre, CTT fournit les moyens de décrire des tâches coopératives. Pour décrire une telle tâche, le modèle de tâche se compose de différents arbres de tâches, un pour la partie coopérative et un pour chaque rôle qui est impliqué dans la tâche. Des tâches sont encore décomposées jusqu'au niveau des tâches de base, qui sont définies en tant que tâches ne pouvant plus être décomposées. Les actions et les objets sont spécifiés pour chaque tâche de base. Les objets peuvent être les objets perceptibles ou les objets d'application ; ces derniers sont par ailleurs tracés sur les objets perceptibles afin d'être présentés à l'utilisateur.

Une caractéristique intéressante du CTT est que les deux actions, d'entrée et de sortie, associées à un objet sont précisées. La spécification des objets est principalement orientée vers la spécification des objets d'interaction (connecteurs). Le concept de plateforme est intégré au formalisme CTT afin de soutenir le développement d'interfaces utilisateur multiplateforme.

Cependant, CTT ne permet pas de prendre en compte certaines éléments importants, tels que : les conditions d'environnement, les circonstances qui peuvent affecter le procédé, les artefacts manipulés pendant la tâche, la charge de travail cognitive et l'état actuel des utilisateurs (i.e. stress, fatigue), ni le détail du type d'interaction de bas niveau ou le changement de point d'intérêt de l'utilisateur. Par contre, l'environnement logiciel proposé « CTTE » permet facilement l'extension de ce formalisme selon les préférences des concepteurs.

Différences entre ces formalismes

Nous avons donné ci-dessus une liste non exhaustive des formalismes de modélisation des tâches. Ces formalismes exhibent une variété de concepts et de relations qui varient principalement sur deux dimensions : sémantique et syntaxique.

a) Différences syntaxiques

Les différences syntaxiques les plus remarquables se résument dans le tableau II.1. La comparaison est basée sur quatre caractéristiques :

- l'organisation des tâches (comment une tâche est-elle liée au but à niveau élevé ?) ;
- l'opérationnalisation (point auquel la méthode fournit des manières de tracer des buts à haut niveau aux méthodes à bas niveau pour les atteindre) ;
- le niveau le plus bas de décomposition de la tâche (le nœud feuille dans la décomposition d'une tâche) ;
- le niveau opérationnel (niveau de décomposition de la tâche où les actions ont lieu).

On peut observer dans le tableau II.1 que des termes semblables ou différents peuvent être employés pour se rapporter aux mêmes concepts. Par exemple, le plan, l'opérateur, le constructeur, et le but sont employés souvent pour discuter la planification à niveau élevé de la tâche. Bien que la plupart des modèles prévoient la décomposition de tâche, structurellement et temporellement, ils ne décrivent pas nécessairement comment la décomposition peut être effectivement réalisée. Par exemple, un scénario est fréquemment considéré comme une instantiation particulière d'un modèle général de tâche qui dépend des circonstances particulières dans le contexte d'utilisation.

	HTA	GOMS	GTA	CTT	MAD	TKS	DIANE+
Organisations des tâches	Plans	Opérateurs	Constructeurs	Opérateurs	Constructeurs	Plans/constructeurs	Buts
Opérationnalisation	N. T.	Méthodes/ Règles de sélections	N. T.	Scénarios	Pré et post Conditions	Procédures	Procédures
Feuilles de l'arbre de la tâche	N. T.	Tâche basique	Tâche basique	Tâche basique	N. T.	Actions	N. T.
Niveau opérationnel	Tâches	Opérateurs	Actions/ Opérations système	Actions	Tâches	N. T.	Opérations

Tableau II.1. Caractéristiques principales des modèles de tâches

Les deux dernières lignes du tableau décrivent comment une tâche est périodiquement décomposée en tâches secondaires pour arriver aux nœuds feuilles. Quelques modèles ne font pas la distinction terminologique entre les différents niveaux de décomposition.

b) Différences sémantiques

Des différences sémantiques sont liées à la variation conceptuelle à travers les modèles. Elles peuvent être d'importance majeure ou mineure. Par exemple, les différences dans les définitions des relations sont considérées majeures ou aussi des cas où le même concept n'est pas défini de la même manière dans les différents modèles). Les différences les moins significatives sont les variations dans la manière d'exprimer les relations. Par exemple, les constructeurs dans GTA, MAD, ou TKS expriment le rapport temporel entre une tâche et ses tâches secondaires (bien que l'ensemble des constructeurs ne soit pas identique dans tous les modèles), tandis que les opérateurs dans CTT sont employés entre les tâches filles de mêmes parents. Les opérateurs utilisés dans GOMS ont une sémantique duelle : ils spécifient des actions (cognitives et motrices) effectuées par l'utilisateur. Le tableau II.2 compare les modèles de tâche sur les critères suivants :

- formalisation : cette dimension spécifie si un modèle est basé sur un système formel ou non. Par exemple, les opérateurs temporels de CTT sont définis dans l'algèbre de processus ;

- aspects cognitifs : cette dimension concerne l'incorporation et/ou l'appui des aspects cognitifs des activités ;
- portée des constructeurs : cette dimension exprime la portée des éléments de tâche sur lesquels les opérateurs temporels agissent. La portée peut être le parent ou l'enfant quand l'opérateur de contrainte temporelle affecte la commande, respectivement entre un nœud père dans la décomposition de tâche et ses enfants (comme dans HTA) ou entre les enfants de mêmes parents (uniquement dans CTT).

	HTA	GOMS	GTA	CTT	MAD	TKS	DIANE+
Formalisation	NON	NON	NON	OUI	OUI	NON	OUI
Aspects cognitifs	Problème d'utilisabilité	Performance usager	Performance usager	NON	NON	Structures de connaissance	NON
Portée des constructeurs	Parent	Niveaux multiples	parent	Sœurs	Parent	Niveaux multiples	Sœurs
Extensibilité	Non	Non	Non	Oui	Non	Non	Oui
Tâches concurrentes	Non	Non	Non	Oui	Non	Non	Non
Réutilisabilité	Non	Non	Oui	Oui	Non	Oui	Oui

Tableau II.2. Différences sémantiques entre les modèles

V.2. L'utilisateur

V.2.1. Profil usager

La dimension contextuelle usager a été qualifiée de dimension contextuelle la plus importante dans le processus de RI contextuelle permettant d'améliorer la pertinence des résultats (Park, 1994). En effet, les travaux dans le domaine de RI contextuelle étudient le « profil usager » afin de prendre en compte tous les facteurs contextuels liés aux utilisateurs.

Selon Kobsa (Kobsa, 2001, 2007), les types de propriétés de l'utilisateur et la façon dont ils sont structurés dans un modèle de l'utilisateur sont largement basés sur l'intuition et l'expérience. L'auteur fournit une liste de propriétés pertinentes de l'utilisateur pour la

création d'un modèle utilisateur. Il s'agit notamment : de données démographiques, d'informations sur le niveau des connaissances du domaine ou d'expérience, d'objectifs et de plans avec des intentions à court / long terme, les centres d'intérêt / préférences, etc.

V.2.2. Au-delà du profil usager classique

En plus du profil classique des usagers, les travaux dans la contextualisation de la RI ont intégré d'autres facteurs contextuels. Ainsi, la dimension contextuelle usager se focalise sur toutes les informations contextuelles se rapportant aux utilisateurs et ce en utilisant les techniques de modélisation de l'utilisateur. Le modèle usager se focalise sur le contexte physiologique de l'utilisateur (par exemple l'âge ou le poids corporel) et sur le contexte mental de l'utilisateur (par exemple sa motivation ou son état émotionnel). En effet, Schmidt et ses collaborateurs (Schmidt, Beigl, et al., 1999) classifient les informations contextuelles liées à l'utilisateur en trois catégories :

- la connaissance issue de ses habitudes comme les centres d'intérêt de l'utilisateur qui constituent l'un des attributs les plus importants qui est couramment modélisé dans la plupart des modèles utilisateur pour la RI. Les centres d'intérêts ont été largement appliqués très tôt dans les systèmes d'information personnalisée (Brusilovsky, 1996) ;
- ses caractéristiques mentales sont parmi les facteurs qui sont représentés par le contexte mental de l'utilisateur nous pouvons citer son identité, ses préférences, ses connaissances et ses compétences énumérées dans le modèle de contexte de Reichenbacher (Reichenbacher, 2007) ;
- et son état physiologique.

D'autres facteurs contextuels ont suscité l'attention des travaux de modélisation de la dimension contextuelle de l'utilisateur (Belkin, Muresan, & Zhang, 2004). Nous pouvons citer la familiarité de l'utilisateur avec le domaine relatif à sa recherche (J. Liu & Belkin, 2010b), l'expérience de l'utilisateur dans l'utilisation du ou des systèmes de RI, le type des documents recherchés (Freund, 2008; Freund & Toms, 2005), les activités de l'utilisateur pendant le processus de recherche. Des facteurs plus récents font leur apparition dans la dimension du contexte usager. Ces facteurs se focalisent sur le facteur contextuel « *état émotionnel* » dans la RI (Arapakis, Jose, & Gray, 2008) tels que la frustration, le stress, le bonheur, etc.

V.3. Environnement

La dimension environnement regroupe tous les autres facteurs contextuels qui ne font pas partie des dimensions présentées dans les sections précédentes. Généralement cette dimension se focalise sur les facteurs spatio-temporels mais la liste peut être plus longue. L'un des premiers facteurs environnementaux ayant été intégré dans la RI contextuelle est la **localisation de l'utilisateur** (Abowd et al., 1997). En effet, la localisation est l'aspect le plus commun du contexte spatial. Des travaux ont proposé des moteurs de recherche qui personnalisent leurs résultats selon le facteur environnemental lié à la position géographique des utilisateurs (Watters & Amoudi, 2003).

D'autres travaux se focalisent sur les entités autour de l'utilisateur comme faisant partie de l'environnement. Cela inclut des **objets de l'environnement** (ex.: bâtiments, infrastructure, etc.) et leur état (température, lumière, humidité, bruit). Par ailleurs, les dispositifs qui sont à proximité des utilisateurs sont également comptabilisés comme faisant partie de l'environnement. Quelques travaux ont étudié l'impact des caractéristiques de ces dispositifs sur le processus de personnalisation de la RI (Ayşe Göker & Myrhaug, 2008; Schmidt, Beigl, et al., 1999). Pour donner quelques exemples de ces derniers, nous pouvons citer la taille de l'écran, la résolution, les capacités sonores, le nombre de périphériques d'entrée, etc.

Par ailleurs, des travaux se focalisent sur d'autres facteurs environnementaux comme la **qualité de l'information** proposée en étudiant son caractère temporel (actualisation), la crédibilité des sources informationnelles, le format des documents renvoyés, etc.

Plus encore, des approches se sont également tournées vers le **contexte social de l'utilisateur** comme élément de l'environnement. Ainsi, la dimension sociale permet de décrire les relations des utilisateurs avec d'autres personnes qui lui sont connectées et ayant les mêmes préférences informationnelles par exemple. Nous trouvons les premières études à avoir intégré cette dimension sociale dans l'environnement dans le domaine du filtrage d'information et plus exactement du filtrage collaboratif d'information (Ricci et al., 2011) (cf. section II.2.1).

Dans la taxonomie du contexte de Fuhr (Fuhr, 2000), l'environnement de recherche comprend deux dimensions. La première dimension sociale qui s'intéresse à l'appartenance

ou non de l'utilisateur à une communauté. La deuxième dimension environnementale est le temps qui permet de définir le contexte temporel du besoin d'information.

Pour Cool et ses collaborateurs (Cool & Spink, 2002) l'environnement de recherche comprend les facteurs professionnels, cognitifs et sociaux qui ont une influence sur le comportement des utilisateurs durant la phase de recherche. Enfin, Ingwersen et ses collaborateurs (Ingwersen & Järvelin, 2005b) ajoutent aux facteurs environnementaux cités plus haut les caractéristiques des documents disponibles ainsi que celles du système utilisé. Les auteurs incluent aussi la caractéristique de l'interaction de l'accès à l'information parmi les facteurs contextuels pertinents faisant partie de la dimension environnement. Dans leur approche, l'interaction peut être caractérisée selon sa durée (interactions à court terme, à long terme ou durant une session de recherche) ou son mode d'interaction.

VI. Conclusion

Nous avons présenté dans ce chapitre un tour d'horizon du domaine de l'accès à l'information et surtout l'émergence de l'intégration du contexte dans les processus d'accès à l'information. Nous avons exposé les travaux se focalisant sur le contexte dans la RI selon quatre principaux niveaux, qui constituent un cadre pour ces travaux. Ces derniers sont : niveau environnemental, niveau humain de RI, niveau interactionnel de la RI et enfin niveau de la requête.

Nous avons également exploré plus spécifiquement dans un deuxième temps, les travaux de RI contextuelle selon les trois dimensions les plus importantes au sein du processus de contextualisation informationnelle. En effet, les dimensions **usager**, **tâche** (tâche de travail et tâche informationnelle) et **environnement** constituent les ensembles de tous les éléments les plus pertinents du contexte pouvant améliorer le processus de recherche.

Cependant, il est difficile de trouver un système de RI contextuel gérant toutes les dimensions contextuelles simultanément. En effet, les travaux que nous avons présentés dans ce chapitre se focalisent sur quelques aspects du contexte selon leur domaine d'application. Les dimensions qu'ils considèrent lors de la prise en compte du contexte peuvent être plus ou moins pertinentes selon les domaines des recherches effectuées. Aucune approche n'envisage d'incorporer tous les éléments du contexte dans la contextualisation de la RI, probablement parce que ces éléments sont très nombreux et dépendent de l'application qui motive la RI.

Avant de passer à la deuxième partie de notre manuscrit focalisé sur notre contribution, nous allons conclure cet état de l'art par une synthèse des deux premiers chapitres en soulignant les lacunes des approches présentées, en plus des solutions que nous proposons pour les combler.

Conclusion état de l'art

L'intérêt pour la notion de contexte ne cesse de grandir dans la communauté informatique. Comme nous l'avons signalé dans le premier chapitre de l'état de l'art (cf. chapitre I.1 section II), le contexte est utilisé dans presque tous les domaines informatiques mais surtout dans le domaine de l'informatique ubiquitaire. Ainsi, la multitude de travaux réalisés a montré les multiples avantages de l'intégration du contexte dans les nouveaux systèmes, dits sensibles au contexte notamment par l'adaptation des données et des services.

Par conséquent, de nombreuses recherches ont proposé plusieurs définitions de cette notion de contexte dans le but de mieux l'intégrer dans les systèmes et de la gérer dans sa complexité. La première limite de ces travaux réside dans le fait qu'ils divergent sur la définition même de la notion de contexte. Ainsi, les définitions présentées dans le premier chapitre sont toutes très spécifiques. Autrement dit, chaque définition présente une vision du contexte qui est propre à son domaine d'application. Plus encore, elles ne considèrent pas toutes les dimensions contextuelles disponibles et prennent donc exclusivement en compte les dimensions spécifiques aux besoins de l'application à contextualiser.

Nous nous sommes donc rendu compte du manque de définition générique de la notion de contexte dans la littérature. Néanmoins, nous sommes conscients que proposer une approche générique du contexte représente un défi de taille. Premièrement, car il est difficile de spécifier tous les éléments du contexte en amont, du fait, comme nous l'avons indiqué dans l'état de l'art, que le contexte n'est pas prévisible et qu'il peut évoluer. En d'autres termes, d'autres dimensions contextuelles considérées pertinentes peuvent venir s'ajouter au contexte initial. Deuxièmement, la difficulté d'avoir une approche générique est due à la convergence des définitions du contexte vers un côté davantage applicatif que conceptuel. En effet, les définitions ont été négligées au profit des modèles du contexte sur lesquels se basent les systèmes. Ainsi, pour ces travaux, le fait de prendre en compte plusieurs dimensions est considérablement difficile. Ils s'orientent donc uniquement vers les quelques dimensions qui sont importantes pour leurs champs d'application.

L'originalité de notre approche est la généralité de la définition du contexte. En effet, nous nous basons sur **les deux vues du contexte** proposées par Dourish (Dourish, 2004) que nous avons présentées dans la section III du premier chapitre de l'état de l'art à savoir la vue *représentative* et la vue *interactionnelle*. De cette manière, nous pouvons, en utilisant ces deux visions, éloigner la définition du contexte de son aspect applicatif et ainsi obtenir une certaine généralité. Ceci permettant donc d'avoir une définition du contexte, qu'il s'agisse de

contexte en général ou de contexte métier, destinée à tous les domaines informatiques, aussi bien le domaine de l'informatique ubiquitaire que celui de l'accès à l'information. En d'autres termes, nous sommes convaincus par la vue interactionnelle de Dourish qui suppose que le contexte est activement produit, entretenu et promulgué dans le courant de l'activité. Toutefois, le contexte spécifique produit doit se baser sur la vue représentationnelle du contexte de Dourish. Ce dernier suppose qu'il est défini par un ensemble fixé de caractéristiques observables, dont la structure n'a pas changé significativement au cours du temps. Par conséquent, nous supposons que les deux points de vue devraient se compléter mutuellement plutôt que s'opposer.

Dans le deuxième aspect concernant la gestion des éléments contextuels, les travaux présentés dans l'état de l'art ont évidemment hérité des limites liées à la non-généricité de leurs définitions. En effet, malgré la multitude des approches de gestion du contexte présentées dans le premier chapitre de l'état de l'art, nous n'avons pas souligné les travaux proposant un gestionnaire de contexte générique prenant en compte tout ce qui peut être qualifié de dimension contextuelle, y compris l'utilisateur lui-même. Plus encore, une deuxième limite attribuée aux travaux de la littérature, réside dans le fait que les gestionnaires de contexte proposés ne prennent pas en considération les éléments contextuels les uns par rapport aux autres au moment de la contextualisation, même si ces derniers peuvent interagir les uns sur les autres. Nous rappelons que même si certains travaux ont utilisé la couche de prétraitement (cf. chapitre 1 section V.1), celle-ci gère simplement l'agrégation des éléments contextuels pour leur ajouter du sens et ne permettent pas une modification dans le modèle ou dans les valeurs des dimensions (éléments) en entrée, en se basant sur leur conjonction pour une contextualisation demandée.

Pour résoudre les problèmes récemment présentés, nous avons voulu proposer un gestionnaire de situations contextuelles qui permette dans un premier temps de **contrôler** les dimensions à contextualiser. C'est-à-dire, vérifier que les différents éléments en entrée peuvent être combinés pour une contextualisation donnée. Nous notons que ce premier point est particulièrement important dans un contexte métier, but de notre travail. De plus, le gestionnaire doit prendre en considération l'impact des éléments contextuels en entrée les uns par rapport aux autres. Ainsi, un tel gestionnaire doit permettre d'éventuelles mises à jour des éléments de départ dans **la situation contextuelle** qui devient le résultat d'un processus d'interprétation du contexte.

Concernant la notion de situation, les travaux présents dans l'état de l'art (cf. chapitre 1 section VII) se sont concentrés sur cette notion pour monter du niveau des informations de contexte brutes à un niveau d'abstraction plus élevé. Cette dernière permet d'inférer principalement l'activité de l'utilisateur. Ces travaux ne font qu'agrèger les informations contextuelles afin d'en générer une vue globale possédant du sens : la situation. Ainsi, ils ne se préoccupent pas des éventuelles **relations** entre des éléments composants cette situation et ce qu'elles peuvent engendrer comme modification dans l'état final de la situation elle-même. Par conséquent, la fidélité de la situation à la réalité peut être remise en cause dans les travaux de la littérature. Ce dernier point peut avoir des conséquences négatives dans des applications pour lesquelles tous les éléments contextuels sont pertinents dans leur processus de contextualisation. D'un autre côté, une deuxième limite réside dans l'absence d'approche utilisant le **feedback** sur les situations enregistrées (historique du contexte) pour améliorer le processus de contextualisation (création des situations) ou rendre possible une telle contextualisation. En effet, les approches propres à la littérature utilisent l'historique de contexte afin d'améliorer le processus de prédiction des situations (activités) pour mieux se rapprocher du besoin global de l'utilisateur, mais sans feedback.

Pour cela, nous proposons une **génération de situation** la plus fidèle possible à la réalité. Dans cette situation, tous les éléments contextuels ont une importance et peuvent être en relation les uns avec les autres en utilisant des connaissances contextuelles supplémentaires. Enfin, le feedback sur ces situations passées ne pourra qu'améliorer a posteriori ce processus de contextualisation. En plus de cela, le feedback des situations passées permet dans le domaine de l'accès à l'information d'adapter le processus informationnel. En effet, en comparant les situations passées, le système proposera à l'utilisateur des actions faites par d'autres utilisateurs qui étaient dans des situations « similaires ».

Nous souhaitons terminer cette partie en faisant le lien entre le contexte d'un point vue global et le contexte dans l'accès à l'information abordé dans le deuxième chapitre de l'état de l'art. Le domaine d'application que nous avons abordé dans notre approche générique de gestion de contexte métier correspond parfaitement à cette dernière et montre ainsi l'efficacité de notre approche à résoudre les limites des travaux existants dans ce domaine. En effet, comme tous les travaux sur le contexte, les approches proposées pour la contextualisation de l'accès à l'information ne convergent pas vers les mêmes dimensions à intégrer. Plus encore,

la spécificité de l'accès à l'information dans un **contexte métier** réside dans le fait que le système est utilisé pour trouver l'information (manquante) nécessaire à la réalisation de la tâche métier. Cette dernière fait partie d'un processus métier bien organisé par le contexte de travail. C'est-à-dire que la tâche métier appartient à une arborescence de tâches et elle est liée aux autres tâches métier. Ainsi, une des limites des systèmes actuels est qu'ils n'ont pas accès à cette modélisation des tâches métier ni aux conjonctions qu'il peut y avoir entre toutes les parties du contexte : les tâches métier et les tâches informationnelles qui leur sont appropriées, les usagers et l'environnement dans lequel se déroule l'activité.

Pour cela, la situation que nous générons peut être fournie aux systèmes d'accès à l'information parce qu'elle prend en considération toutes conjonctions possibles de dimensions contextuelles non prédéterminées, pertinentes pour favoriser l'accès à l'information. Enfin, comme nous l'avons expliqué plus haut, le point fort de cette situation est qu'elle est la plus fidèle possible à la réalité. Ainsi, les systèmes d'accès à l'information se basant sur de telles situations fourniront aux usagers des informations pertinentes nécessaires à leur tâche et tenant compte de l'ensemble complet du contexte de travail.

Partie II : Contribution

Chapitre II.1 : **Le questionnaire** **de situations** **contextuelles métier**

I.	Introduction.....	115
II.	Pourquoi un gestionnaire de situations contextuelles ?	115
III.	Le contexte	116
	III.1. La définition du contexte.....	117
	III.2. Les concepts du contexte.....	118
	La dimension contextuelle	119
	L'élément contextuel.....	119
	a) Nom.....	119
	b) Valeur.....	119
	c) Constance	119
	d) Fiabilité	120
IV.	Interprétation du contexte	120
	IV.1. Exemples d'interprétation du contexte.....	120
	a) Exemple de contexte initial	121
	b) Exemple d'instances de dimensions contextuelles.....	121
	c) Exemple 1 : conservation des dimensions initiales.....	122
	d) Exemple 2 : modification des modèles initiaux des dimensions contextuelles	122
	e) Exemple 3 : modification des valeurs des éléments contextuels initiaux	123
	IV.2. Rôles des interprétations du contexte.....	124
	IV.2.1. Validité de l'interprétation du contexte	125
	IV.2.2. Prise en compte de l'impact des éléments contextuels les uns sur les autres.....	125
	IV.2.3. Prise en compte des informations contextuelles manquantes.....	126
V.	La Situation	128
	V.1. Objectifs	128
	V.2. Définition de la situation	128
	V.3. Les concepts de la situation.....	129
	V.4. Exemple de situation	130
VI.	Gestionnaire de situations contextuelles	130
	VI.1. Architecture du gestionnaire de situations contextuelles	131
	VI.2. Le gestionnaire de données contextuelles	132

VI.2.1.	Présentation	132
VI.2.2.	La base des règles	133
a)	Les trois types de règles	133
b)	La définition des règles	134
VI.3.	Le processus de Mise En Situation	135
VI.4.	Le processus d'extraction de règles.....	137
VI.5.	Les engagements du GSC envers les applications	138
VI.5.1.	Vérification des situations valides	138
VI.5.2.	Comblé le manque d'information dans le contexte.....	139
VI.5.3.	Proactivité/anticipation	140
VI.5.4.	Fournir les éléments contextuels	140
VI.6.	Synthèse	140
VII.	Démarche de création de la situation	141
VII.1.	Les informations contextuelles nécessaires à la création des situations	142
VII.1.1.	Les informations contextuelles transitant par le GDC.....	142
VII.1.2.	Les informations contextuelles transitant par le MES.....	144
VII.2.	Fonctionnement du processus MES	146
VII.3.	Algorithme général du MES	146
VII.4.	Adaptation des dimensions contextuelles	147
VII.4.1.	Principe	147
VII.4.2.	Exemple illustratif	148
VII.5.	Stabilisation du contexte	150
VII.5.1.	Principe	150
VII.5.2.	L'algorithme de stabilisation du contexte	150
VII.5.3.	Illustration de l'algorithme de la stabilisation	152
VIII.	La démarche d'extraction de règles	154
VIII.1.	La fouille de règles d'association.....	155
VIII.1.1.	Les concepts de la fouille de règles d'association	155
VIII.1.2.	Exemple	157
VIII.2.	Fonctionnement du processus d'extraction des règles	158

VIII.2.1.	Motivation.....	158
VIII.2.2.	Les étapes de l'extraction des règles.....	160
VIII.2.3.	L'algorithme d'extraction de règles.....	161
VIII.2.4.	Exemple d'extraction de règles dans notre approche	162
IX.	Conclusion.....	164

I. Introduction

Nous présentons notre contribution dans cette deuxième partie de notre manuscrit. En effet, nous exposerons dans ce chapitre le Gestionnaire de Situations Contextuelles (GSC) qui se fonde sur l'ensemble des approches que nous proposons et qui constitue l'apport de notre travail. Le GSC a un rôle primordial pour garantir la pertinence et la précision des informations contextuelles provenant des différentes sources du contexte. Ces informations sont proposées aux systèmes tiers qui font appel à notre gestionnaire à des fins d'adaptation par exemple.

Le GSC, comme la définition de contexte qui en est la base, est générique et doit prendre en considération tous les éléments contextuels qualifiés de pertinents pour une application ou pour un système donné. Ainsi, nous exposons dans ce chapitre cette nouvelle définition du contexte qui vise à donner à cette notion une généricité et une base commune issue des différents travaux.

Nous présentons également les éléments qui composent le contexte dans notre approche, en détaillant les notions intrinsèques à notre vision du contexte. Nous détaillons aussi la relation entre situation et contexte comme nous l'avons fait dans la section VII du chapitre I.1 de l'état de l'art. À cet effet, nous exposons notre propre définition de la notion de situation comme une interprétation du contexte à un instant donné. La situation renferme ainsi un ensemble défini d'informations provenant du contexte. La situation que nous proposons est donc une représentation particulière du contexte à un instant t résultant de l'interaction entre les éléments qui le composent.

II. Pourquoi un gestionnaire de situations contextuelles ?

Le GSC doit être le garant de la collecte, de la gestion et de la présentation des informations contextuelles pour le bénéfice des applications. Il permet de proposer un **service intermédiaire de gestion de contexte** essentiel pour les applications parce qu'il *gère l'accès aux sources de contexte en plus d'effectuer de l'inférence sur les informations contextuelles*. Les gestionnaires de contexte sont utilisés principalement dans le domaine de l'informatique ubiquitaire, c'est-à-dire des systèmes sensibles au contexte. Ce type de systèmes distribués opère dans des environnements dont ils doivent continuellement gérer le contexte d'exécution pour détecter les conditions de l'adaptation.

L'objectif principal du GSC est de fournir des **situations contextuelles** réalistes. Ces situations enregistrées dans le temps représentent l'historique du contexte et peuvent être utilisées par les applications reposant sur notre approche de deux façons :

- une utilisation à court terme comme exploiter des situations enregistrées pour l'adaptation et la personnalisation ;
- l'utilisation des situations proposées par le gestionnaire peut être un moyen de prédiction des valeurs d'éléments contextuels dans les situations futures.

Les situations contextuelles sont le noyau de notre approche. Elles doivent donc être les plus fidèles possibles à la réalité. Autrement dit, les situations reflètent une **conjonction approuvée** des différents éléments contextuels composant le contexte. Pour ce faire, nous avons particulièrement besoin du GSC afin de mettre en relation ces éléments du contexte les uns avec les autres en utilisant des connaissances contextuelles. Ce besoin est naturellement né avec celui de créer les situations. En effet, la gestion de la connaissance contextuelle est indispensable pour donner du sens aux situations. Autrement dit, le GSC doit avoir cette connaissance qui lui permet, dans un premier temps, de lier les uns aux autres les éléments contextuels présents dans les différentes dimensions et de gérer leur impact dans la situation produite. Le GSC gère cette connaissance que nous modélisons sous forme de règles. Ces règles seront présentées dans la section VI.2.2.

Comme nous l'avons décrit précédemment, le GSC se base sur les informations contextuelles pour créer les situations. Pour cela, nous exposerons dans ce qui suit notre propre vision du contexte, sa définition et ses composants.

III. Le contexte

L'état de l'art présenté dans la première partie du mémoire a mis en lumière l'axe principal de notre travail qui est la gestion de contexte dans le cadre de l'accès à l'information. Le contexte en général a constitué notre point de départ vers la notion de contexte dans l'accès à l'information. Nous allons appliquer la même démarche pour la présentation de cette notion de contexte générique dans notre approche. Nous allons aborder le problème d'une manière générale. En effet, nous nous devons de présenter les (nombreuses) spécificités propres au contexte afin de pouvoir l'instancier dans le domaine de l'accès à l'information.

Les définitions de la notion de contexte présentées dans l'état de l'art convergent et s'opposent à la fois sur plusieurs points, notamment selon les domaines d'applications ciblés. En réalité, la divergence de ces définitions est surtout liée à l'objet pour lequel le contexte est défini. Ce dernier dépend donc de l'entité ou de l'acteur auquel le contexte est destiné (Göker et al., 2009). Toutefois, les définitions divergent surtout par leur spécificité. Dans ces définitions du contexte, les auteurs se focalisent sur des aspects bien particuliers du contexte, i.e. ils se concentrent sur les éléments bien définis de ce dernier selon les besoins de leurs applications. Par exemple, les travaux s'inscrivant dans la contextualisation de l'accès à l'information se focalisent principalement sur les caractéristiques de l'utilisateur contrairement aux travaux en informatique ubiquitaire qui ont tendance à privilégier les facteurs environnementaux. La définition du contexte doit donc être générique afin d'y intégrer tout ce qui peut être considéré comme dimensions contextuelles.

III.1. La définition du contexte

Quelques travaux de référence dans la formalisation du contexte soulignent le fait que le contexte doit être relatif à quelque chose : le contexte d'une action, les interactions avec le système, etc. (Pomerol & Brézillon, 2001). Ils ont défini la connaissance contextuelle « comme la partie du contexte qui est pertinente dans une situation donnée pour un opérateur donné ». Cela peut être considéré comme le sous-ensemble du contexte dans lequel l'utilisateur peut trouver les connaissances nécessaires pour interpréter et expliquer la situation. En outre, si nous indiquons que le contexte est relatif à quelque chose, nous pouvons en déduire que chaque dimension contextuelle possède son propre contexte qui est formé par les dimensions contextuelles restantes du contexte global.

En partant de cette idée, nous pouvons donner notre définition du contexte d'un objet :

Définition. Le contexte d'un objet correspond à toutes les dimensions contextuelles pouvant avoir un impact sur cet objet.

En d'autres termes, chaque objet du contexte possède, lui aussi, son propre contexte. Ainsi, la définition que nous proposons met en évidence les interactions qui existent entre les différents contextes des objets. Des dimensions contextuelles qui n'ont pas d'impact (une influence) sur un objet ne peuvent donc pas être considérées comme parties intégrantes de son contexte. Par exemple, pour l'objet contextuel *usager*, son contexte pourrait être composé de la *tâche* qu'il réalise ainsi que de l'*environnement* dans lequel il se trouve parce qu'ils peuvent

avoir une influence directe sur l'*usager*. Par ailleurs, de manière similaire, la *tâche* a aussi son propre contexte ; ce dernier pourrait être composé de l'*usager* qui l'effectue et de l'*environnement* dans lequel elle est réalisée, seulement si ces deux dimensions ont un impact sur elle.

III.2. Les concepts du contexte

Le contexte comme nous le proposons dans nos travaux (Chaker, Chevalier, Soulé-Dupuy, & Tricot, 2010a) est défini par :

- Contexte = { dimension contextuelle }
- Dimension contextuelle = { Élément contextuel }
- Élément contextuel = { Élément contextuel } | { Nom, Valeur, Constance, Fiabilité }

Dans ce qui suit, nous détaillons ces différents concepts et les éléments qui les composent en se basant sur l'exemple illustré dans la figure III.1.

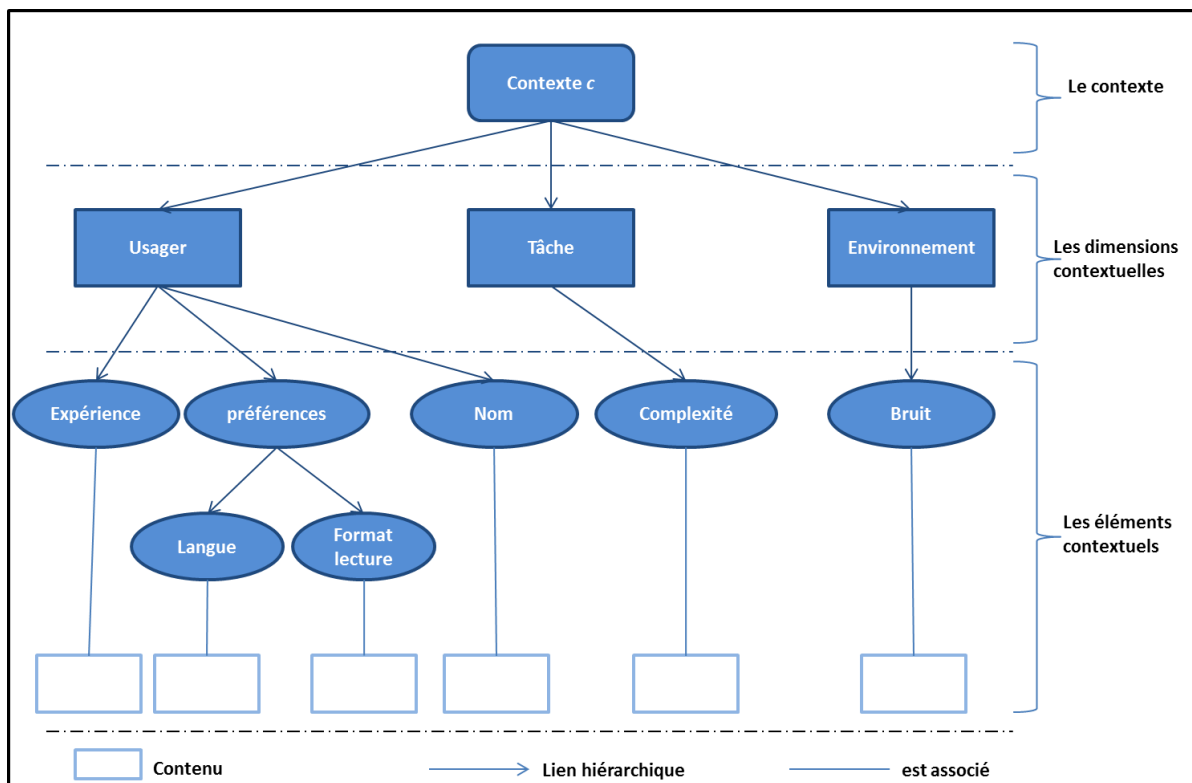


Figure III.1. Un exemple de contexte

La dimension contextuelle

Les dimensions contextuelles sont les dimensions composant un contexte et regroupant tout ce qui peut être quantifié et regroupé dans une dimension pour décrire la même entité du monde réel. Les dimensions contextuelles sont composées d'éléments contextuels et ont une structure hiérarchique.

L'élément contextuel

Les éléments contextuels ont aussi une structure hiérarchique. Ils peuvent être soit des éléments contextuels soit les nœuds feuilles des dimensions contextuelles. Par exemple, l'élément contextuel préférences (cf. figure III.1) dans la dimension contextuelle « usager » est composé d'éléments contextuels élémentaires tels que la « langue préférée » et « le format de lecture préféré ».

Un élément contextuel élémentaire est caractérisé par : un **Nom**, une **Valeur**, une variable **Constance** et la **Fiabilité** de l'élément. Nous détaillons dans ce qui suit chacune des caractéristiques d'un élément contextuel.

a) Nom

Le nom de l'élément contextuel est ce qui le différencie des autres éléments contextuels des différentes dimensions.

b) Valeur

Un élément contextuel est également caractérisé par sa valeur. La valeur possède un type particulier qui peut être qualitatif (chaîne de caractères, caractère...) ou quantitatif (un réel, un entier).

c) Constance

Un élément contextuel est caractérisé par une valeur booléenne qui indique si l'élément peut évoluer d'une situation à une autre dans le temps. Dans notre exemple (cf. figure III.1) l'élément contextuel « *nom utilisateur* » appartenant à la dimension contextuelle « usager » pourrait être considéré comme constant, contrairement à un l'élément contextuel « *expérience du domaine* » qui peut évoluer selon les situations (l'expérience du domaine d'un usager va augmenter dans le temps en fonction des activités accomplies et des heures de travail). La caractéristique *Constance*, permet de savoir quels sont les éléments contextuels qui peuvent ou ne peuvent pas évoluer dans le temps.

d) Fiabilité

Un élément contextuel est caractérisé par sa valeur de fiabilité (de type réel) qui permet au GSC de mesurer peu à peu la précision de la valeur. En effet, certains changements peuvent se produire sur les valeurs des caractéristiques des éléments contextuels à cause des éventuelles interactions entre les dimensions contextuelles. Il est donc important de proposer au système d'information une valeur de fiabilité pour chacun des éléments contextuels. Il est évident qu'un élément contextuel dont la valeur est communiquée directement d'une source de contexte sûre (exemple des capteurs) possède une fiabilité maximale, i.e. égale à 1.

Un exemple d'élément contextuel, si on prend la dimension contextuelle « usager » :
Nom : Expérience, **Valeur** : Expert, **Constance** : Vrai, **Fiabilité** : 0.5.

IV. Interprétation du contexte

Après avoir exposé notre définition générique de la notion de contexte, cette section illustre l'interprétation de cette notion. En effet, pour des besoins applicatifs, l'interprétation du contexte est indispensable au processus de sélection et par la suite à la conjonction des différents éléments contextuels. Nous devons alors expliquer dans un premier temps les différences entre le contexte et son interprétation. Ensuite, nous devons montrer l'intérêt de l'interprétation dans notre approche.

L'interprétation d'un contexte doit être considérée en priorité par rapport aux informations contextuelles initiales, car elle comporte en plus des dimensions contextuelles ainsi que les éléments contextuels les plus pertinents pour une activité donnée, des connaissances contextuelles supplémentaires.

Nous introduisons dans ce qui suit la notion d'interprétation à travers d'exemples en utilisant un jeu de dimensions issues d'un contexte initial. Les exemples proposés permettent dans un deuxième temps de montrer les rôles de l'interprétation de contexte

IV.1. Exemples d'interprétation du contexte

Dans cette section, nous allons donner quelques exemples simplifiés d'interprétations d'un contexte initial. Ces exemples illustrent les différents cas de figure de l'état du contexte et ses possibilités d'interprétation.

a) Exemple de contexte initial

Nous allons prendre un exemple simplifié de contexte initial que nous appelons *CI*. Cet exemple est considéré comme simple parce que le contexte *CI* n'est composé que de deux dimensions contextuelles comportant chacune uniquement deux éléments contextuels. Les deux dimensions sont nommées *DCA* et *DCB*. La figure III.2 illustre les deux dimensions contextuelles et les deux éléments contextuels qui composent chacune d'elles. La première dimension *DCA* est composée des deux éléments contextuels *EA1* et *EA2*. La deuxième dimension *DCB* est elle composée des deux éléments contextuels *EB1* et *EB2*.

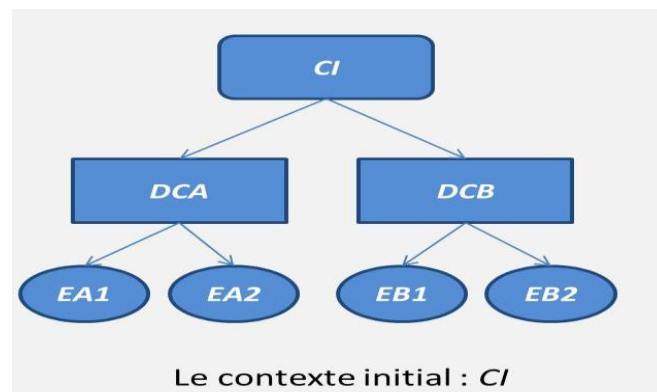


Figure III.2. Exemple de contexte initial composé de deux dimensions contextuelles

b) Exemple d'instances de dimensions contextuelles

En prenant le contexte initial *CI*, nous présentons trois exemples simplifiés destinés à mettre en lumière la différence entre les informations contextuelles initiales et leur interprétation. Similairement, à travers ces exemples nous montrons les différentes interprétations possibles pour un ensemble d'instances de dimensions contextuelles issues d'un même contexte initial *CI*. Les instances sont issues des dimensions contextuelles *DCA* et *DCB* (cf. figure III.2). Nous reprenons dans la figure III.3 le contexte initial *CI* en nous focalisons sur les deux instances *A1* et *B1* des dimensions contextuelles *DCA* et *DCB*. Ainsi, *A1* et *B1* sont caractérisées par les valeurs (a, b, c, d) des éléments contextuels qu'elles comportent (cf. figure III.3).

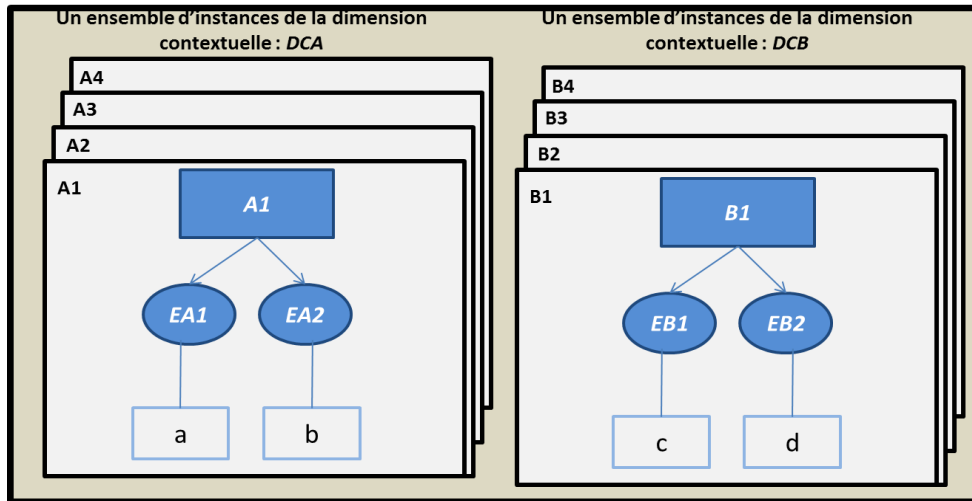


Figure III.3. Exemple d'instances issues des dimensions contextuelles de *CI*

Remarque : les deux instances *A1* et *B1* seront utilisées dans tous les exemples de cette section pour illustrer les différentes possibilités d'interprétation du contexte.

c) Exemple 1 : conservation des dimensions initiales

Dans la figure III.4 nous illustrons le premier cas d'interprétation du contexte. En effet, dans cette interprétation les deux instances des dimensions contextuelles de départ sont conservées fidèlement. Dans ce cas, l'instance de contexte à un instant donné est un ensemble composé de tous les éléments contextuels issus des deux instances de dimensions contextuelles *A1* et *B1* (cf. figure III.3).

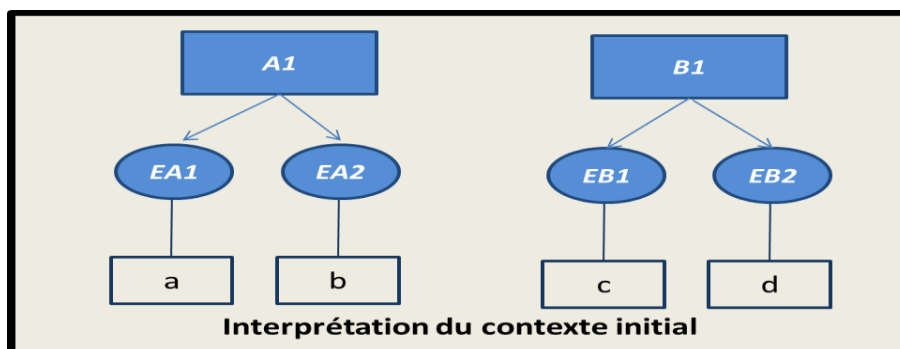


Figure III.4. Exemple d'interprétation de contexte avec conservation des dimensions initiales

d) Exemple 2 : modification des modèles initiaux des dimensions contextuelles

La figure III.5 présente un deuxième cas de figure d'interprétation du contexte à un instant donné. Dans cette interprétation les deux instances des dimensions contextuelles de départ sont conservées. Néanmoins, comme l'illustre la figure III.5, l'instance *B1* de la

dimension contextuelle *DCB* ne possède désormais qu'un seul élément contextuel *EB1* au lieu des deux éléments de départ. En effet, le modèle initial des dimensions contextuelles peut être modifié dans l'interprétation du contexte pour, par exemple, les besoins d'une activité donnée. Si nous reprenons notre exemple initial, l'élément contextuel *EB2* a alors été supprimé de la dimension contextuelle *B1*. Autrement dit, l'élément *EB2* est facultatif et n'a donc pas d'impact sur le déroulement de l'activité donnée à l'instant *t*. Ainsi, le résultat de l'interprétation du contexte à un instant *t* est différent du contexte de départ.

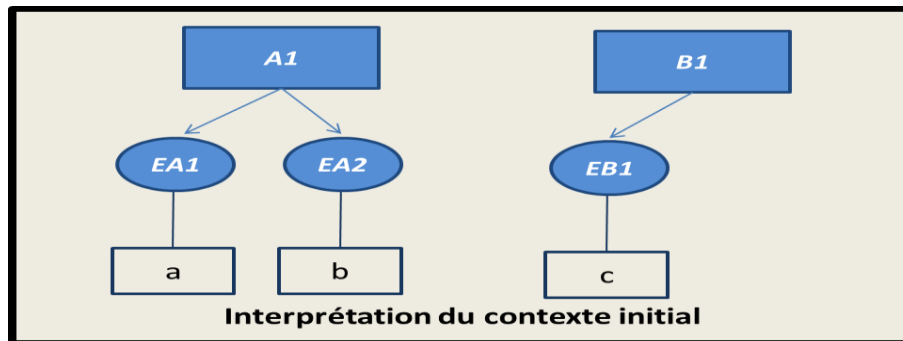


Figure III.5. Exemple d'interprétation de contexte avec modification des modèles initiaux des dimensions contextuelles

e) Exemple 3 : modification des valeurs des éléments contextuels initiaux

La figure III.6 présente un troisième et dernier cas de figure d'interprétation du contexte dans le temps. Dans cette interprétation, les deux modèles des dimensions contextuelles de départ sont conservés. Toutefois, la modification dans ce cas de figure touche les valeurs des éléments contextuels. Ainsi, la valeur de l'élément contextuel *EA1* appartenant à la dimension contextuelle *A1* passe d'une valeur de départ « *a* » à une valeur « *a'* » dans l'interprétation. De ce fait, le résultat de l'interprétation du contexte à un instant donné est donc différent du contexte de départ. Par rapport aux caractéristiques des éléments contextuels présentées dans la section III.2, nous notons que la valeur « *a'* » peut remplacer la valeur déjà présente dans le contextuel initial, si et seulement si sa **fiabilité** est supérieure à celle de l'élément du contexte initial (dans notre exemple, c'est la valeur « *a* »).

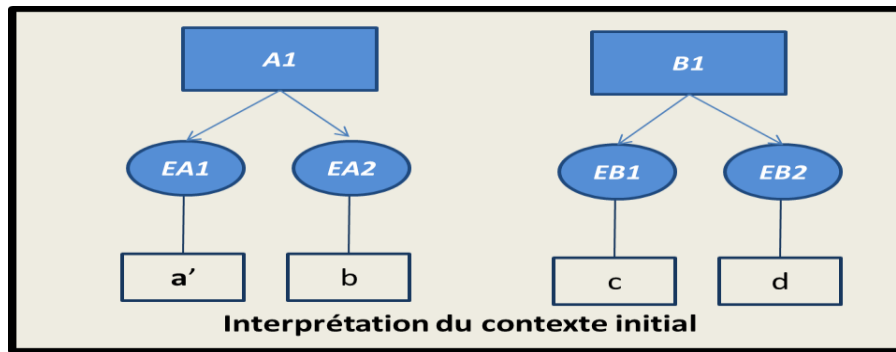


Figure III.6. Exemple d'interprétation de contexte avec modification dans les valeurs des éléments contextuels initiaux

En guise de conclusion, nous pouvons constater dans les exemples présentés précédemment que, pour un simple contexte initial donné, nous pouvons avoir plusieurs interprétations possibles. Nous allons évoluer dans notre approche en spécifiant les motivations pour ces interprétations et plus précisément, ce que leur utilisation peut ajouter par rapport aux contextes de départ. C'est pourquoi nous abordons dans la section suivante les rôles des interprétations du contexte. De la sorte, nous expliquons comment les interprétations du contexte sont indispensables aux applications se basant sur les informations contextuelles. De même, nous détaillerons **les raisons** de certains écarts, entre les interprétations du contexte et leurs contextes de départ.

IV.2. Rôles des interprétations du contexte

Dans notre approche, une interprétation du contexte est fondamentale à la contextualisation des applications sensibles au contexte. Cette importance peut être interprétée selon trois niveaux complémentaires :

- premièrement, selon ce que nous appelons la **validité** (qui est présentée dans la section suivante). Cette dernière s'intéresse particulièrement à la possibilité de coexistence des différentes dimensions contextuelles et ainsi des différents éléments contextuels issus du contexte initial ;
- le deuxième point concerne les conséquences de cette coexistence des différents éléments contextuels. Autrement dit, ce qui nous intéresse est l'**impact de ces éléments contextuels les uns sur les autres** dans une même interprétation de contexte ;
- en dernier lieu, nous abordons le besoin des interprétations suivant le **manque d'informations contextuelles** qui peut survenir dans les contextes initiaux.

IV.2.1. Validité de l'interprétation du contexte

Le premier point très important sur lequel nous montrons l'importance d'une interprétation du contexte est la validité de toute interprétation. En effet, la notion de validité établit le besoin de considérer et de vérifier la coexistence des éléments contextuels dans les interprétations de contexte au lieu d'envisager simplement les dimensions contextuelles et ce qu'elles contiennent comme éléments contextuels. Le fait de regrouper toutes les dimensions contextuelles - avec tous leurs éléments contextuels - issues d'un contexte de départ doit être contrôlé par un ensemble de connaissances contextuelles.

Par exemple, si nous reprenons l'exemple du contexte initial de la figure III.1. Un exemple d'interprétation invalide peut être un « usager » ayant l'élément contextuel « expérience » égal à « novice » (débutant dans son domaine), et une « tâche » ayant l'élément contextuel « complexité » égal à « contraignante » (exigeant un niveau minimum d'expérience). Ces deux instances de valeurs des deux éléments contextuels ne peuvent pas être réunies pour un objectif de contextualisation.

Nous ne pouvons donc pas considérer directement le contexte de départ si ses éléments ou une partie de ses éléments ne peuvent pas être contextualisés.

IV.2.2. Prise en compte de l'impact des éléments contextuels les uns sur les autres

Nous allons montrer dans un deuxième temps le besoin de l'interprétation de contexte pour les systèmes utilisant les informations contextuelles. En effet nous allons renforcer le premier niveau de la validité et ainsi aller plus loin dans notre raisonnement en considérant l'impact que peut avoir la coexistence de tous les éléments contextuels pris simultanément.

Pour cela, nous avons pris le même exemple de contexte initial *CI* avec les mêmes valeurs des éléments contextuels qui composent les instances des dimensions contextuelles *A1* et *B1* (cf. figure III.3). Supposons que nous ajoutons à cela une information contextuelle supplémentaire telle que : si l'élément contextuel *EAI* appartenant à *DCA* possède une valeur de « *a* » et si l'élément contextuel *EBI* appartenant à *DCB* possède une valeur de « *c* », alors l'élément contextuel *EB2* appartenant à *DCB* doit avoir une valeur de « *x* ». Pour détailler encore plus ce dernier, nous pouvons dire que cette information (ou contrainte) que nous venons d'ajouter est une règle que nous pouvons écrire de la façon suivante : *DCA. EAI = « a » et DCB. EBI = « c » → DCB. EB2 = « x ».*

Remarque : partie importante de notre approche, les règles sont définies dans la section VI.2.2.

Si nous appliquons donc cette règle alors l'instance du contexte initial va évoluer. En effet, comme le schématise la figure III.7, considérer seulement le contexte initial ne peut pas prendre en compte cette connaissance traduite par la règle. Certes, nous avons besoin d'une interprétation de contexte qui prend en compte d'autres connaissances contextuelles en plus des instances des dimensions et les traduit sur un support commun. Dans notre exemple, proposer le contexte initial hors règle à une application sensible au contexte est une déformation de la réalité comme nous l'illustrons sur la figure III.7 (a) et par conséquent accentue le besoin d'un autre niveau pour l'interprétation du contexte.

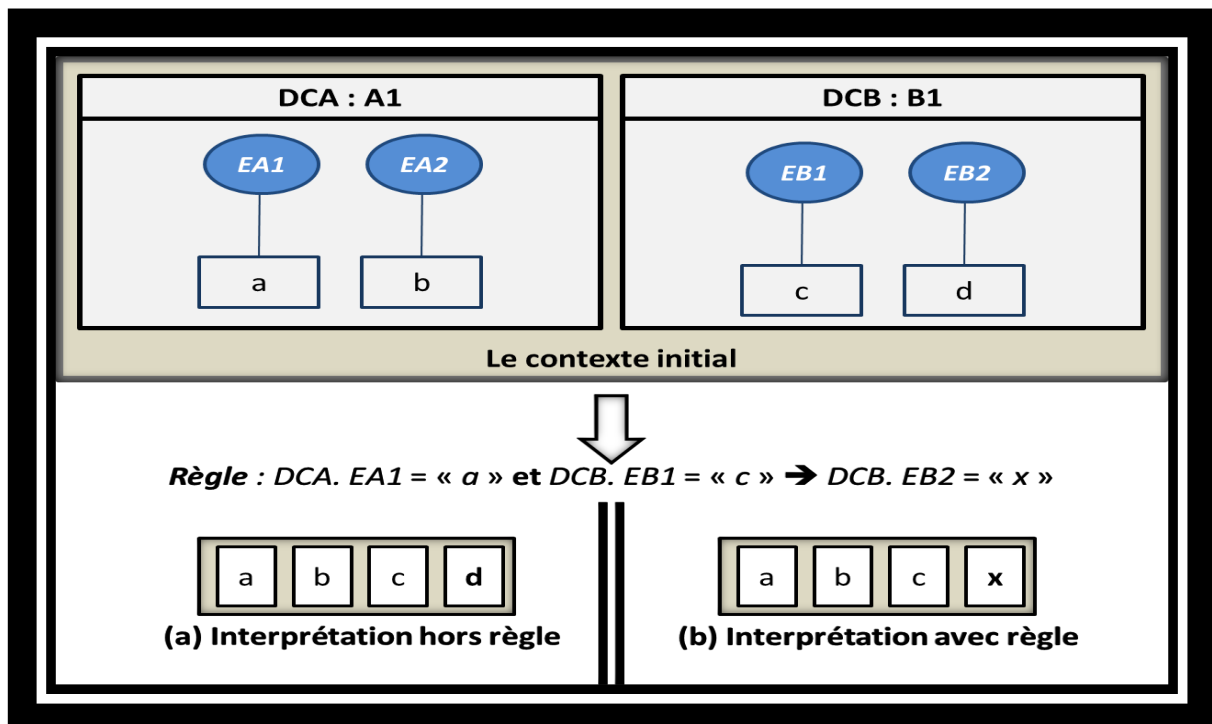


Figure III.7. L'impact des éléments contextuels dans l'interprétation

IV.2.3. Prise en compte des informations contextuelles manquantes

L'information contextuelle manquante est tout simplement une absence dans une ou plusieurs valeurs des éléments contextuels qui composent les dimensions contextuelles d'un contexte donné. La cause de cette absence peut être, par exemple, une défaillance dans un des capteurs qui fournissent les valeurs des différents éléments contextuels ou encore par une indisponibilité de l'information contextuelle au moment de l'instanciation du contexte...

La figure III.8 illustre un exemple simplifié de contexte avec deux dimensions contextuelles, où la valeur de l'élément contextuel *EB2* est manquante. Nous avons schématisé d'un côté (cf. figure III.8 (a)) l'éventuel contexte résultat en reproduisant fidèlement les valeurs des différents éléments contextuels de départ. De l'autre côté (cf. figure III.8 (b)), nous observons une interprétation de contexte qui va enrichir le contexte de départ en comblant les informations manquantes. C'est le cas de l'élément *EB2* dans notre exemple qui prend la valeur « *d'* ». En d'autres termes, la valeur « *d'* » est une valeur statistiquement possible pour *EB2* qui est inférée dans l'interprétation du contexte en utilisant des mécanismes utilisées par le GSC et que nous détaillerons dans les sections suivantes. Il est important de retenir ici le besoin de l'interprétation dans le cas des données manquantes. En effet, il est essentiel de proposer à une application sensible au contexte une interprétation du contexte à un instant donné qui soit la plus complète possible, i.e. elle doit fournir d'autres connaissances pour combler le vide dans les éléments contextuels.

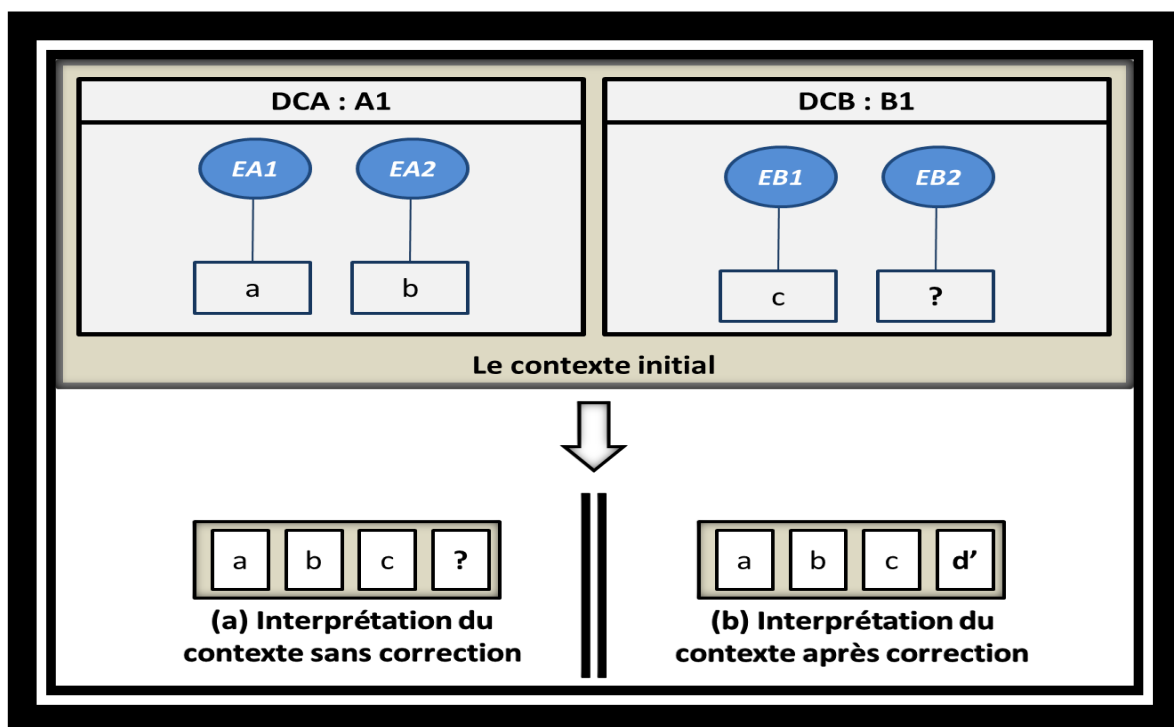


Figure III.8. Comblant les informations manquantes dans l'interprétation

Nous avons présenté trois niveaux pour montrer quel est notre besoin d'une interprétation de contexte et pourquoi nous ne considérons pas seulement les éléments contextuels de départ pour les applications sensibles au contexte. Ainsi, la conjonction de toutes ces exigences et la nécessité d'avoir des mécanismes pour interpréter le contexte nous conduisent vers la notion de situation. Dans la section suivante, nous définissons donc la

situation en nous basant à la fois sur l'interprétation du contexte et sur la définition générique de la notion de contexte présentée dans la section III.

V. La Situation

V.1. Objectifs

Nous proposons notre propre vision de la situation de la même manière que nous avons donné celle du contexte, c'est-à-dire, en gardant un niveau d'abstraction assez élevé pour montrer la généralité de notre approche. De ce fait, la situation offre selon notre approche le support d'interaction et d'adaptation du contexte. Les dimensions contextuelles qui composent le contexte en entrée sont reproduites et adaptées au niveau de la situation pour générer ensemble, à l'instant t , une situation unique. Ainsi, l'interaction entre les différentes dimensions contextuelles peut entraîner des modifications (insertions, modifications, suppressions, etc.) du contenu de ces dimensions issues du contexte (Chaker, Chevalier, Soulé-Dupuy, & Tricot, 2010b). Ces adaptations aux travers des éléments contextuels qui composent les différentes dimensions contextuelles, permettent de produire la photographie la plus fidèle possible d'une interprétation du contexte.

V.2. Définition de la situation

Nous présentons dans ce travail la notion de situation comme étant profondément liée à la définition du contexte. Similairement aux définitions du contexte, nous avons montré dans la section VII du chapitre I.1 qu'il n'existe pas une définition unique de la notion de situation dans la littérature. Les situations sont plus stables et plus faciles à définir que les informations contextuelles qui changent fréquemment (Bettini et al., 2010). L'interprétation est de fait pour nous un passage obligé vers la notion de situation, telle que la définition générique dans ce qui suit :

Définition. La situation est une interprétation stable d'un contexte à un instant t .
--

Cette notion même d'**interprétation stable** du contexte est le point le plus important de notre travail et en fait son originalité. La stabilité réside dans le fait que les différents éléments contextuels peuvent, en interagissant à un instant donné, avoir un effet les uns sur les autres. Cet effet, que nous appelons « adaptation des éléments contextuels », induit une modification des valeurs des éléments de départ. L'adaptation des éléments contextuels dans

une situation donnée s'arrête quand tous les éléments contextuels de cette situation n'ont plus d'impact les uns sur les autres. Prenons un exemple simplifié de trois dimensions contextuelles « usager », « tâche » et « environnement ». Une tâche effectuée par un usager peut avoir un impact sur lui-même (exemple : tâche contraignante peut stresser l'utilisateur) et sur l'environnement dans lequel il se trouve. De même, l'environnement d'un usager peut impacter sur lui (exemple : un environnement trop bruyant le rend moins efficace) ainsi que sur sa tâche (exemple : exécution du modèle de la tâche peut changer selon la disponibilité des ressources environnementales). Quand il n'existe plus d'interactions entre les différentes dimensions, alors la situation engendrée est stable et unique à un instant t .

Remarque : Pour obtenir la stabilité à l'instant t , un processus du GSC, présenté dans la section VI.3, est appliqué : toutes les dimensions contextuelles sont alors « adaptées » en fonction des interactions qu'elles ont les unes avec les autres. Une telle adaptation est continuellement répétée jusqu'à obtenir une stabilité.

V.3. Les concepts de la situation

Comme nous l'avons indiqué dans la section précédente, une situation est une photographie stable du contexte à un instant t . Ainsi chaque situation engendrée dans le temps possède des concepts et des valeurs qui lui sont propres. Nous détaillons ces concepts dans ce qui suit :

Situation_t = { {dimensions contextuelles_t}, Validité, {état_n, {actions}}, @Situation_{t-1} }

Une situation est donc composée de quatre éléments. Premièrement, elle se compose d'un **ensemble de dimensions contextuelles** issu du contexte. Deuxièmement la **validité** qui indique si une situation est valide ou non en fonction des règles. Elle est étroitement liée à la validité de l'interprétation du contexte illustrée dans la section IV.2.1. Nous rappelons que la validité est la garantie d'une conjonction conforme de tous les éléments contextuels des dimensions contextuelles. Elle constitue donc un moyen de contrôler les situations avant leurs créations en utilisant des règles prédéfinies. Le troisième élément est l'**ensemble des actions** qui se sont déroulées durant un laps de temps correspondant à la situation. Enfin, toutes les situations qui sont associées doivent être liées. Pour cela, le quatrième élément de la situation est un **lien (pointeur) vers la situation** qui l'a précédée et dont la fin déclencha sa création.

Précédemment, nous avons mentionné le laps du temps correspondant à la situation. Celui-ci n'est pas constant et peut changer selon l'usage. En conséquence, les valeurs des éléments contextuels des différentes dimensions, en plus des actions, peuvent naturellement varier dans une même situation. Pour prendre en compte ces variations, la situation dans notre approche doit passer par plusieurs états avant d'être enregistrée puis abandonnée pour une nouvelle situation. En effet, pour une même situation, un nouvel état est créé chaque fois qu'il y a un changement dans les valeurs des éléments contextuels.

V.4. Exemple de situation

Prenons par exemple des dimensions contextuelles reflétant le cadre de ce travail qui s'intègre dans la recherche d'information dans un contexte métier identifié dans : la dimension usager, la dimension tâche métier et enfin la dimension environnement. La situation unique à l'instant t serait la photographie de l'activité de l'utilisateur effectuant sa tâche métier dans un environnement particulier. Nous pourrions également supprimer, dans la situation, tous les éléments contextuels qui n'ont aucun impact sur la réalisation de la tâche par l'utilisateur dans l'environnement actuel. Ceci permet de créer une situation unique et reflétant au plus près la réalité de l'exécution de la tâche qui sera exploitée par la suite par le système à des fins d'adaptations (Chaker, Chevalier, Soulé-Dupuy, & Tricot, 2010c).

Dans ce qui suit, et après avoir défini la situation comme étant l'interprétation stable du contexte, nous évoluons d'une façon logique vers l'approche de sa création. Pour cela, nous allons définir le GSC, puis présenter son architecture et enfin détailler la méthodologie associée permettant créer des situations dans le temps.

VI. Gestionnaire de situations contextuelles

Contrairement aux travaux existants qui visent à modéliser le contexte physique, l'originalité de notre approche réside dans la modélisation de la situation contextuelle, qui est la combinaison de toutes les dimensions contextuelles (filtrées et adaptées). En effet, comme nous l'avons présenté dans la section précédente, ces situations découlent de notre nouvelle définition du contexte et du processus de stabilisation de l'ensemble de ses dimensions. **C'est par ailleurs la dynamique de toutes les dimensions contextuelles confrontées les unes aux autres, puis adaptées à la situation qui donne un aperçu réaliste du contexte dans à un instant t .**

Partant de ce point, le besoin d'un modèle gérant les différentes dimensions contextuelles devient évident. Le gestionnaire doit ainsi faire la liaison entre tous les éléments contextuels d'un contexte en entrée pour créer la situation.

VI.1. Architecture du gestionnaire de situations contextuelles

Le GSC se veut générique de la même manière que la définition du contexte donnée dans la section III.1 et sur laquelle il repose. La généralité du gestionnaire réside dans le fait qu'il doit prendre en compte tout ce qui peut être qualifié de dimension contextuelle. Autrement dit, il ne doit pas exclure *a priori* des informations jugées contextuelles (Chaker, Chevalier, Soulé-Dupuy, & Tricot, 2011a).

Dans cette section nous présentons l'architecture globale de notre GSC. Nous pouvons voir sur la figure III.9 les trois parties principales qui composent le GSC :

- la première partie (numéro 1 sur la figure), que nous appelons **gestionnaire de données contextuelles** et qui sera détaillée par la suite, se concentre sur les données purement contextuelles en plus de la connaissance acquise par le GSC ;
- la deuxième partie (numéro 2 sur la figure), appelé **processus de mise en situation**, gère la création de situations en prenant en compte la connaissance (les règles) que possède le gestionnaire de données contextuelles ;
- la troisième et dernière partie du GSC (numéro 3 sur la figure) est appelé **processus d'extraction de règles**. Ce dernier se focalise sur les connaissances du GSC modélisées sous forme de règles afin de les faire évoluer dans le temps.

Sur le schéma, nous notons aussi la présence de **l'application sensible au contexte** reposant sur le GSC. Elle le sollicite pour la contextualisation de ses données et ses services. Ces applications communiquent au GSC, à travers des **capteurs applicatifs**, des informations contextuelles qui lui sont propres.

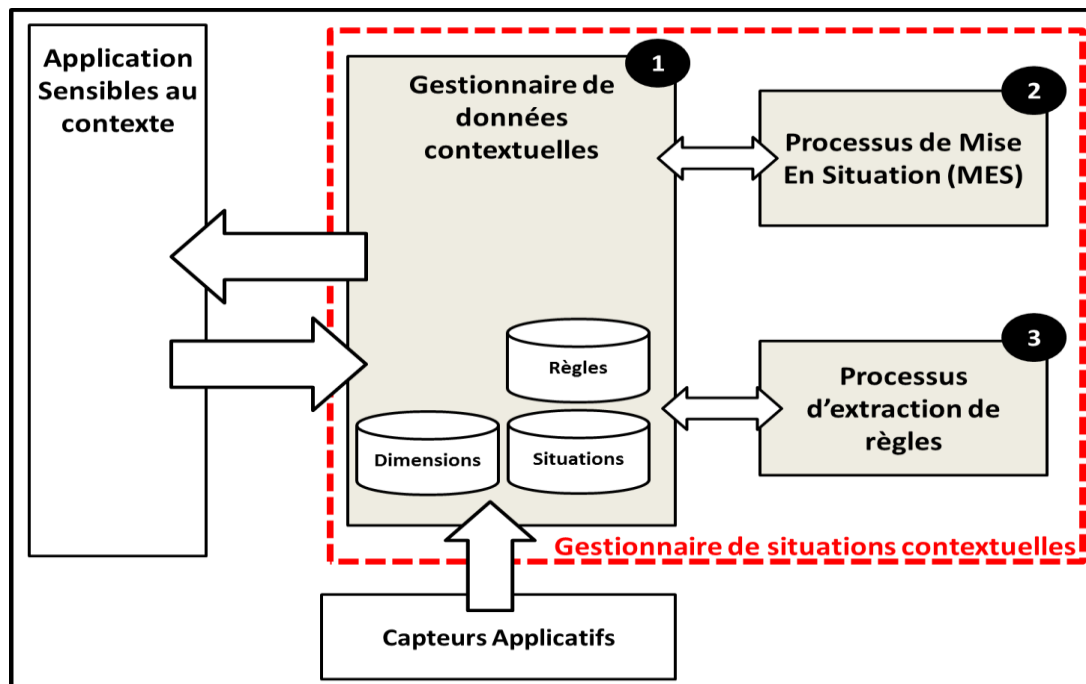


Figure III.9. Architecture globale du gestionnaire de situations contextuelles

VI.2. Le gestionnaire de données contextuelles

VI.2.1. Présentation

Le gestionnaire de données contextuelles (GDC) (cf. numéro 1 sur la Figure III.9) fait partie des trois composantes du GSC. Le GDC, comme son nom l'indique, possède en son sein toutes les données contextuelles. En effet, le GDC stocke l'ensemble des dimensions contextuelles utilisées pour la contextualisation des applications. En plus de cela, le GDC détient l'ensemble des règles qui représente la connaissance contextuelle. Le GDC possède également toutes les situations enregistrées dans le temps qui constituent l'historique du contexte. Enfin, le GDC est aussi connecté à différentes sources d'informations contextuelles applicatives (capteurs applicatifs). Ce type d'informations contextuelles proposées directement par les applications va permettre au GDC d'obtenir toutes les informations contextuelles supplémentaires et pertinentes aux applications s'appuyant sur notre GSC.

Le GDC s'occupe principalement de la communication entre les différentes composantes du GSC d'une part et des applications en quête d'informations contextuelles d'autre part. Le GDC joue le rôle d'une boîte noire en conservant séparément toutes les informations contextuelles propres aux différentes applications sensibles aux contextes. En d'autres termes, Le GDC est la composante qui permet à notre GSC de prendre en compte simultanément plusieurs contextes métier pour différentes applications. En effet, il préserve

distinctement toutes les dimensions contextuelles en plus de toutes les règles issues des différentes applications quels que soient leurs domaines ou leurs champs d'applications.

Il est important maintenant de définir les différents types de règles qui font partie du GDC. Celles-ci sont utilisées par le GSC pour modéliser les connaissances liées au contexte. Les objectifs du GSC sont amplement liés à ces types de règles (cf. section VI.5).

VI.2.2. La base des règles

Les règles serviront de support à la connaissance détenue par le gestionnaire. Elles sont le moyen qui permet au GSC de confronter les éléments contextuels les uns aux autres et de créer une situation unique à un instant t . Principalement, le GSC se fonde sur les règles pour « **adapter** » les éléments contextuels issus de chacune des différentes dimensions contextuelles en mettant à jour leurs valeurs ainsi que leurs fiabilités.

Les règles sont utilisées par le GSC dans trois buts :

- contrôler la validité des nouvelles situations. Le GSC, à travers un type de règles, contrôle la coexistence autorisée des éléments contextuels dans une même situation ;
- gérer l'adaptation des dimensions contextuelles en donnant un sens à l'interaction entre leurs différents éléments contextuels ;
- gérer le lien entre les actions et la situation permettant au GSC d'être proactif.

Ces trois objectifs d'utilisation des règles seront détaillés dans la section VI.5. Cependant, nous énumérerons dans la section ci-après les trois types de règles utilisées par notre GSC et la raison de cette catégorisation des règles.

a) Les trois types de règles

Nous présentons ici les trois types de règles utilisées par notre GSC **métier**, en sachant que le contexte métier du gestionnaire influence amplement cette catégorisation des règles. En effet, les deux premiers types de règles sont fournis directement par les experts métier du domaine d'application du GSC. Ces deux types de règles sont dits *règles métier légales* et *règles métier*, considérées comme la connaissance spécifique liée aux éléments contextuels qui affectent le règlement et le bon déroulement des activités de travail. Dans ce qui suit, nous donnerons une description plus détaillée de ces deux premiers types de règles :

- **les règles métier légales** : ce sont des règles dont la violation est strictement interdite par la législation d'une organisation, ou le règlement interne d'une entreprise par exemple. Ainsi, il est obligatoire de se conformer à ces règles au moment de la création d'une nouvelle situation. Les *règles métier légales* font partie de la connaissance détenue par le GSC et servent au contrôle de la validité des situations. La violation de ces règles peut être causée par la coexistence d'éléments du contexte non autorisée. Notons que les *règles métier légales* ne peuvent pas évoluer ;
- **les règles métier** : ce sont des règles également fournies par les experts métier du domaine afin d'améliorer les performances des activités ou d'établir des procédures et processus de travail. Contrairement au premier type de règle, l'exécution de celles-ci n'est pas régie par une loi interne du domaine. Notons, par ailleurs, que les experts peuvent observer les situations passées pour améliorer (mettre à jour) ces *règles métier*.

Le troisième type de règles non fournies par les experts correspond aux **règles inférées** (ou connaissances générées liées au contexte). Ces règles sont extraites de l'historique des situations valides. Elles permettent au système d'identifier automatiquement de nouvelles connaissances induites par les interactions pertinentes entre les différents éléments contextuels appartenant aux situations passées (Chaker, Chevalier, Soulé-Dupuy, & Tricot, 2011b). Nous n'allons pas détailler ici ce processus d'extraction des *règles inférées* parce qu'il fait par la suite l'objet d'une section.

Nous allons dans ce qui suit définir les règles utilisées par notre gestionnaire de situations contextuelles métier.

b) La définition des règles

Les règles utilisées dans notre travail ont une même et unique définition malgré leur différence de type. Toutes ces règles sont une implication de la forme X implique Y, notée $X \rightarrow Y$, où X est une conjonction d'éléments contextuels provenant des différentes dimensions contextuelles et Y est un élément contextuel unique qui n'est pas présent dans X ou une fonction permettant de modifier la structure de la dimension contextuelle (suppression d'un ou plusieurs éléments). Enfin, dans notre approche, chaque règle possède une **Priorité** (valeur de type réel) transmise par les experts (pour les deux premiers types) ou calculée par le GSC (pour les *règles inférées*).

Pour mieux comprendre la définition des règles, voici un exemple abstrait d'une règle :

Règle : $Dimension_1.Element_1=valeur_1$ **et** $Dimension_2.Element_2=valeur_2$

→ $Dimension_1.Element_3=valeur_3$ (**Priorité** = p)

Nous avons pris dans cet exemple deux dimensions contextuelles la $Dimension_1$ et la $Dimension_2$. Ces deux dimensions sont composées de trois éléments contextuels différents. Le GSC interprète cette règle comme une connaissance contextuelle qui lui permet d'attribuer la valeur « $valeur_3$ » à l'élément contextuel $Element_3$ issu de $Dimension_1$, si les conditions de cette règle sont réunies. C'est-à-dire que la règle est satisfaite si l' $Element_1$ de $Dimension_1$ a la valeur « $valeur_1$ » et en même temps si l' $Element_2$ de $Dimension_2$ a la valeur « $valeur_2$ ».

Après avoir défini les règles utilisées par notre GSC, nous présentons dans ce qui suit les règles d'association. Celles-ci s'avèrent particulièrement adaptées à la base de connaissance de notre GSC métier. Certes, la similarité entre notre définition des règles et la définition des règles d'association proposée par Agrawal et ses collaborateurs (Agrawal, Imielinski, & Swami, 1993) renforce notre choix envers cette description des données qui permet surtout d'extraire des corrélations tenant compte ou non d'une notion d'ordre.

Une règle d'association peut être définie formellement d'après (B. Liu, 2007) :

Définition : Soit $I = \{i_1, i_2, \dots, i_m\}$ un ensemble d'items. Soit $T = \{t_1, t_2, \dots, t_n\}$ un ensemble de transactions, telles que t_i soit un sous-ensemble de I (i.e. $t_i \subseteq I$). Une règle d'association s'exprime sous la forme :

$$X \rightarrow Y, \text{ où } X \in T, Y \in T \text{ et } X \cap Y = \emptyset$$

Nous présentons dans la section suivante le processus de mise en situation. Ce processus fait partie des trois composantes du GSC et assure l'interprétation du contexte et la création des différentes situations dans le temps.

VI.3. Le processus de Mise En Situation

Le processus de mise en situation (MES) (cf. numéro 2 sur la Figure III.9) est la partie la plus importante du GSC en termes d'objectifs atteints. Le processus MES gère la mise en

relation des différentes dimensions contextuelles pour atteindre l'**interprétation stable** du contexte initial et ainsi créer des situations uniques dans le temps. Le MES permet soit de conserver les éléments contextuels des différentes dimensions de départ dans la situation sans les modifier, soit de les adapter en fonction des différentes interactions qui puissent exister entre eux. Nous assumons donc qu'une dimension contextuelle (i.e. ses éléments contextuels) est adaptée selon le reste des dimensions contextuelles. Autrement dit, une dimension contextuelle est adaptée selon son propre contexte (i.e. les autres dimensions restantes). Cette définition de l'adaptation d'un objet contextuel est en phase avec la vision du contexte donnée par Brézillon et ses collaborateurs (Brézillon, Pasquier, & Pomerol, 2002) (cf. section III.1). Par conséquent, nous admettons qu'une dimension contextuelle dépend de son contexte et doit s'y adapter s'il influe sur elle (cf. figure III.10).

Pour réaliser l'adaptation d'une dimension en fonction des autres, le processus MES utilise les types de règles adéquates (cf. section VI.2.2) pour décrire les diverses transformations à appliquer comme l'illustre la figure III.10.

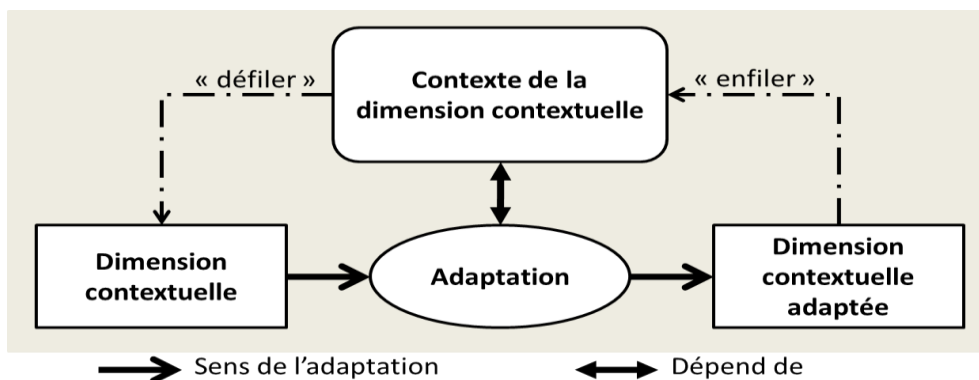


Figure III.10. Le principe de l'adaptation d'une dimension contextuelle par le MES

Le processus MES doit garantir également la **stabilité** de la situation. En effet, celle-ci réside dans le résultat de la dynamique des interactions entre les différents éléments contextuels présents au moment de la création d'une nouvelle situation. Le processus MES applique successivement les règles sur les dimensions jusqu'à ce qu'un point de stabilité soit trouvé, c'est ce que nous appelons **la situation**. La stabilité est atteinte quand il n'y a plus de règles à appliquer, et peut être issue après un grand nombre d'itérations (cf. section VII.5).

Le processus MES est générique, puisqu'il prend en considération toutes sortes de dimensions contextuelles à condition que le GDC lui fournisse la base de règles lui permettant de les contextualiser (cf. Figure III.11).

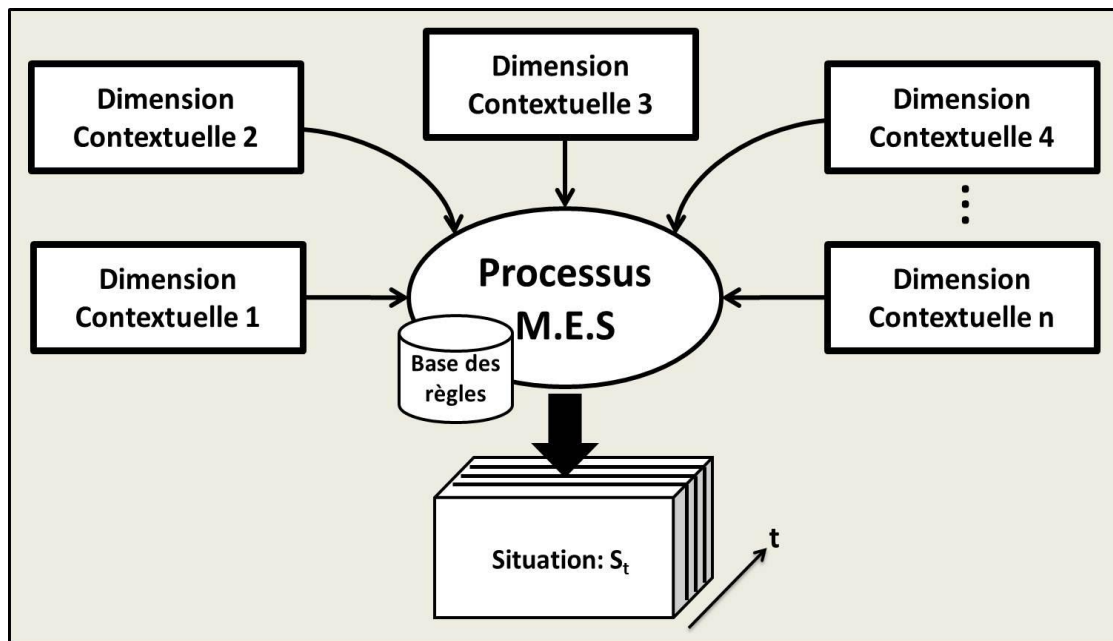


Figure III.11. Création de situations dans le temps

Nous remarquons que le processus MES ainsi que la toute la démarche de création de situation sont détaillés dans la section VII.

VI.4. Le processus d'extraction de règles

Le GSC ou plus précisément la connaissance qu'il possède sur les contextes doit pouvoir évoluer avec le temps, c'est-à-dire, que l'ensemble de règles exploitées par le GSC doit évoluer pour prendre en compte l'historique du contexte. Pour cela, nous appliquons un procédé d'extraction de règles (cf. numéro 3 sur la Figure III.9) sur l'historique des situations valides enregistrées. Notre intérêt pour les situations passées est clairement justifié étant donné que dans ces situations nous avons toutes les informations sur les éléments du système. De plus, l'interaction des différents éléments contextuels au moment de l'adaptation peut elle-même causer un changement dans les valeurs de départ (cf. figure III.6) ou dans les modèles initiaux des dimensions (cf. figure III.5). En effet, apprendre des situations passées ne peut que donner à notre GSC une vue réaliste des interactions qui ont réellement eu lieu entre des éléments contextuels spécifiques pour créer une situation. C'est donc au fil du temps que le système devient de plus en plus efficace et pertinent. C'est-à-dire que l'adaptation des éléments contextuels devient de plus en plus représentative de la réalité, générant ainsi des situations fidèles à leur contexte, et par conséquent aux interactions de toutes ses dimensions.

Il faut cependant préciser que le processus d'extraction de règles se concentre uniquement sur les situations valides. Certes, les situations invalides ne sont utilisées que pour améliorer *les règles métier légales* et elles ne sont pas exploitées pour extraire *les règles inférées*. Finalement, les situations non valides sont filtrées avant le traitement d'extraction (cf. section VIII.2).

De la sorte, le processus d'extraction de règles propose au GSC de nouvelles règles qui vont servir à alimenter la base initiale des règles. Le schéma sur la Figure III.12 permet d'illustrer d'une manière simplifiée le fonctionnement du processus d'extraction de règles. Ce dernier ainsi que toute la démarche d'extraction de règles sont détaillés dans la section VIII.

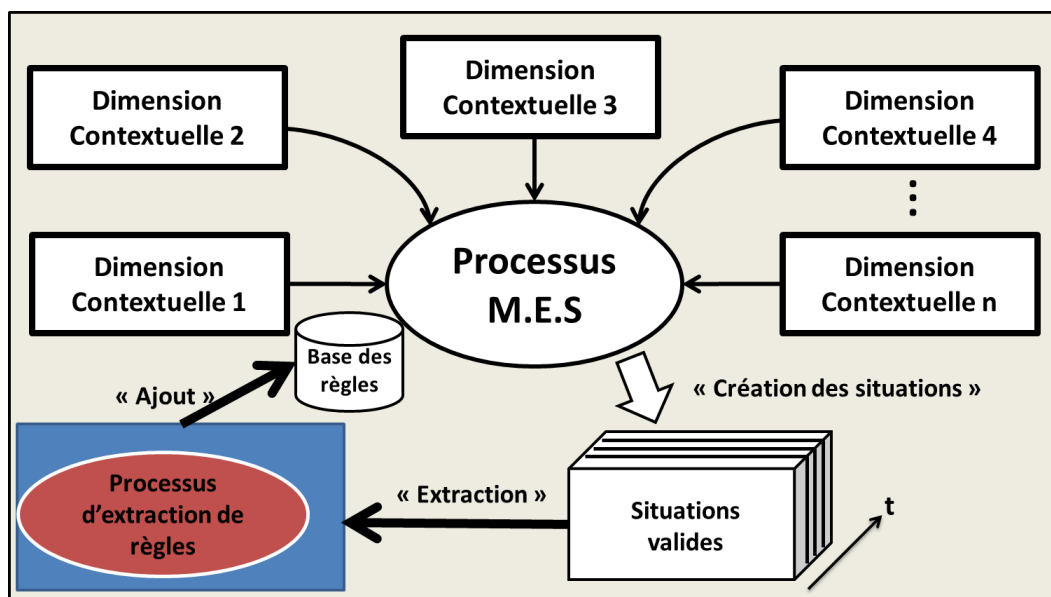


Figure III.12. Le processus d'extraction de règles

VI.5. Les engagements du GSC envers les applications

En se basant sur les trois types de règles présentés dans la section précédente (cf. section VI.2.2), nous présentons les quatre objectifs fixés par notre GSC envers les applications sensibles au contexte.

VI.5.1. Vérification des situations valides

L'objectif premier de notre GSC est la vérification de la validité des situations. Une situation valide est une situation qui ne contredit pas les *règles métier légales*. Pour remplir cet objectif, l'application utilisant notre GSC envoie les informations contextuelles nécessaires à la création de la situation en plus des actions entreprises par l'utilisateur.

Le gestionnaire de situations contextuelles alerte l'application si la situation n'est pas valide : les éléments contextuels des différentes dimensions contextuelles ne peuvent pas être mis en situation à cause de la violation d'une ou de plusieurs *règles métier légales*. Par exemple, en maintenance aéronautique, les techniciens ayant une qualification « catégorie A » (les moins qualifiés) ne peuvent pas faire des tâches exigeantes ni travailler sur le tarmac...

Nous rappelons que les situations dites invalides sont stockées par le GSC afin de faire remonter l'information aux experts du domaine. Cela permettra de faire évoluer le SI en mettant à jour la base des règles légales. Nous pouvons donner l'exemple d'un circuit d'information découvert par l'usage récurrent d'un document « interdit » pour une catégorie d'utilisateur.

VI.5.2. Comblant le manque d'information dans le contexte

Le deuxième objectif du GSC est d'utiliser les règles pour combler un manque d'informations contextuelles. Nous rappelons que les deux types de règles utilisées pour cet objectif sont les deux types de règles dites non légales. Nous précisons que le GSC garde en mémoire toute l'activité de la réalisation d'une tâche entamée. C'est-à-dire, il conserve l'évolution des situations qui s'enchaînent et qui sont liées par la même tâche initiale (Chaker, Chevalier, Soulé-Dupuy, & Tricot, 2010d).

Le manque d'informations contextuelles peut survenir dans les deux cas suivants :

- au moment de l'initialisation du système, c'est-à-dire, au moment de la création de la première situation de l'activité, il se peut qu'il manque d'informations contextuelles du fait par exemple d'une défaillance d'un des capteurs. Dans ce cas, GSC utilise les *règles métier* puis les *règles inférées* pour prédire les informations manquantes, pour ensuite créer la situation ;
- entre une situation $t-1$ et une situation t (ou deux états de la même situation), le GSC récupère les informations contextuelles de la situation précédente (ou de l'état précédent de la situation courante) en plus des éléments contextuels mis à jour par l'application pour créer la nouvelle situation t (un nouvel état de la situation). Le GSC utilise donc les règles pour adapter les éléments contextuels et mesurer l'impact des informations contextuelles mises à jour sur le reste des éléments contextuels. Enfin, la situation est générée par le GSC.

VI.5.3. Proactivité/anticipation

Le troisième objectif du GSC est d'anticiper les événements et de les proposer automatiquement à l'application. Dans ce cas de figure, l'application en question envoie les actions des usagers au GSC et le gestionnaire peut à son tour recommander des actions à l'application. En effet, toutes les situations stockées conservent les actions effectuées par les utilisateurs de l'application (cf. section V.3). Cela permet au GSC de les regrouper en utilisant des critères de similarité (raisonnement par cas sur les situations). Ainsi, le GSC propose les actions les plus récurrentes pour des situations relativement similaires. Par exemple, pour une situation donnée, le GSC recommande un document ignoré par l'utilisateur et dont la lecture est présente dans le log des actions de toutes les autres situations similaires.

VI.5.4. Fournir les éléments contextuels

L'application peut, à tout moment, demander au GSC les éléments contextuels actuels à des fins d'adaptation. C'est-à-dire les valeurs actuelles des différents éléments contextuels prises dans la situation courante. C'est pourquoi le GSC doit mettre à la disposition de l'application non seulement les valeurs des éléments contextuels, mais encore **la fiabilité** de leur valeur respective (cf. section III.2). Précisons que l'origine des fiabilités est issue des règles utilisées pour obtenir la valeur de ces éléments contextuels.

Nous venons de présenter les engagements du GSC envers les applications sensibles au contexte. Nous ne pouvons pas garantir l'aboutissement de ces objectifs si les informations contextuelles ne sont pas fidèles à la réalité. Autrement dit, et pour faire le lien avec ce que nous avons présenté dans les sections précédentes, le seul garant de l'atteinte de ces objectifs est une situation **la plus fidèle possible à la réalité**.

VI.6. Synthèse

En générant les situations, nous nous attardons sur tous les éléments contextuels à l'issue d'une activité et sur tous les changements qui en découlent (adaptations). Dans ces situations, notre GSC injecte des connaissances supplémentaires, en plus de l'ensemble des éléments contextuels issus du contexte initial. De cette manière, nous savons exactement ce qui s'est passé dans notre situation et également les transformations ayant eu lieu dans le contexte initial. Nous pouvons prendre en compte les relations entre les éléments contextuels dans l'ensemble des dimensions. Ce sont ces relations qui peuvent par la suite aboutir à la

génération de nouvelles règles. Ainsi, notre approche offre des connaissances issues des régularités constatées dans les situations stockées. Ce sont ces connaissances mêmes que nous ne pouvons pas détecter directement au sein du contexte et qui pourraient être déduites uniquement à partir des relations entre les éléments contextuels dans diverses dimensions contextuelles.

En conclusion, nous remarquons que le GSC, à travers ses trois composantes (cf. figure III.9), forme une chaîne fermée qui caractérise notre approche. La dernière maille de la chaîne se focalise sur la gestion des règles et cela à travers le processus d'extraction de règles. Ainsi, la caractéristique majeure de notre proposition réside dans le fait que le GSC peut améliorer ses connaissances par les règles extraites des situations passées. Par conséquent, les nouvelles connaissances contextuelles acquises par le gestionnaire améliorent *a posteriori* la génération des situations futures. Cela signifie que notre ensemble de relations entre les éléments contextuels des différentes dimensions se développe au fil du temps et que le gestionnaire devient de plus en plus efficace dans la création des situations (plus fidèles à la réalité).

Nous allons détailler dans le reste du chapitre les deux approches proposées par le GSC. La première se consacre à la démarche de création de situation, tandis que la seconde approche se focalise sur la fouille de nouvelles règles.

VII. Démarche de création de la situation

Dans cette section, nous allons nous focaliser sur la démarche complète de la création de situations (Chaker et al., 2011a). L'interprétation présentée dans la section IV n'est que le support commun qui regroupe tous les éléments contextuels issus des différentes dimensions contextuelles d'un contexte initial sur lequel se base la situation. En d'autres termes, une formalisation de l'interprétation du contexte à laquelle nous ajoutons les autres concepts présentés dans la section V.3, à savoir la validité et l'ensemble des actions, forment la situation. La génération de la situation est fondée sur une méthodologie originale que nous aborderons dans ce chapitre.

Plus généralement, nous avons expliqué tout au long de ce chapitre que la situation est exploitée par les applications sensibles au contexte afin de contextualiser soit leurs traitements soit les données. Pour cela, l'application définit les dimensions contextuelles dont elle

souhaite la contextualisation. Ce dernier point garantit la généralité du GSC, qui ne se limite pas à des dimensions contextuelles prédéfinies.

Plus précisément, la tâche de création des situations est partagée entre le GDC et le processus MES (cf. figure III.9). Nous présentons dans ce qui suit les rôles respectifs joués par ces deux composantes du GSC pour générer les situations. En effet, les deux composantes en question communiquent non seulement entre elles, mais communiquent aussi avec l'application afin de lui proposer la situation voulue.

VII.1. Les informations contextuelles nécessaires à la création des situations

VII.1.1. Les informations contextuelles transitant par le GDC

Le GDC est une des deux composantes du GSC qui ont un rôle dans le processus de création des situations. Nous rappelons que le GDC gère en premier lieu de toutes les données contextuelles du gestionnaire.

Avant de détailler les informations contextuelles nécessaires à la création d'une situation à l'instant t , il est primordial de retourner à t_0 pour présenter l'étape d'**initialisation** du GSC. En effet, avant toute demande de création de situations, l'application doit transmettre les informations contextuelles spécifiques de son domaine métier en plus des dimensions contextuelles qu'elle traite. Le GDC récupère ces informations et les utilisera pour toutes demandes ultérieures de contextualisation. Ainsi, comme le schématise la figure III.13, trois informations contextuelles sont communiquées par l'application au GDC. Premièrement, l'ensemble des règles spécifiques à l'application qui vont servir au processus de contextualisation (cf. numéro 1 sur la figure III.13). Evidemment, l'application ne détient pas le troisième type de règles (*règles inférées*) et ne transmet donc aux GDC que les *règles métier légales* et les *règles métier*. Deuxièmement, l'application envoie au GDC la structure et les valeurs de toutes les dimensions contextuelles (cf. numéro 2 sur la figure III.13). De la sorte, le GSC possède dans ses connaissances la structure hiérarchique des différentes dimensions contextuelles de l'application ainsi que les valeurs des éléments qui les composent. Enfin, le troisième élément envoyé par l'application y représente une information contextuelle très importante liée au domaine métier. Cette dernière est une liste d'éléments contextuels prédéfinis par les experts du domaine déclenchant, s'ils sont mis à jour, un changement de situation (cf. numéro 3 sur la figure III.13). Autrement dit, chaque fois que l'application communique un ou plusieurs éléments contextuels de cette liste (par exemple,

l'identifiant d'une nouvelle tâche), le GSC sait qu'il doit terminer la situation courante pour ensuite créer une nouvelle situation qui se basera sur les dernières informations contextuelles reçues. Par ailleurs, pour tous les autres éléments contextuels mis à jour par l'application, le GSC crée un nouvel **état** pour la même situation afin de prendre en compte les changements demandés.

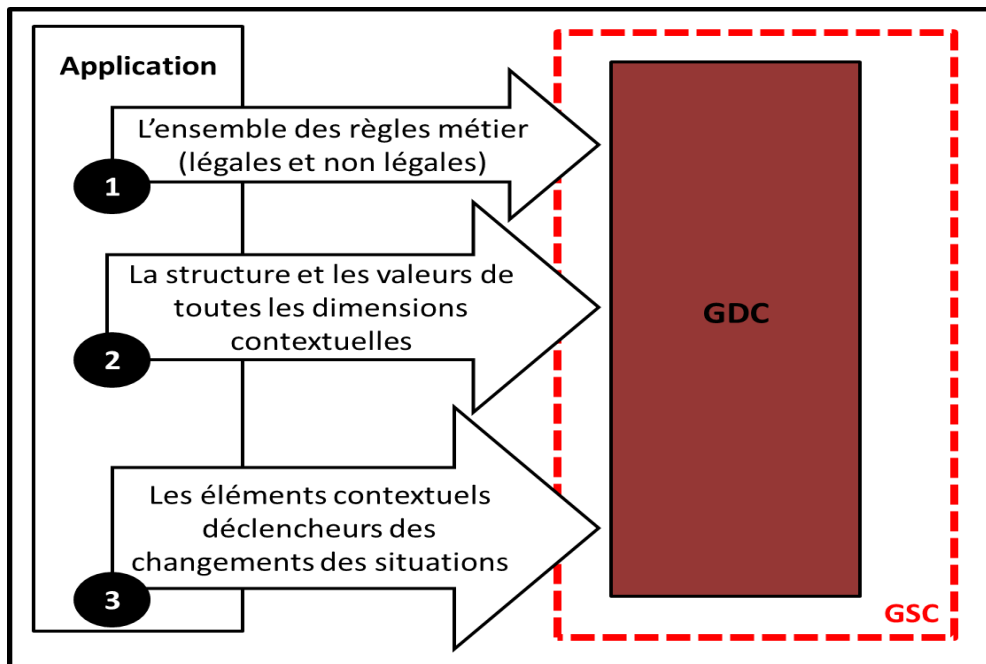


Figure III.13. Les informations communiquées par une application pour l'initialisation du GDC

La figure III.14 schématise le rôle du GDC dans la communication entre le processus MES et l'application afin de créer une situation (ou un nouvel état de la situation). Pour expliciter le flux de données entre les différentes composantes, nous allons suivre la démarche chronologique de la création d'une situation donnée en allant de t_1 à t_3 (cf. figure III.14). Ainsi, c'est l'application qui enclenche cette démarche en envoyant **les valeurs des dimensions contextuelles** sur lesquelles la contextualisation est souhaitée (cf. à l'instant t_1 sur la figure III.14). Autrement dit, l'application communique les instances des dimensions contextuelles qu'elle voudrait mettre en situation. Par conséquent, le GDC retransmet au processus MES le besoin d'une mise en situation d'un ensemble de valeurs de dimensions contextuelles (cf. à l'instant t_2 sur la figure III.14). Plus précisément, le GDC donne au processus MES toutes les informations contextuelles nécessaires à la création de cette situation particulière. Ainsi, toutes **les valeurs des éléments contextuels** issus des dimensions en question sont transmises au processus MES en plus de **la base des règles** spécifique à cette application. En plus des

dimensions contextuelles et les règles, le GDC administre **les informations contextuelles applicatives**. Celles-ci sont collectées continuellement par le GDC au cours de la création de la situation et ainsi transmises au processus de MES (l'intervalle entre t_2 et t_3 , comme nous le schématisons sur la figure III.14).

Enfin, le GDC propose à l'application un premier état de la situation en plus des actions recommandées qui serviront à l'adaptation (cf. à l'instant t_3 sur la figure III.14). Rappelons qu'à chaque début d'une nouvelle situation, le GDC fournit l'état de la situation à l'application. Ensuite, comme nous l'avons expliqué plus haut, si l'application ne signale pas préalablement au GDC la fin de la situation courante (pas de changements dans les valeurs des éléments contextuels de la liste prédéfinie des déclencheurs). Dans ce cas, à moins que l'application ne mette fin à cette situation, le GDC continue à réclamer au processus MES des mises en situation quand les informations contextuelles applicatives sont mises à jour, i.e. création de nouveaux états de la situation courante.

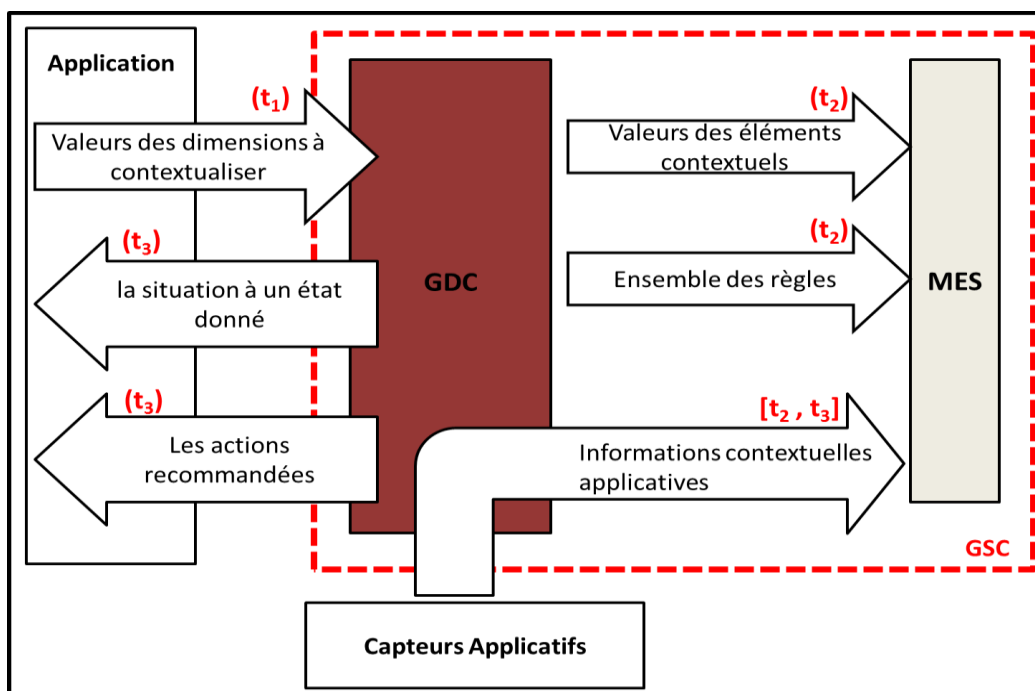


Figure III.14. Les flux de données provenant du GDC pour la création de situations

VII.1.2. Les informations contextuelles transitant par le MES

Le MES est la deuxième composante du GSC qui joue un rôle dans la création des situations ; et il en est de même la composante la plus importante. Le processus MES a deux

tâches principales dans le procédé de création des situations. Ainsi, ce dernier contrôle en premier lieu la validité des situations. Par conséquent, deux cas de figure peuvent apparaître :

Cas 1. Le contenu des dimensions contextuelles à mettre en situation (l'ensemble des valeurs des éléments contextuels) ne peut pas être réuni dans une même situation. Dans ce cas, le processus MES averti le GDC de l'invalidité de la situation de telle sorte que la mise en situation ne soit pas possible.

Cas 2. Le contenu des dimensions contextuelles peut être mis en situation. Dans ce deuxième cas de figure la situation est considérée valide et donc le processus MES peut commencer sa deuxième tâche pour générer la situation.

La figure III.15 illustre les informations contextuelles *fournies par le processus MES au GDC* qui le sollicite. En effet, le processus MES envoie au GDC, en plus de **la validité de la situation**, la situation qui peut être soit **un état de la situation**, soit **la situation finale**. Nous remarquons par ailleurs que le processus MES est continuellement informé des mises à jour dans **les informations contextuelles applicatives** qui agissent sur le changement des états des situations.

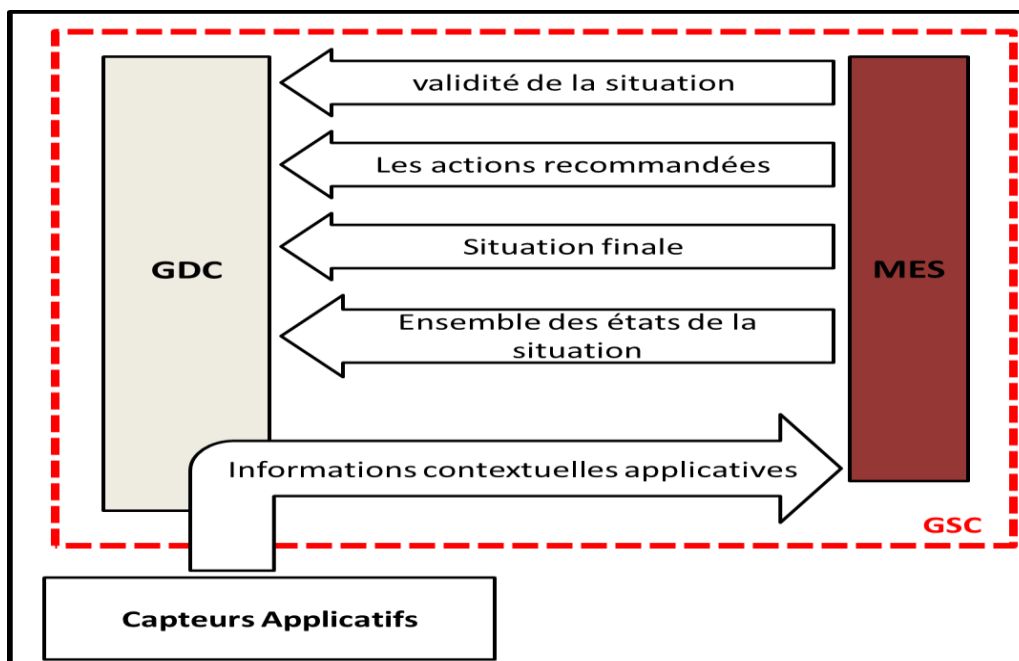


Figure III.15. Les flux de données provenant du processus MES pour la création de situations

VII.2. Fonctionnement du processus MES

Dans la section précédente, nous avons présenté les deux composantes impliquées dans la création des situations. Ainsi, nous nous sommes focalisés sur les informations contextuelles qui circulent entre elles et qui serviront à la création des situations. Les informations en question concernent dans un premier temps la relation entre l'application demandant la situation contextuelle et le GSC. Également, elles portent aussi sur l'échange d'informations qui circulent entre le processus MES et le GDC.

Pour aller plus loin dans la description de l'approche de création des situations, nous allons nous focaliser dans cette section sur le processus MES. Pour cela, nous exposons dans ce qui suit notre approche de mise en situation. Plus précisément, nous expliquons les étapes par lesquelles passe le processus MES pour la création de la situation, notamment, l'adaptation des différentes dimensions contextuelles ainsi que le principe de stabilisation du contexte.

VII.3. Algorithme général du MES

Dans cette section nous présentons l'algorithme principal de la création des situations. Cet algorithme reste général et sera détaillé par un autre algorithme à la section V.5.2.

L'algorithme général du processus MES permettant la création des situations dans le temps est présenté dans ce qui suit comme étant l'algorithme 1. Pour chaque situation à créer, le processus MES vérifie impérativement la validité de la situation. Le processus MES vérifie la totalité des *règles métier légales* par rapport au contexte initial. Par conséquent, dans le cas où une de ces règles n'est pas vérifiée, le processus de mise en situation est stoppé et la situation est stockée dans la base des situations invalides. Dans le cas où la coexistence de tous les éléments contextuels de départ est autorisée par les *règles métier légales*, la situation est considérée valide et le processus MES poursuit la création de la situation. Ainsi, la **stabilisation du contexte** est lancée sur les dimensions composant le contexte initial (cf. l'algorithme 2 de la section VII.5.2).

Algorithme 1 : Processus MES

Données : ContexteInitial : ensemble des valeurs des différentes dimensions contextuelles

Règles : la base des règles;

Résultat : Situation : la situation unique à l'instant t ;

Début

Vérifier toutes les règles métier légales;

Tant que Situation.validité \leftarrow vrai **et** Règles contient règlesLégales **faire**

Si règlesLégales est non vérifiée **alors**

 Situation.validité \leftarrow faux;

Fin Si;

Fin tant que;

Si Situation est valide **alors**

Stabilise Contexte; – (cf. algorithme 2)

Fin Si;

Fin

Pendant le cycle d'adaptation, le processus MES se base sur les règles fournies par le GDC pour adapter dans un premier temps une dimension contextuelle par rapport au reste du contexte initial. Par la suite, il entame une fonction itérative d'adaptation de toutes les dimensions. C'est ce que nous appelons la stabilisation du contexte (cf. algorithme 2).

De ce fait, nous détaillons dans ce qui suit l'approche d'adaptation des dimensions contextuelles, car elle est la base de la stabilisation, en l'appuyant avec un exemple illustratif. Ensuite, nous présentons le cycle d'adaptation et en particulier le principe de stabilisation du contexte.

VII.4. Adaptation des dimensions contextuelles

VII.4.1. Principe

L'adaptation d'une dimension contextuelle est la transformation provoquée par l'application d'une règle sur celle-ci. Cette transformation touche soit les valeurs des éléments contextuels soit les modèles des dimensions contextuelles de départ (cf. Section IV.1). Ainsi comme l'illustre l'exemple pris sur la figure III.16, la dimension contextuelle est adaptée en utilisant une règle. Il en résulte une nouvelle dimension contextuelle (adaptée) avec l'apparition d'une nouvelle valeur de son élément contextuel $E2$. La règle qui a déclenché cette adaptation fait partie de l'ensemble des règles du contexte initial qui, si elle est vérifiée,

déclenche cette adaptation. Nous soulignons qu'une règle est utilisée pour l'adaptation d'un élément contextuel si et seulement si sa **priorité** est strictement supérieure à la **fiabilité** de cet élément.

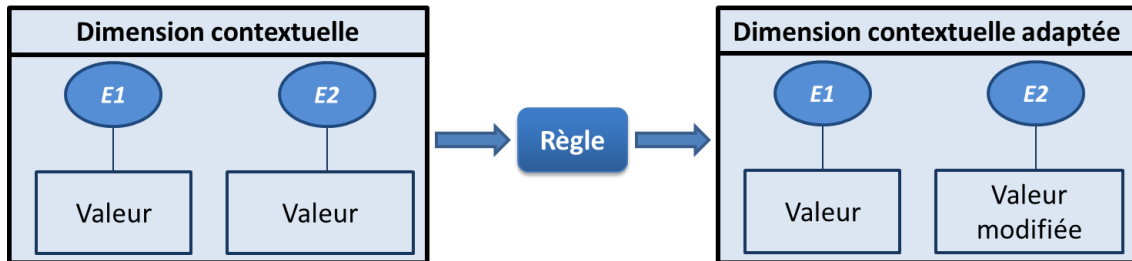


Figure III.16. Le principe de l'adaptation d'une dimension contextuelle

VII.4.2. Exemple illustratif

Nous allons, dans le reste de ce chapitre, prendre un exemple de contexte initial illustré par la figure III.17. Le GDC fournit au processus MES cet ensemble de dimensions contextuelles et leurs éléments contextuels respectifs. Dans notre exemple, ce contexte est composé des trois dimensions contextuelles suivantes :

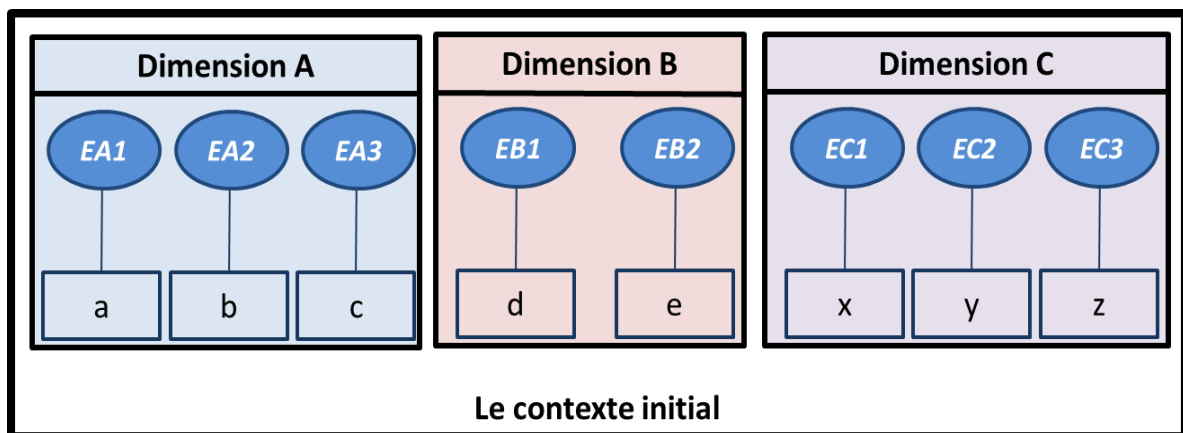


Figure III.17. Exemple d'un contexte composé de trois dimensions

Les règles utilisées pour l'adaptation sont classées selon leurs priorités comme l'illustre la figure III.18.

- R1** : C.EC3 = « z' » \wedge A.EA2 = « y » \rightarrow B.EB2 = « e' » Priorité : 0,9
- R2** : B.EB1 = « d' » \wedge C.EC1 = « x » \rightarrow A.EA3 = « c' » Priorité : 0,9
- R3** : A.EA3 = « c' » \wedge B.EB2 = « e » \rightarrow C.EC3 = « z' » Priorité : 0,7
- R4** : A.EA1 = « a » \wedge C.EC2 = « y » \rightarrow B.EB1 = « d' » Priorité : 0,6

Figure III.18. Un exemple simplifié d'une base de règles

Nous remarquons que les quatre règles proposées dans cet exemple permettent de modifier uniquement les valeurs des éléments contextuels des dimensions contextuelles et non pas leurs modèles initiaux.

Par cet exemple nous voulons illustrer dans un premier temps le cycle d'adaptation de l'ensemble des dimensions contextuelles puis nous focaliser sur la stabilisation du contexte qui en découle. Ainsi dans l'exemple ci-dessus, la mise en situation repose sur les quatre règles pour lesquelles une priorité est donnée, permettant de sélectionner les règles devant être appliquées si les conditions s'y prêtent. Dans notre cas, seule la règle **R4** sera appliquée car elle est la seule dont les conditions sont satisfaites. Par conséquent, l'adaptation de la dimension contextuelle *B* donne une nouvelle dimension contextuelle *B'* avec une nouvelle valeur « d' » à la place de « d » pour l'élément contextuel *EB1* (cf. figure III.19).

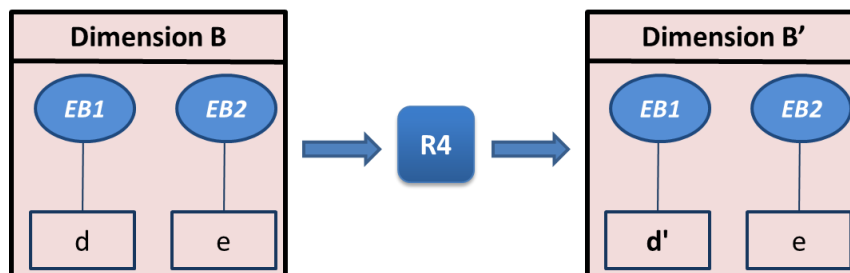


Figure III.19. L'adaptation de la dimension contextuelle B

Ainsi, le processus MES se fonde sur le principe de l'adaptation pour créer des situations uniques dans le temps qui reflètent la réalité. Plus précisément, la dimension adaptée *B'* remplace *B* avec le reste des dimensions contextuelles et les règles sont donc appliquées successivement sur ces dimensions au fur et à mesure du processus de stabilisation du contexte. L'approche de cette stabilisation est détaillée dans la section suivante.

VII.5. Stabilisation du contexte

La stabilisation du contexte est comme nous venons de l'introduire, un cycle basé sur d'adaptation des dimensions contextuelles les unes par rapport aux autres. Ainsi, pour fournir plus de détail sur la stabilisation du contexte, nous exposons dans un premier temps ce principe sur lequel se base le processus MES pour la création des situations. Nous présentons ensuite l'algorithme de stabilisation du contexte et enfin, nous illustrons notre approche par un exemple.

VII.5.1. Principe

Le fonctionnement du processus MES repose sur un principe de stabilisation qui est en phase avec notre vision du contexte. En effet chaque dimension du contexte possède, elle aussi, son propre contexte qui est le reste des dimensions du contexte. Nous pouvons donc souligner le fait que la construction d'une situation est une démarche itérative. Ainsi, nous ne pouvons pas considérer une des dimensions du contexte initial séparément. En effet, la stabilisation du contexte donne naissance à la dynamique des interactions entre les éléments contextuels des différentes dimensions contextuelles. Cette dynamique est un des points forts de notre approche pour la construction des situations.

Le principe de contextualisation est d'adapter une des dimensions du contexte initial par rapport aux restantes. Pour obtenir une photographie complète, cette adaptation doit être réalisée successivement pour chacune des différentes dimensions. Ces adaptations successives définissent alors ce que nous appelons la stabilité du contexte. En d'autres termes les règles sont appliquées sur les dimensions (adaptées et initiales) jusqu'à ce qu'un point de stabilité soit atteint : c'est la situation unique à un instant t . Ainsi, une situation peut être générée après plusieurs itérations d'adaptation des dimensions (cycle d'adaptation).

VII.5.2. L'algorithme de stabilisation du contexte

La stabilisation du contexte est donc la fonction de construction de la situation et son algorithme est donné par l'algorithme 2. Ce dernier implémente la fonction itérative qui est utilisée par le processus MES pour la stabilisation du contexte et ainsi la génération de la situation. Cet algorithme a été proposé dans (Chaker et al., 2010a).

Algorithme 2 : Stabilise Contexte

Données: ContexteInitial : ensemble des valeurs des différentes dimensions contextuelles
Règles : la base des règles;

Résultat: Situation : la situation unique à l'instant t

Début

FileDeContexte : la file contenant l'ensemble des dimensions contextuelles formant le contexte initial;

FileTampon : la file contenant les dimensions contextuelles adaptées (ou initiales) durant chaque itération;

Const N <Entier > = 1000 ; un seuil pour arrêter les itérations;

FileDeContexte \leftarrow copier (ContexteInitial) ; -- copier toutes les dimensions contextuelles dans la file;

ContexteStable \leftarrow Faux;

Tant que (Contexte.Stable \leftarrow Faux et nbIterations < N) **faire**

 ContexteStable \leftarrow Vrai;

Pour chaque DimensionContextuelle \in FileDeContexte **faire**

 DimensionContextuelleActuelle \leftarrow défiler (FileDeContexte);

 règlesFiltrées \leftarrow filtrer (DimensionContextuelleActuelle, FileDeContexte, Règles);

Si (cardinal(règlesFiltrées) > 0) **alors**

 DimensionContextuelle _{t} \leftarrow appliquer (DimensionContextuelleActuelle, règlesFiltrées);

 ContexteStable \leftarrow Faux;

 FileDeContexte \leftarrow enfiler (DimensionContextuelleActuelle);

 FileTampon \leftarrow enfiler (DimensionContextuelle _{t});

Sinon

 FileTampon \leftarrow enfiler (DimensionContextuelleActuelle);

 FileDeContexte \leftarrow enfiler (DimensionContextuelleActuelle);

Fin Si;

Fin Pour;

 nbIterations ++;

 FileDeContexte \leftarrow FileTampon;

Fin Tant que;

Retourner Situation \leftarrow FileDeContexte;

Fin

VII.5.3. Illustration de l'algorithme de la stabilisation

Nous allons revenir à l'exemple précédent pour illustrer l'algorithme de stabilisation du contexte et ainsi la mise en situation du processus MES.

Nous rappelons que seule la règle R4 a été appliquée à la fin de la première itération du cycle d'adaptation. Il en résulte une nouvelle dimension contextuelle B' (cf. figure III.19). Par conséquent, pour obtenir la situation, le processus MES continue la stabilisation du contexte.

En effet, la dimension contextuelle adaptée B' remplace la dimension initiale B dans le contexte initial et delà le processus MES effectue une deuxième itération.

2^{ème} itération :

La nouvelle valeur de l'élément contextuel EBI permet à la règle R2 d'être vérifiée (cf. figure III.18). Ainsi, MES poursuit le processus de stabilisation et adapte la dimension contextuelle A à l'aide de la règle R2. Par conséquent, cette adaptation modifie l'élément contextuel EA3 dans la dimension A qui passe à une valeur « c' » dans la dimension adaptée A'.

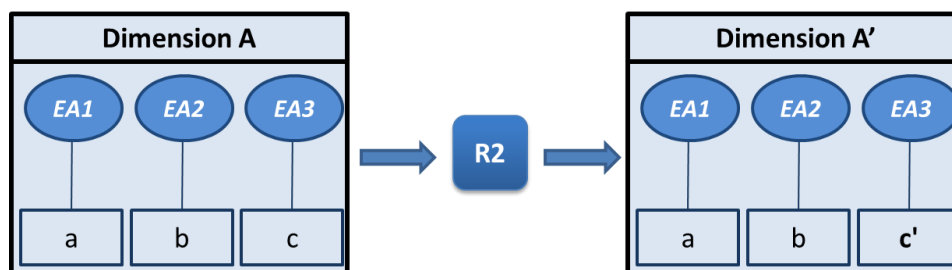


Figure III.20. Résultat de la deuxième itération

3^{ème} itération :

De la même manière, la nouvelle dimension contextuelle adaptée A' ou plus exactement la nouvelle valeur de l'élément EA3 va avoir un impact sur les autres dimensions. En effet, les conditions de la règle R3 sont alors vérifiées. R3 sera donc appliquée par le processus MES qui continue la stabilité de situation tant qu'il reste des règles à appliquer. Il en résulte une nouvelle dimension C' qui sera insérée à la fin de cette itération avec le reste des dimensions contextuelles (cf. figure III.21). C'est ainsi qu'à la fin de la troisième itération de l'algorithme 2 le contexte initial contient les dimensions adaptées A', B' et C'.

Il est important de noter que la règle R2 et R3 ne pouvaient pas être appliquées (vérifiées) si les dimensions contextuelles de départ (cf. figure III.17) n'avait pas été remplacées au fur et à mesure par celles adaptées à la fin de chaque itération du cycle de stabilisation du contexte. C'est-à-dire, la dimension B' a remplacé B à la fin de la première itération et la dimension A' remplacé A à la fin de la deuxième itération.

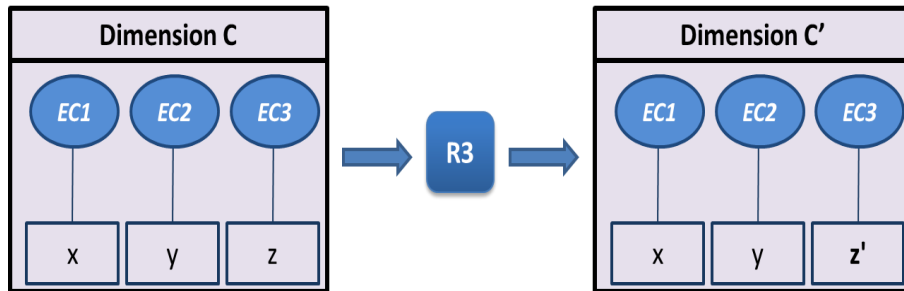


Figure III.21. Résultat de la troisième itération

4^{ème} itération :

Comme nous l'avons présenté précédemment, les trois dimensions contextuelles sont adaptées à la fin de la troisième itération du cycle de stabilisation du contexte. À ce stade, le processus MES n'arrête pas la mise en situation et une quatrième itération est effectuée. Par conséquent, la règle R1 peut être appliquée et la dimension contextuelle B (i.e. B') est adaptée une deuxième fois. L'élément contextuel $EB2$ est donc mis à jour et prend la valeur « e' » (cf. figure III.22).

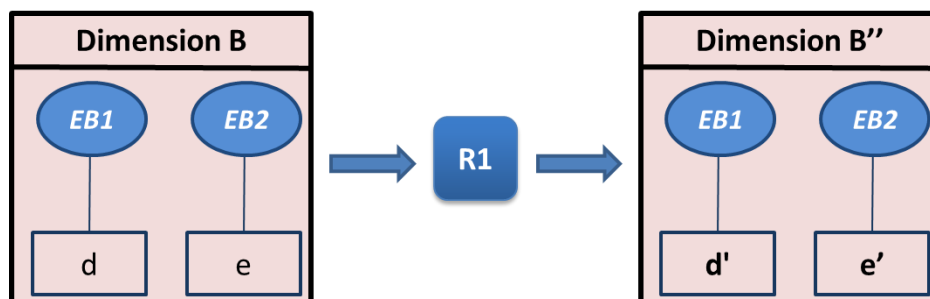


Figure III.22. Résultat de la quatrième itération

Enfin, le processus MES peut arrêter la stabilisation du contexte après la quatrième itération puisque le point de stabilité est atteint. *Le processus MES n'a en effet plus de règles pouvant être appliquées, la situation est donc générée.*

La figure III.23 illustre la situation résultante de la stabilisation du contexte composée des trois dimensions initiales. Elle reflète donc l'impact qu'ont les éléments contextuels de départ les uns sur les autres et la dynamique des relations réciproques qui en résulte. Le GSC fournira alors cette situation à l'application l'ayant sollicité pour cette contextualisation en plus des actions recommandées pour cette configuration contextuelle.

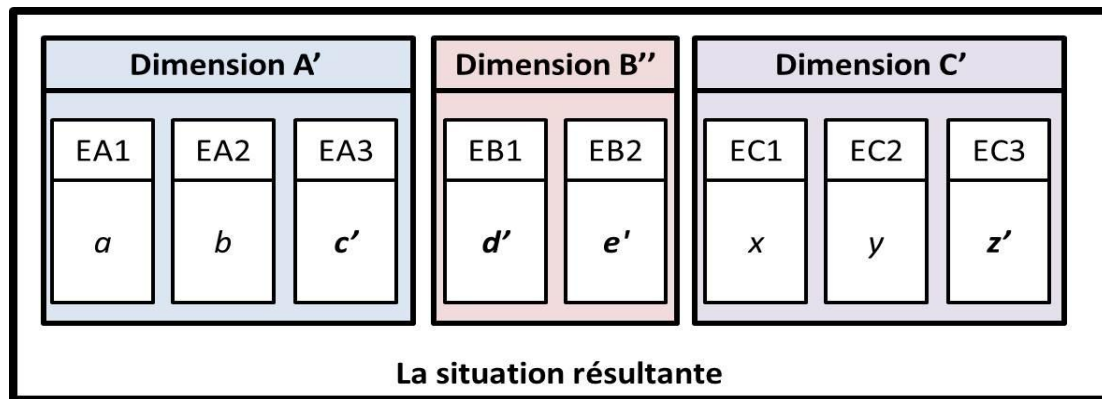


Figure III.23. la situation finale générée par le processus MES

Notons que pour les éléments contextuels adaptés, la fiabilité de leur valeur est obtenue à partir des propriétés des règles qui les ont modifiées. C'est-à-dire que pour l'élément EA3, par exemple, la fiabilité de sa valeur « c' » communiquée par le GSC à l'application est 0,9. Elle correspond à la priorité de la règle R2. De telle sorte, l'application connaît la fiabilité de tous les éléments contextuels afin de les utiliser par la suite dans son processus d'adaptation.

VIII. La démarche d'extraction de règles

Nous nous sommes tournés vers le domaine de la fouille de données et plus précisément vers les approches de fouille de règles d'association pour extraire de nouvelles règles à partir des situations enregistrées. En effet, les travaux existants dans ce domaine vont nous permettre de faire évoluer notre gestionnaire de données en lui ajoutant la composante d'apprentissage du processus d'extraction de règles (cf. section VI.4).

La présente section introduit la fouille de règles d'association ainsi que les approches qui permettent d'extraire des règles d'une base de données. Par la suite, nous décrivons notre démarche d'extraction de règle. Pour cela, nous détaillons l'algorithme du fonctionnement du processus d'extraction de règles tout en l'illustrant par un exemple

VIII.1. La fouille de règles d'association

La fouille de règles d'association est une tâche fondamentale dans la fouille de données. Son objectif est de trouver toutes les relations de cooccurrence, appelé les associations, parmi les toutes les données enregistrées. Elle a été proposée par Agrawal et ses collaborateurs dans (Agrawal et al., 1993) et appliquée dans l'analyse du panier de la ménagère. Celle-ci vise à découvrir comment les articles achetés par les clients dans un supermarché sont associés. En effet, pour Agrawal et ses collaborateurs, la base de données est constituée de l'ensemble des transactions des clients. Chaque transaction est composée d'un ensemble d'items qui représente les produits achetés par le client. La fouille de règles d'association consiste donc à récupérer des ensembles d'items fréquemment liés dans une même transaction, ainsi que des règles les combinant. Un exemple très utilisé de règle est : « 80% des gens qui achètent du fromage achètent également de la bière ».

Nous notons que les items associés dans les règles d'association dans le cas du panier la ménagère (par exemple : fromage et bière), communiquent des connaissances inconnues jusqu'à là par les vendeurs (exemple : les enseignes de grandes surfaces). Cette règle leur montre que ces deux produits doivent être associés dans la vente pour inciter le client à consommer, i.e. présentés aux clients pas loin l'un de l'autre.

VIII.1.1. Les concepts de la fouille de règles d'association

Nous avons défini dans la section VI.2.2 les règles d'association :

$$X \rightarrow Y, \text{ où } X \in T, Y \in T \text{ et } X \cap Y = \emptyset$$

Nous présentons dans ce qui suit les concepts utilisés pour l'extraction des règles d'association définies précédemment. En effet, trois outils statistiques permettent d'évaluer la qualité des règles d'association : **le support, la confiance et le lift.**

Le support d'une règle d'association, $X \rightarrow Y$, est le pourcentage de transactions de T qui contient $X \cup Y$, et qui peut être considéré comme une estimation de la probabilité, $P(X \cup Y)$. Le support de la règle d'association détermine ainsi quelle est la fréquence de l'application de la règle dans l'ensemble des transactions T . Soit n le nombre de transactions dans T . Le support de la règle $X \rightarrow Y$ est calculé de la manière suivante :

$$Support = \frac{\text{nombre d'apparitions } (X \cup Y)}{n}$$

La confiance d'une règle d'association, $X \rightarrow Y$, est le pourcentage des transactions appartenant à T qui contiennent X et qui contiennent également Y . La confiance peut être considérée comme une estimation de la probabilité conditionnelle $P(Y / X)$. La confiance de la règle $X \rightarrow Y$ est calculée de la manière suivante :

$$Confiance = \frac{Support(X \cup Y)}{Support(X)}$$

La confiance détermine donc la prévisibilité de la règle. Si la confiance d'une règle est trop élevée (tend vers 1), alors nous sommes presque sûrs de prédire Y si nous avons X . Par contre, une règle avec une faible confiance est d'une utilité limitée.

Le Lift d'une règle d'association, $X \rightarrow Y$, représente le rapport à l'indépendance de celle-ci. Le lift de la règle $X \rightarrow Y$ est calculé de la manière suivante :

$$Lift = \frac{Support(X \cup Y)}{Support(X) * Support(Y)}$$

Autrement dit, en divisant l'indice brut d'association entre X et Y ($Support(X \cup Y)$) sur l'indépendance entre ces deux éléments ($Support(X) * Support(Y)$), le lift donne la « distance » à l'indépendance de la règle. Par exemple, pour cette règle, un lift égal à 3 montre qu'un client qui achète l'item X a trois fois plus de chances qu'une personne normale d'acheter l'item Y .

A partir d'une base de transaction T , la question est d'extraire des règles d'association dont le support est supérieur à un seuil minimum défini par l'expert ($minSupp$) et dont la confiance est supérieure à une confiance spécifiée par l'expert ($minConf$). Plusieurs algorithmes basés sur ce principe ont été proposés dans (Agrawal et al., 1996, 1993; Agrawal, Srikant, & others, 1994). Cependant, d'autres travaux adaptent ces deux premières mesures en utilisant le lift afin d'obtenir des règles plus pertinentes (Brin, Motwani, Ullman, & Tsur, 1997). De cette manière, le lift des règles extraites doit être supérieur ou égal à 1 pour qu'elles soient fiables.

Afin de mieux appréhender les notions et concepts présentés dans cette section, nous allons donner un exemple.

VIII.1.2. Exemple

Nous proposons un exemple simplifié comportant un ensemble d'items et un jeu de transactions pouvant contenir ces items. La figure III.24 illustre ainsi une base d'enregistrements binaires comprenant cinq items (*A*, *B*, *C*, *D*, *E*) et cinq transactions numérotés de 1 à 5. Pour chaque transaction de l'ensemble des enregistrements de l'exemple, nous désignons si l'item lui appartient par le symbole « X ». Autrement dit, pour chaque transaction nous nous intéressons à l'ensemble d'items qui appartient à chaque transaction. Par exemple, la transaction numéro 1 est composée des items *A*, *C*, et *E*. Si nous voulons faire le lien entre cet ensemble et le panier de la ménagère introduit dans la section précédente, les items de la transaction numéro 1 sont les articles achetés par un client à une date donnée.

Par conséquent, pour extraire des règles d'association de ces enregistrements nous devons alors fixer un seuil minimum au support et à la confiance. Pour cet exemple nous supposons que le seuil minimum du support nommé minSup est égal à 0.7 et le seuil minimum de la confiance nommé minConf est égal à 0.6.

	A	B	C	D	E
1	X		X		X
2		X	X		
3	X	X	X		X
4		X	X	X	
5		X	X		

Figure III.25. Exemple de transactions

Les règles qui peuvent être extraites de cet ensemble de transactions et donc vérifiant les deux seuils sont la règle $B \rightarrow C$ [80%, 100%] et la règle $C \rightarrow B$ [80%, 80%]. En effet, la première règle $B \rightarrow C$ possède un support = $\frac{\text{nombre d'apparitions } (B \cup C)}{n} = \frac{4}{5} = 0.8$ et une confiance = $\frac{\text{Support } (B \cup C)}{\text{Support } (B)} = \frac{4}{4} = 1$.

Le même principe nous permet d'extraire la règle $C \rightarrow B$ avec un support et une confiance tous les deux égaux à 0.8.

Les deux règles extraites fournissent une connaissance générée à partir de cet ensemble de données enregistrées. Cette connaissance est un ensemble de corrélations entre les items composant ces transactions et peut être utilisée pour prédire des futures transactions. Ainsi, en utilisant par exemple la règle $B \rightarrow C$, nous sommes en mesure de prédire avec 100% de certitude que l'item C appartiendra à une future transaction si celle-ci contient l'item B.

VIII.2. Fonctionnement du processus d'extraction des règles

Dans cette partie, nous allons nous intéresser à la démarche complète de l'extraction de règles d'association à partir des situations enregistrées. Ainsi, après avoir présenté la notion de fouille de règles d'association en plus de ses concepts, nous allons montrer en premier lieu l'apport de cette technique de fouille de données pour notre GSC. Nous détaillerons en deuxième lieu la méthode utilisée du processus d'extraction de *règles inférées* (cf. section VI.4).

Pour cela nous décrivons le fonctionnement du processus d'extraction de règles que nous avons abordé dans (Chaker et al., 2010a). Ainsi, nous commencerons par exposer nos motivations pour la fouille de règles d'association à partir des situations passées. Par la suite, nous fournissons l'algorithme général du processus d'extraction de règles d'association en plus d'un exemple illustratif afin de mieux comprendre le déroulement de l'approche sur des situations enregistrées dans le temps.

VIII.2.1. Motivation

La première motivation pour l'extraction de connaissances sur le plan général et pour la fouille de règles d'association en particulier dans notre approche est la base disponible de situations générées par le GDC. Effectivement, les situations contextuelles stockées dans le temps représentent potentiellement une base très abondante de connaissances contextuelles. En effet, cette masse stockée d'informations mise en relation par le GDC doit être exploitée afin de permettre une analyse encore plus fine de son contenu. D'ailleurs, les situations représentent la traçabilité de toutes les mises en situation des différentes dimensions contextuelles, i.e. les situations constituent l'intégralité de l'historique du contexte.

En plus de cela, l'intérêt que nous portons pour les situations passées est justifié non seulement parce que celles-ci contiennent tous les éléments contextuels mis en situation, mais plus encore, car elles contiennent les différentes relations exploitables qui sont issues du cycle d'adaptation du processus MES. En d'autres termes, la composition des valeurs des éléments contextuelles dans les situations permet de faire ressortir des régularités provenant de la coexistence entre ces éléments dans une même situation. C'est la raison pour laquelle, nous nous sommes tournés vers les techniques de fouille de données pour extraire ces nouvelles connaissances.

Le dernier point concernant la coexistence des éléments contextuels et l'impact qu'elle peut avoir sur les situations enregistrées nous a poussé à nous focaliser précisément sur la fouille des règles d'association. Effectivement, nous avons constaté que la création des situations par le processus MES peut engendrer de nouvelles relations inconnues du GSC pouvant être exploitées. C'est-à-dire, les situations comportent des relations fonctionnelles qui peuvent exister entre les éléments contextuels qui la composent et donc la technique d'extraction de règles d'association nous offre la possibilité d'extraire ces corrélations entre les données contextuelles enregistrées.

En plus de tous les avantages mentionnés plus haut, l'extraction de nouvelles règles est pertinente puisque le MES se base lui aussi sur des règles pour la création des situations. Ainsi, les connaissances traduites par ces nouvelles corrélations entre les éléments contextuels des situations vont se traduire par de nouvelles règles qui alimentent notre base initiale de règles.

Nous allons prendre le même exemple pris dans la section VIII.1.2, pour montrer l'analogie entre la fouille de règles d'association dans le panier de la ménagère et celle abordée dans notre approche. Ainsi, la figure III.25 illustre un ensemble de cinq situations enregistrées dans le temps. Ces situations peuvent être considérées comme un jeu de transactions d'achat de clients et les items qui les composent correspondent alors aux différents éléments contextuels du contexte initial. Les items dans cet exemple sont les cinq éléments contextuels A, B, C, D et E. Une transaction (la situation) sur cet exemple est donc un sous ensemble composé des éléments contextuels de départ.

L'ensemble des éléments contextuels issus des différentes dimensions du contexte

	A	B	C	D	E
1	x		x		x
2		x	x		
3	x	x	x		x
4		x	x	x	
5		x	x		

Figure III.25. Exemple de situations composées d'éléments contextuels

Nous remarquons que la figure III.25 illustre un exemple simplifié pour montrer notre motivation d'extraction de nouvelles règles d'association à partir des situations passées. Néanmoins, la situation dans notre approche est plus qu'un simple sous ensemble binaire d'occurrence des éléments contextuels de départ, comme le montre l'exemple. Elle représente une collection spécifique d'éléments contextuels issus du contexte initial ainsi que leurs valeurs. Les nouvelles règles extraites vont donc faire ressortir des corrélations entre les différentes valeurs de ces éléments contextuels.

VIII.2.2. Les étapes de l'extraction des règles

Le processus d'extraction de règles se base sur les informations contextuelles fournies par le GDC pour inférer les nouvelles règles. En effet, le GDC met à la disposition du processus d'extraction de règles la base des situations passées. En plus des situations enregistrées, le processus d'extraction de règles exploitent les dimensions contextuelles stockées dans le GDC pour donner une couche contextuelle aux nouvelles règles séquentielles extraites (cf. figure III.26). C'est-à-dire que les techniques de fouilles de règles d'association appliquées par le processus sur la base des situations vont générer de nouvelles règles composées simplement d'items non liés avec le contexte. Ces derniers sont à la base les éléments contextuels issus des différentes dimensions contextuelles. Le rôle du processus d'extraction de règles est de relier chacun de ces items avec l'élément contextuel qui lui correspond et ainsi avec la structure de sa dimension contextuelle. Par conséquent, nous

aurons des règles d'associations exploitables par le GSC et qui seront ajoutées à la base de celles détenues par le GDC.

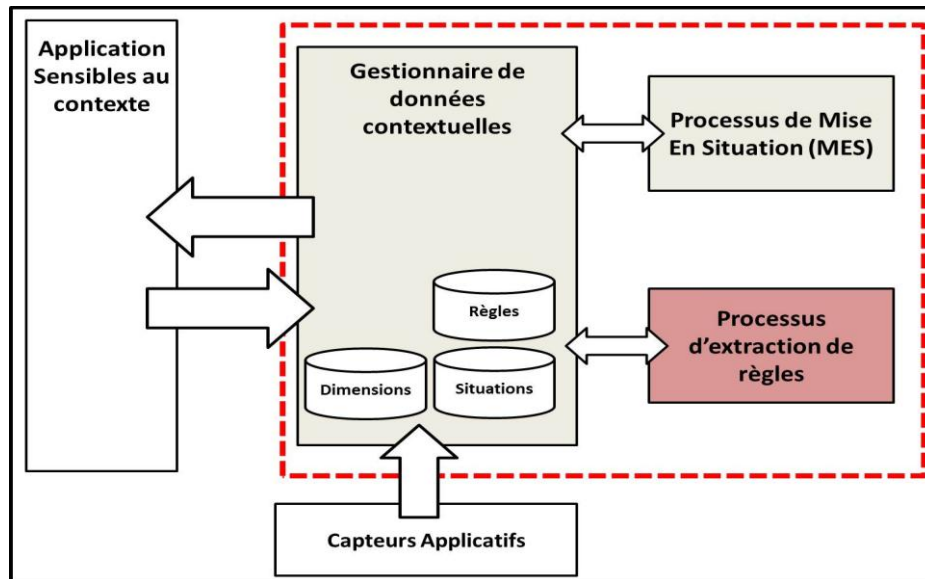


Figure III.26. Le processus d'extraction de règles

VIII.2.3. L'algorithme d'extraction de règles

Le processus d'extraction de règles se base sur les algorithmes existants d'extraction de règles d'associations pour inférer les nouvelles règles à partir des situations passées. Dans notre approche nous utilisons l'algorithme *apriori*, algorithme pionnier d'extraction de règles d'association (Agrawal et al., 1993). A ce niveau, le processus d'extraction de règles obtient de simples corrélations entre des items issues de différentes transactions (situations).

Par conséquent, le processus d'extraction de règles continue sa démarche en se basant sur les dimensions contextuelles qui constituent le contexte initial des situations enregistrées. Ces dimensions sont fournies par le GDC et vont servir à construire les règles de type *inférées* à partir des règles associations extraites dans l'étape précédente. Le rôle du processus dans cette étape est donc de donner un sens contextuel aux règles d'association leur permettant d'être exploitable par le GSC.

Enfin, le processus d'extraction de règles alimente la base de règles détenue par le GDC avec les nouvelles *règles inférées*. Chaque règle inférée est classée selon sa **priorité**. Dans notre approche, la **confiance** de la *règle inférée* est sa priorité.

Algorithme 3 : Extraction de nouvelles règles

Données : ContexteInitial : ensemble des valeurs des différentes dimensions contextuelles

Situations : l'ensemble des situations enregistrées

Règles : la base de toutes les règles;

minSup : le seuil minimum pour le support

minConf : le seuil minimum pour la confiance

minLift : le seuil minimum pour le lift

Résultat : RèglesInférées : l'ensemble des règles extraites ;

Début

SituationValides \leftarrow filtrerValidité (Situations) ;

règlesExtraites \leftarrow AlgorithmeFouilleDeRègles (SituationValides, minSup, minConf, minLift) ;

Pour chaque règle ϵ règlesExtraites **faire**

Tant que \exists élément ϵ règle **faire**

 dimension \leftarrow matcher (Contexte, élément) ;

 règleTampon \leftarrow ajouter (dimension, élément) ;

Fin tant que;

 règleInférée \leftarrow construire (règleTampon, priorité) ;

 RèglesInférées \leftarrow stocker (règleInférée) ;

Fin Pour;

Retourner ordonner (RèglesInférées, priorité) ;

Fin

Nous remarquons que l'intégration des nouvelles règles extraites dans la base initiale des règles n'est pas une procédure automatique. En effet, dans un contexte métier, les règles fouillées par le processus ne peuvent pas être intégrées automatiquement à la base de règle sans une validation de l'expert du domaine.

VIII.2.4. Exemple d'extraction de règles dans notre approche

Nous présentons un exemple pour mieux appréhender l'approche de fouille de règles sur les données contextuelles. Ainsi, nous supposons un contexte de départ composé de deux dimensions contextuelles nommées *DCA* et *DCB*. De plus, ces deux dimensions sont chacune composées de quatre éléments contextuels : *EA1*, *EA2*, *EB1* et *EB2*.

La figure III.27 illustre un exemple d'une base de situations composée de quatre situations enregistrées par le GSC. Nous remarquons que dans cet exemple et pour des raisons

de simplicité, nous nous intéressons uniquement à l'ensemble des éléments contextuels des situations. Par conséquent, nous n'aborderons pas l'ensemble des actions et la validité des situations qui font également partie des concepts de la situation (cf. section V.3).

	EA1	EA2	EB1	EB2
Situation 1	a	b	c	d
Situation 2	x	y	z	p
Situation 3	n	y	c	d
Situation 4	a	b	c	d

Figure III.27. Exemple de quatre situations enregistrées composées de quatre éléments contextuels

À partir de cet ensemble d'enregistrements, nous pouvons extraire de nouvelles règles et les ajouter à notre base, et plus précisément y ajouter les *règles inférées*. Nous fixons d'abord les deux seuils minimums du support et de confiance indispensables à la fouille de règles d'association. Ainsi, le seuil minimum du support est fixé à 60% de l'ensemble des situations enregistrées et le seuil minimum de confiance est fixé à 50%. Par conséquent, la règle extraite est : $EB1 = \langle c \rangle \rightarrow EB2 = \langle d \rangle$. En effet, cette règle a un support de 75%, i.e. sur 75% des situations enregistrées, les deux éléments contextuels *EB1* et *EB2* possèdent respectivement les valeurs « c » et « d ». Cette règle a également une confiance maximale égale à 100% car à chaque fois qu'une situation contient l'élément contextuel *EB1* avec la valeur « c », la même situation contient l'élément contextuel *EB2* avec toujours la valeur « d ».

La nouvelle règle extraite $DCB.EB1 = \langle c \rangle \rightarrow DCB.EB2 = \langle d \rangle$ est du type *règle inférée* qui va être ajoutée dans notre base de règles. Ainsi, cette règle pourra être utilisée dans la création des situations postérieures à la « situation 4 » (cf. figure III.27). Cette nouvelle règle offre à notre GSC une connaissance contextuelle supplémentaire, mais surtout pertinente. En effet, le processus de MES, lors de la création de nouvelles situations, saura qu'à chaque fois que les dimensions à contextualiser comportent l'élément contextuel *EB1* avec la valeur « c » ; alors l'élément contextuel *EB2*, si celui-ci fait partie de la même situation, aura la valeur « d ». Autrement dit, le processus MES sera capable en utilisant cette règle de prédire la valeur de *EB2* avec certitude.

Remarque : Ce que nous qualifions de **priorité** de la règle est sa **confiance** (c'est-à-dire 100% pour cet exemple). La priorité permet aussi de déterminer la **fiabilité** de la valeur « *d* » de l'élément contextuel *EB2* (égal à 1 pour cet exemple).

IX. Conclusion

Nous avons présenté au cours de ce chapitre le cœur de notre contribution : le GSC. Pour cela, nous avons défini la notion de contexte sur lequel repose notre travail. La définition donnée du contexte ainsi que les concepts qui le composent sont « génériques » et respectent un niveau d'abstraction assez élevé. Nous avons ensuite défini la situation qui découle directement du contexte et qui en est une interprétation à un instant *t*. Autrement dit, c'est une photographie d'une partie du contexte à un moment donné générée en utilisant une approche originale de création de situations.

Notre approche de création de situations est fondée sur le principe de l'interprétation du contexte que nous avons présenté au début de ce chapitre. L'originalité de cette approche à ce niveau réside dans :

- une approche de contexte générique qui n'exclue *a priori* aucune dimension contextuelle ;
- l'interprétation du contexte comme principe sur lequel se base la création de la situation. En effet, l'idée de l'interprétation d'un contexte motive le besoin de création des situations et l'avantage qu'elles offrent aux applications par rapport aux contextes initiaux ;
- la démarche de création de la situation basée sur la définition du contexte. Ainsi, chaque dimension est considérée séparément et adaptée selon son contexte, i.e. le reste des dimensions du contexte initial ;
- la stabilisation du contexte comme une étape importante dans cette démarche de création de la situation. Son originalité réside dans le fait qu'elle se concentre davantage sur la dynamique entre tous les éléments contextuels que sur la façon dont l'information est représentée.

Nous avons renforcé le GSC par un deuxième processus d'extraction de règles. En effet, ce processus permet à notre GSC d'évoluer au fil du temps. En d'autres termes, l'ensemble des règles utilisées par le processus MES devrait croître pour améliorer la précision des informations contextuelles. Plus précisément, le contexte stable traduit par la

situation est le fruit d'un cycle d'adaptation. Ce dernier n'est que la dynamique qui existe entre les éléments contextuels et qui impacte la constitution finale de la situation dont on ignore au préalable le point de stabilité et les valeurs des éléments qui la composent. Dans ces conditions le processus d'extraction a donc pour tâche principale d'identifier les nouvelles relations qui pourraient exister dans ces situations.

En résumé, notre GSC est un système « intelligent » qui utilise de façon optimale tous les éléments contextuels combinés avec les connaissances acquises à partir des situations passées. Les règles extraites des activités réelles permettent au GSC d'évoluer dans le temps, ce qui signifie que le processus MES, et plus précisément son cycle de stabilisation du contexte, devient de plus en plus efficace. En effet, l'adaptation des dimensions contextuelles (la contextualisation des dimensions) devient plus pertinente parce qu'elle prend en compte l'historique du contexte exprimé par les situations enregistrées.

Le GSC a été mis en œuvre à travers un prototype présenté dans la partie suivante. Le but de cette réalisation est de montrer la faisabilité de notre approche et ce qu'elle pourrait apporter aux applications sensibles au contexte, notamment celles utilisées dans des contextes métier considérés plus complexes (**maintenance aéronautique**). Cependant, le prototype déployé est destiné spécialement à améliorer le processus informationnel des applications métier à besoin informationnel. Pour cela, nous allons nous baser sur les travaux présentés dans le deuxième chapitre de l'état de l'art et plus précisément, sur les dimensions contextuelles les plus pertinentes dans la contextualisation de l'accès à l'information (cf. chapitre I.2 section V).

**Chapitre II.2 :
Implantation
et expérimentation -
application à l'accès
à l'information dans la
maintenance aéronautique**

I.	Introduction	171
II.	Spécificités du GSC pour l'accès à l'information.....	172
	II.1. Objectif.....	172
	II.2. Différentes composantes du GSC pour les SAI métier	173
	II.3. Exemple de mise en situation pour un SAI métier	174
	II.4. Exemple d'extraction de règles pertinentes pour un SAI métier	176
III.	Architecture globale du prototype implanté	177
	III.1. Serveur de situations	179
	III.2. Serveur de données.....	179
	III.3. Gestionnaire de données contextuelles	180
	III.4. Le processus MES	181
	III.5. Processus de fouille de règles.....	181
IV.	Expérimentation basée sur les données de la maintenance aéronautique.....	
	182
	IV.1. Etude du domaine de la maintenance	182
	IV.2. Modélisation des trois dimensions pour la maintenance.....	183
	IV.2.1. Dimension environnement.....	183
	IV.2.2. Dimensions usager.....	184
	IV.2.3. Dimension tâche	184
	IV.3. Simulation de mises en situations	185
	a) Transformation des tâches.....	185
	b) Modélisation des règles	187
	IV.4. Fouille de règles basée sur des situations réelles	188
	IV.4.1. Méthodologie.....	188
	IV.4.2. Résultat de la fouille de règles.....	189
V.	Conclusion.....	190

I. Introduction

Nous avons présenté dans le premier chapitre de la deuxième partie de notre travail une approche générique de gestion des situations contextuelles. En effet, le présenté ne se limite pas à un domaine métier en particulier. Il se fonde sur notre définition générique du contexte et n'exclut aucune dimension contextuelle qui pourrait être représentée dans notre gestionnaire.

Nous proposons de présenter le champ d'application dans lequel nous avons étudié nos problématiques : le domaine de la maintenance aéronautique. Ce champ d'application va nous servir de base pour l'évaluation de notre approche. La caractéristique principale de ce domaine métier réside dans le fait que l'enjeu de fiabilité est fort et que ses tâches sont précisément définies. La garantie de cette fiabilité est étroitement liée à l'accès à l'information des usagers et au contexte dans lequel se déroule la tâche de maintenance. En effet, la documentation est très importante dans le domaine de la maintenance aéronautique et son utilisation a également une fonction légale. Dans ce cadre, le GSC proposé prend tout son sens.

Les tâches métier qui nous intéressent ici ont un objectif précis, éventuellement critique, elles nécessitent de la part de l'utilisateur une connaissance clairement identifiée. Pour pallier le manque d'informations, le Système d'Accès à l'Information (SAI) va tenter d'assister l'utilisateur pour trouver l'information adéquate par le biais de tâches de recherche d'information. Un besoin en information non satisfait peut être lourd de conséquences pour la réalisation de la tâche et peut engendrer des erreurs très coûteuses.

Pour atteindre ce but, il est important que le SAI puisse répondre correctement aux besoins de l'utilisateur tout en respectant les contraintes liées à son contexte, dans lequel il effectue sa tâche. Le contexte métier sera donc exploité par le SAI afin d'adapter (1) le processus de recherche d'information lui-même ou (2) les informations retournées à l'utilisateur (en termes de contenu/visualisation/structure) voire (3) les tâches métier elles-mêmes.

Le SAI a donc besoin de tout ce qui peut être considéré comme informations contextuelles pertinentes pour modéliser le contexte. Pour cela, nous rappelons que les travaux dans le domaine de l'accès et la recherche d'information se sont mis d'accord autour

de trois dimensions principales de la contextualisation du processus d'accès à l'information (cf. chapitre I.2 section V) : les dimensions **usager**, **tâche** et **environnement**.

Notre GSC a pour rôle de gérer ces trois dimensions contextuelles et de fournir aux SAI les situations générées. Rappelons que nous ne nous intéressons pas à la manière dont les SAI adaptent leurs processus informationnels, mais plutôt au contrôle et à la gestion des informations contextuelles afin de leur garantir la meilleure photographie possible du contexte à un instant donné.

II. Spécificités du GSC pour l'accès à l'information

II.1. Objectif

Dans le cadre métier de la maintenance aéronautique, le SAI a pour objectif principal d'assister les usagers en leur permettant de trouver l'information nécessaire au bon déroulement de leurs tâches de maintenance. Le SAI sera jugé efficace si l'activité de l'utilisateur pour effectuer sa tâche tend vers un chemin optimum. Le SAI doit se baser sur les trois dimensions du contexte qualifiées par la littérature comme importantes afin de fournir l'information pertinente à un usager, avec toutes ses spécificités qui réalise une tâche métier précisément définie (avec tout ce qu'elle nécessite comme processus informationnel) dans un environnement particulier. Notons que le système doit également tenir compte des relations de cette tâche métier par rapport aux autres tâches qui lui sont liées.

Plus précisément, au-delà des dimensions du contexte, le GSC gère la conjonction de tous ces éléments contextuels issus des différentes dimensions du contexte métier (cf. chapitre II.1 section VII). Le GSC propose aussi au SAI des connaissances supplémentaires extraites de l'historique du contexte (par exemple, des actions) qui vont lui permettre, par exemple, d'adapter les processus informationnels.

Finalement, l'intérêt d'exploiter les situations par le SAI réside dans le fait que ce dernier aura un accès complet (tous les éléments contextuels) et facilité à toutes les caractéristiques de la situation sans pouvoir, préalablement, interférer avec les modèles initiaux des dimensions contenues dans le contexte. C'est-à-dire, le SAI n'interagit qu'avec les situations fournies par le GSC et ne se préoccupe pas d'aller chercher les valeurs des éléments contextuels directement dans les dimensions adéquates.

II.2. Différentes composantes du GSC pour les SAI métier

La figure IV.1 illustre le modèle général du GSC pour les SAI. Nous rappelons que celui-ci est composé du GDC, du processus MES et enfin du processus d'extraction de règles (cf. chapitre II.1 section VI).

Pour cela, le SAI métier définit le triptyque sur lequel il souhaiterait que la contextualisation soit effectuée à un instant donné. Autrement dit, le SAI demande au GSC de générer la situation comportant l'utilisateur qui effectue une tâche précise dans l'environnement donné.

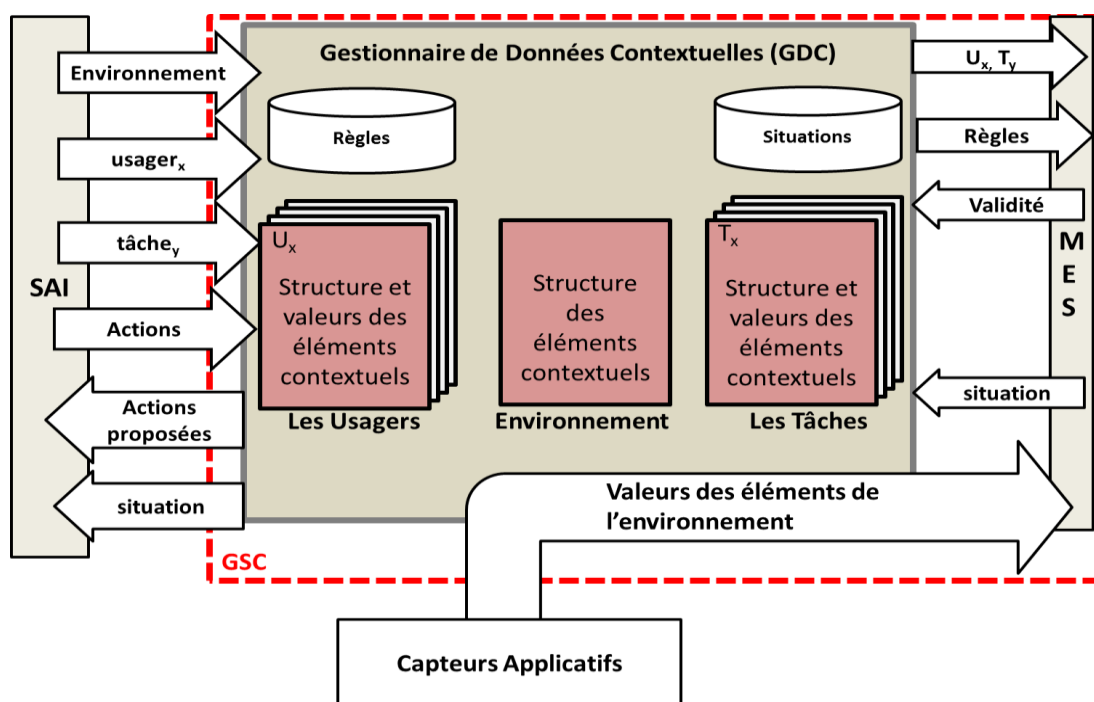


Figure IV.1. Les différentes composantes du GSC orienté SAI métier

Le GDC possède tous les modèles (organisation hiérarchique) des trois dimensions contextuelles (usager, tâche et environnement) et les types de chacun de leurs éléments contextuels. De plus, le GDC stocke les instances des deux premières dimensions car la dimension environnement ne peut pas avoir d'instances connues au préalable. Le GDC contient donc toutes les valeurs de ces deux dimensions. Autrement dit, il gère l'ensemble des usagers ainsi que les tâches métier pour un SAI particulier. Nous précisons que le GDC gère les données des dimensions contextuelles séparément pour chaque SAI possédant son propre domaine et caractéristiques. Chaque SAI envoie, au moment de l'initialisation du système, toutes ces informations contextuelles au GSC et plus précisément au GDC. De la même façon,

le SAI envoie aussi les deux types de règles fournies par les experts du domaine : les règles *métier légales* et *métier* (cf. chapitre II.1 section VII.1.1).

Le GDC conserve aussi les informations contextuelles produites par le GSC. En effet, les règles *inférées* par le processus d'extraction de règles ainsi que les situations finales créées par le MES sont conservées par GDC.

Dans notre exemple sur la figure IV.1, le SAI communique au GDC l'*usager_x*, et la *tâche_y* en plus des informations décrivant l'environnement. Par conséquent, le GDC retransmet au processus MES les valeurs dimensions contextuelles correspondantes pour une mise en situation. Plus précisément, dans notre exemple, le GDC fournit au MES les instances U_x et T_y provenant respectivement des dimensions contextuelles usagers et tâches. Le GDC apporte aussi au MES toutes les valeurs des éléments de la dimension environnement en plus de la base des règles spécifiques à ce SAI métier. Enfin, le GDC propose la situation générée par le MES au SAI qui l'avait sollicité (cf. chapitre II.1 section VII.1.1).

II.3. Exemple de mise en situation pour un SAI métier

Nous voulons reprendre la démarche de création de situation présentée dans la section VII du chapitre II.1 afin de monter le déroulement de la mise en situation. La figure IV.2 illustre un exemple simplifié de contexte initial communiqué au processus MES par le GDC. Ce contexte est composé des trois valeurs des dimensions contextuelles **usager** U , **tâche** T et **environnement** E . L'usager U possède trois éléments contextuels qui sont : son nom, son expérience du domaine et enfin son état émotionnel. De même, la tâche T possède deux caractéristiques qui sont : le temps moyen requis pour son exécution ainsi que son objectif. Enfin, l'environnement dans lequel U va accomplir T est représenté par les trois éléments contextuels : température, lieu et niveau sonore.

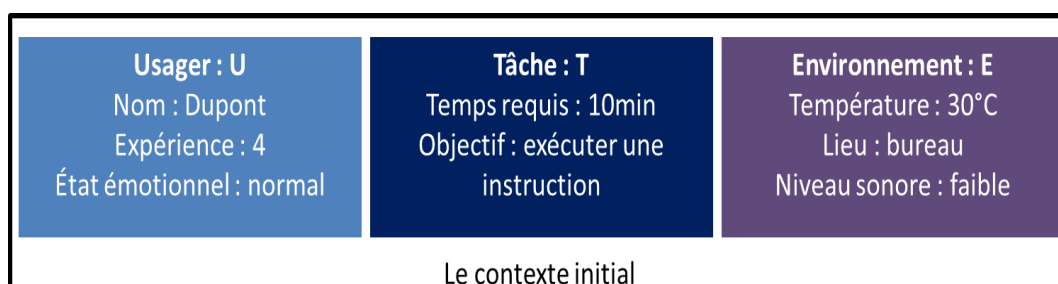


Figure IV.2. Exemple d'un contexte initial pour les SAI métier

Pour réaliser le cycle d'adaptation, le processus MES repose sur un ensemble de cinq règles permettant de décrire les différentes transformations à appliquer (cf. figure IV.3). Les règles R2, R3 et R4 sont des règles métier fournies par les experts du domaine et R1 et R5 sont des règles inférées créées par le processus d'extraction de règles. Nous nous focalisons dans cette section sur la fonction itérative de mise en situations. Ainsi, les *règles métier légales* ne sont pas abordées dans cet exemple parce qu'elles interviennent uniquement pour le contrôle de la validité et.

- R1:** Usager.experience ≤ 4 **ET** Environnement.lieu = bureau
 → Tâche.temps requis = 50min Priorité : 0,9
- R2:** Tâche.temps requis > 40min **ET** Environnement.température > 25°C
 → Usager.état émotionnel = stressé Priorité : 0,8
- R3:** Usager.état émotionnel = stressé **ET** Tâche.objectif = exécuter une instruction
 → Environnement.niveau sonore = fort Priorité : 0,7
- R4:** Usager.experience > 8 **ET** Environnement.lieu = maison
 → Tâche.temps requis = 5min Priorité : 0,6
- R5:** Usager.état émotionnel = stressé **ET** Environnement.niveau sonore = fort
 → Tâche.temps requis = 70min Priorité : 0,95

Figure IV.3. Un exemple de cinq règles

Dans notre cas, seule la règle R1 sera appliquée impliquant sur la tâche adaptée une durée requise de 50 min au lieu de 10 min avec une fiabilité de 0.9 (cf. figure ci-dessous). Grâce à cette information le SAI sait que le temps d'exécution de la tâche sera supérieur à son temps de référence (i.e. 10min). Dans ce cas, le SAI pourra par exemple recommander directement des informations pertinentes à l'utilisateur pendant le déroulement de la tâche en se basant sur les actions des situations passées afin de réduire le temps d'exécution.

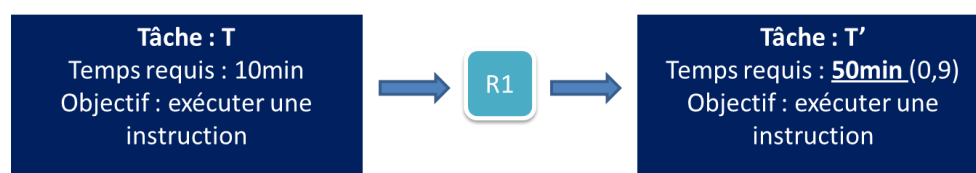


Figure IV.4. L'adaptation de la tâche T

En déroulant l'algorithme de création de situation comme nous l'avons fait dans la section VII.5.3 du chapitre II.1, le point de stabilité sera atteint après quatre itérations. Le processus MES engendre ainsi la situation de la figure IV.5.

La figure IV.5 illustre la situation résultante des trois dimensions simplifiées considérées dans cet exemple. Cette situation sera fournie par le GSC au SAI métier qui avait demandé sa création.

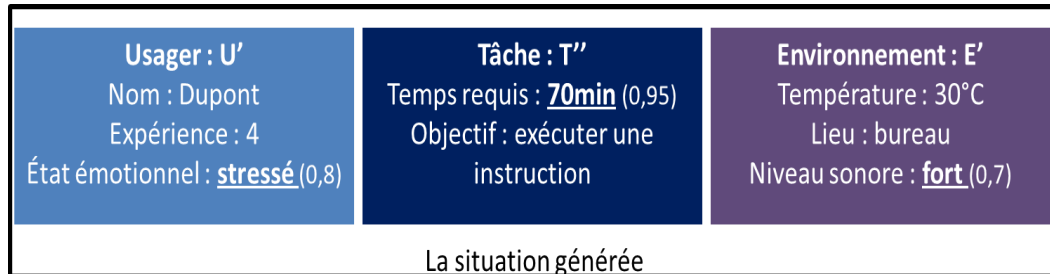


Figure IV.5. la situation finale générée par le processus MES

Nous pouvons voir plus clairement l'intérêt du processus de MES dans le cas des SAI dédiés dans des contextes métier : le GSC va leur fournir des situations qui reflètent la réalité contextuelle du domaine la plus fidèle possible (en utilisant les règles). Le SAI pourra ainsi adapter soit son processus d'accès à l'information, soit son traitement de l'information en se basant sur les valeurs des éléments de ces situations ainsi que sur leur fiabilité (poids).

II.4. Exemple d'extraction de règles pertinentes pour un SAI métier

Pour illustrer le rôle du processus d'extraction de règles nous allons prendre un exemple simplifié de situations valides enregistrées :

	Usager				Tâche			Environnement		
	Nom usager	Rôle	Expérience	Sexe	Temps requis	Nom de la tâche	Pré-condition	Lieu	Niveau sonore	Température
1	101	Tech	Faible	M	>70	T10	Rien	Out	Fort	Normale
2	202	Ing	Faible	F	>70	T5	AAA	Out	Faible	Normale
3	1005	Tech	Faible	M	>70	T12	Rien	Bureau	Fort	Élevée
4	75	Ing	Faible	F	[10, 50]	T25	ABC	Out	moyen	Normale
5	450	Tech	Faible	M	>70	T5	AAA	Bureau	Fort	Élevée
6	30	Sec	Faible	F	< 10	T111	Rien	Bureau	moyen	Élevée
7	48	Tech	Faible	M	>70	T14	rien	Out	Fort	Normale
...										

Figure IV.6. Exemple de situations valides enregistrées

En partant de cet exemple, le processus MES va extraire un certain nombre de règles dont celle qui suit (cf. Figure IV.7). Cette règle est très pertinente pour cet ensemble de situations parce qu'elle a une confiance maximale, égale à 1. Chaque fois qu'un technicien avec une faible connaissance effectue une tâche dans un environnement bruyant, le GSC est sûr à 100% d'après l'historique des situations enregistrées que le temps d'exécution de la tâche va augmenter et devient supérieur à 70 minutes.

Nouvelle règle : Usager.rôle = « Tech » ET Usager.expérience = « Faible »
 ET Environnement.bruit = « Fort » → **Tâche.temps requis = « >70 »** Priorité = 1

Figure IV.7. Un exemple de règle extraite

III. Architecture globale du prototype implanté

Afin de montrer la faisabilité du GSC et de valider les propositions présentées dans ce mémoire de thèse, nous avons implémenté un prototype. Ce dernier s'articule aux applications en quête de contextualisation (exemple : SAI dans la maintenance aéronautique). Le prototype permet de réaliser les engagements du GSC envers les applications (cf. chapitre II.1 section VI.5).

Plus précisément, le prototype du GSC assure les fonctionnalités suivantes :

- contrôler les différentes informations contextuelles en entrée pour vérifier la validité des situations ;
- générer la situation en utilisant la démarche de création de situations présentée dans le chapitre II.1 section VII ;
- permettre la fouille de nouvelles règles et l'intégration de celles-ci dans la base des *règles inférées* ;

Cependant, il reste encore à :

- permettre de la recommandation des actions en se basant sur la comparaison de similitude entre les situations. L'objectif de proactivité du GSC n'a pas encore été implémenté (le module grisé sur la figure IV.8) ;
- permettre de gérer simultanément plusieurs applications. Le GDC n'est pas encore paramétré pour prendre en compte plusieurs domaines métiers.

Le prototype implémenté repose sur une architecture client-serveur schématisée dans la Figure IV.8. Cette architecture est basée d'une part, sur le SGBD « Oracle 10g2 » pour le stockage des situations, des différentes actions et des éléments qui les composent et d'autre part, sur une interface client « Java 1.6 » afin de simuler des mises en situation et l'extraction de nouvelles règles.

Le prototype se base sur le modèle du GSC et respecte donc les trois composantes présentées dans le chapitre II.1 : le MES, le GDC et le processus d'extraction de règles. Celles-ci sont à leur tour décomposées en modules (cf. Figure IV.8) dont les rôles sont détaillés ci-dessous. Pour son initialisation, le prototype communique avec l'application qui lui fournit : les *règles métier légales* et les *règles métier* ainsi que leurs structures en format XML ; les dimensions contextuelles en format XML ; et un fichier contenant la liste des noms d'éléments contextuels prédéfinis par les experts du domaine comme étant les déclencheurs du changement de situations.

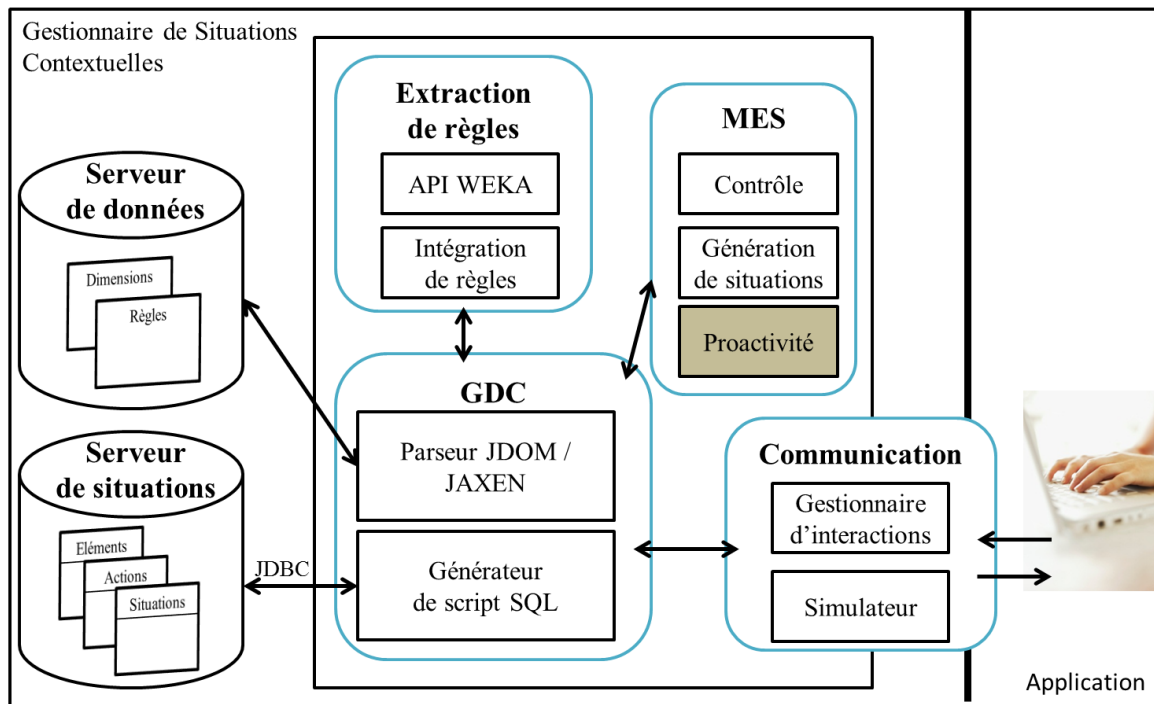


Figure IV.8. Architecture globale du prototype

III.1. Serveur de situations

Le module serveur de situations est une base de données relationnelle reposant sur le SGBD Oracle 10g2 ; il implante le modèle des situations sous forme de tables relationnelles.

Le serveur de situations communique avec l'application Java au travers de l'API JDBC qui fait partie de la composante GDC. Il permet d'exécuter des scripts SQL générés à partir d'un générateur de scripts SQL.

III.2. Serveur de données

Le module serveur de données est une base de données XML contenant l'ensemble des dimensions contextuelles en plus des trois types de règles. Un fichier XML est implanté pour chacune des dimensions contextuelles et pour chaque base de règles. Un exemple de ces fichiers est présenté dans la figure IV.9. Cette dernière montre un exemple de contenu de la base de *règles inférées* par notre prototype. La première règle illustrée sur la figure et dont la priorité est 0.8 admet que *si l'utilisateur est du domaine informatique et possède un niveau d'expérience élevé (expert) alors la tâche n'a pas de précondition.*

Le serveur de données communique exclusivement avec la composante GDC au travers de l'API JDOM / JAXEN.

```

<?xml version="1.0" ?>
<rules>
  <rule>
    <inputs>
      <input>
        <type>user</type>
        <attribute>/profils/personne/domaine</attribute>
        <value>informatique</value>
      </input>
      <input>
        <type>user</type>
        <attribute>/profils/personne/experience</attribute>
        <value>expert</value>
      </input>
    </inputs>
    <output>
      <type>task</type>
      <attribute>/tasks/task/precondition</attribute>
      <value>null</value>
    </output>
    <priority>0.8</priority>
  </rule>
  <rule>
    <inputs>
      <input>
        <type>environnement</type>
        <attribute>/environnement/cas/bruit</attribute>
        <value>0</value>
      </input>
    </inputs>
    <output>
      <type>user</type>
      <attribute>/pr ofils/personne/nom</attribute>
    </output>
  </rule>
</rules>

```

Figure IV.9. Exemple du contenu de la base des *règles inférées*

III.3. Gestionnaire de données contextuelles

Comme nous l'avons présenté dans la partie contribution de notre travail, le GDC est la composante qui gère la communication entre les modules du GSC, les bases de données et l'application. Le GDC se décompose en deux modules : le parseur JDOM / JAXEN et le générateur de script SQL.

Nous avons utilisé une API JDOM afin de manipuler soit les fichiers XML du serveur de données soit les fragments XML communiqués par l'application représentant le changement dans les informations contextuelles (i.e. changement de situation ou simplement l'état de situation). L'API JDOM offre un moyen de manipulation (lecture et d'écriture) de données XML en utilisant des notations simples. Contrairement aux autres API, l'API JDOM ne surcharge pas la mémoire.

Au travers de l'API JDOM, nous pouvons utiliser les éléments contextuels issus des différentes dimensions et les différents types de règles pour la mise en situation. Nous pouvons aussi écrire de nouvelles règles dans la base XML des *règles inférées*.

Nous avons aussi utilisé API JAXEN pour la manipulation des chemins XPATH. Nous nous sommes basés sur ces chemins pour gérer tous les noms des éléments contextuels afin de garantir leur unicité parmi toutes les dimensions contextuelles. Jaxen est une bibliothèque XPath open source écrite en Java. Elle est adaptable à de nombreux modèles d'objets différents, y compris les DOM, XOM, dom4j et JDOM.

III.4. Le processus MES

La composante MES de notre prototype se compose actuellement de deux modules parmi les trois de la figure IV.8 :

- le module de contrôle permet de vérifier avant chaque mise en situation si les valeurs des éléments contextuels ne contredisent pas les *règles métier légales* présentes dans la base des données. Dans ce cas, ce module déclenche la fonction d'invalidité de la situation : communique l'invalidité à l'application (usager) et au GDC pour qu'il enregistre la situation active (ou l'état de la situation) dans la base des situations. Les situations invalides sont enregistrées dans une table spécifique séparément des situations valides ;
- le module de génération des situations se base sur les règles métier et les règles légales pour créer les situations valides. Ce module est une implémentation de l'algorithme 2 présenté dans le chapitre II.1 section VII.5.2.

III.5. Processus de fouille de règles

Le module d'extraction de règles assure son objectif de fouille en se basant sur l'API Weka². Cette dernière est une collection d'algorithmes d'apprentissage automatique pour les tâches de fouille de données. L'API Weka est appelée à partir du code Java et offre ainsi des outils pour les données : pré-traitement, classification, *clustering*, règles d'association, et visualisation.

Le deuxième module d'intégration de *règles inférées* a pour tâche principale d'exploiter les règles extraites avec l'API Weka. Ainsi, il prépare le résultat des règles envoyées par Weka afin qu'elles soient exploitables par le parseur JDOM pour alimenter la base des règles inférées.

² <http://www.cs.waikato.ac.nz/ml/weka/>

IV. Expérimentation basée sur les données de la maintenance aéronautique

IV.1. Etude du domaine de la maintenance

Nous avons travaillé avec des experts du domaine de la maintenance aéronautique de l'entreprise AIRBUS pour mieux comprendre les besoins métiers de la maintenance et les spécificités des trois dimensions contextuelles.

Nous avons constaté, lors des réunions menées avec les experts, que malgré l'importance des facteurs contextuels sur le déroulement de procédures, la notion de contexte n'est pas encore utilisée dans les systèmes actuels.

Nous avons étudié avec eux quelques procédures / tâches (cf. annexe I) pour faire ressortir les trois dimensions contextuelles pertinentes dans l'accès à l'information. Les procédures actuelles regroupent ces trois dimensions : le profil de la personne qui réalise la tâche (dimension **Usager**), le déroulement de la tâche (dimension **Tâche**) et enfin dans quelles conditions il peut la réaliser (dimension **Environnement**).

Nous avons constaté également que les procédures / tâches ont chacune un objectif précis à atteindre. Elles ont une structure hiérarchique, i.e. elles sont décomposées en sous tâches (objectifs secondaires).

Enfin, nous avons constaté que les experts de la maintenance aéronautique classent les procédures en plusieurs groupes selon la nature de la tâche (cf. annexe II). Nous avons remarqué que ce classement des tâches (fait par les experts) est étroitement lié aux caractéristiques des autres dimensions (usager et environnement). Autrement dit, il existe un lien entre les différentes dimensions contextuelles, ce qui est la base de notre approche. Par exemple, le groupe de niveau trois (cf. annexe II) représente les tâches de maintenance qui doivent être faites dans des conditions environnementales bien particulières (hangar, ...). Un autre exemple montre que les tâches de nature « test et réglage » (*adjustement / test*) **exigent** de l'utilisateur une certaine concentration lors de l'exécution en plus d'un niveau minimum d'expérience. Un dernier exemple est que les tâches de nature « service » (*servicing*) sont **sensibles** à l'état des variables environnementales (tempête, grêle, foudre, etc.).

En conclusion, nous avons compris que les dimensions pertinentes de la maintenance aéronautique correspondent aux trois dimensions pertinentes présentées dans le deuxième chapitre de la partie I. Les caractéristiques de ces dimensions doivent être intrinsèquement

liées pour le bon déroulement de la procédure. Cependant, aucun contrôle ni feedback ne peut assurer qu'il y ait eu un respect des normes prescrites.

IV.2. Modélisation des trois dimensions pour la maintenance

Après la validation des trois dimensions avec les experts, la deuxième étape de notre expérimentation était de faire ressortir les variables pertinentes (éléments contextuels) pour chacune d'entre elles. En effet, ces variables sont sous-jacentes aux procédures de maintenance et ne sont donc pas modélisées dans les systèmes de maintenance actuels. Nous nous sommes alors fondés sur l'analyse des procédures et sur les discussions que nous avons eues avec les experts du domaine.

IV.2.1. Dimension environnement

Nous avons conclu avec l'aide des experts du domaine que les éléments contextuels de la dimension environnement peuvent avoir un impact sur le déroulement des procédures de maintenance aéronautique ce sont :

- les conditions météorologiques (**météo**) dans lesquelles se déroule la tâche de maintenance (beau, nuages, pluie, orageux, neige) ;
- la vitesse du **vent** en Km/h ;
- l'heure et le jour ;
- le **niveau sonore** qui est mesuré aux alentours de l'opérateur effectuant sa tâche. Le niveau sonore est calculé en décibel (dB), mais nous avons conclu que trois niveaux peuvent être utilisés selon la valeur : niveau fort, niveau moyen et niveau faible ;
- le **milieu** où se déroule la tâche de maintenance est très important et il est lié à la nature de la tâche et aux conditions météorologiques. Les milieux retenus sont : tarmac, atelier et hangars d'entretien ;
- la taille de l'**écran** aussi peut être un facteur environnemental très important dans la maintenance aéronautique. En effet, les opérateurs utilisent soit des ordinateurs de bureau (PC) ou des PDA ;
- le niveau de **luminosité** est un facteur important dans la maintenance. Il affecte beaucoup l'efficacité de la réalisation de la tâche.

IV.2.2. Dimensions usager

De même, les éléments contextuels de la dimension usager qui sont considérés les plus pertinents pour le domaine de maintenance aéronautique sont :

- l'**expertise** des opérateurs représente sans doute l'élément de l'utilisateur le plus important dans la maintenance aéronautique. Le niveau d'expérience de l'opérateur (débutants, intermédiaire, experts) détermine si ce dernier peut ou non effectuer une procédure de maintenance, i.e. certaines tâches de maintenance exigent des opérateurs un certain niveau d'expertise ;
- en plus de leur niveau d'expertise, dans la maintenance aéronautique, sont attribuées aux opérateurs des catégories bien définies de **qualification**. Les opérateurs ont une qualification parmi les quatre suivantes : catégorie A, catégorie B1, catégorie B2, catégorie C. Certaines tâches nécessitent une qualification spécifique ;
- enfin le **rôle** de l'opérateur est aussi une des variables significatives dans le domaine. Nous avons identifié les rôles suivants : mécaniciens avec spécialités (aéronautique, cellule, électronicien, monteur câbleur aéronautique), planificateur, réviseur, formateur.

IV.2.3. Dimension tâche

Les éléments contextuels de la dimension tâche que nous avons extraits des procédures avec l'aide des experts du domaine de maintenance aéronautique sont :

- les tâches de maintenance aéronautique, qui peuvent être réparties en deux **types** : les opérations de maintenance programmées ou maintenance en atelier et les opérations de maintenance non programmées ou maintenance d'escale ;
- comme nous l'avons signalé précédemment, les experts en maintenance aéronautique font une catégorisation des tâches selon leur **nature**. Nous pouvons citer : *maintenance practices, cleaning (c), Servicing, Inspection/Check, Deactivation/Reactivation, Removal/Installation, Test(T), Mmel* ;
- la **fréquence** de la tâche est un élément contextuel ayant une importance dans le bon déroulement des procédures. Nous avons remarqué que les tâches peuvent avoir trois fréquences possibles : routinière, perçue plus facile ; peu fréquente et enfin exceptionnelle ;

- dans notre domaine la tâche est caractérisée selon son **exigence**. Ainsi, selon l'objectif à atteindre par la procédure, les experts du domaine jugent si la tâche est exigeante ou peu exigeante pour l'opérateur qui la réalise ;
- enfin, le **temps d'exécution** de la tâche (en seconde) est une estimation faite par les experts pour déterminer la durée moyenne nécessaire à sa réalisation.

IV.3. Simulation de mises en situations

Nous avons utilisé des procédures de maintenance aéronautique fournies par les experts pour simuler la création de situations avec notre prototype. Pour cela nous avons modélisé les tâches pour les rendre exploitables. De plus, nous avons extrait des règles à partir des procédures fournies.

La simulation de création de situations nous permet d'expérimenter le contrôle et la mise en situation avec des éléments extraits du contexte de la maintenance aéronautique.

a) *Transformation des tâches*

Nous avons présenté dans le deuxième chapitre de la première partie les travaux qui se sont penchés sur les formalismes de modélisation des tâches mais peu d'entre eux ont abouti à des outils utilisables, étant donné la difficulté de construire des modèles de tâche.

Nous considérons que *ConcurrentTaskTree* de (Paterno, 2000) est la notation la plus appropriée grâce à son apparence graphique et à l'outil qui lui est associé (l'environnement CTTE). CTT est une notation graphique utilisée pour spécifier les modèles de tâche pour les applications coopératives par une structure hiérarchique en indiquant les relations temporelles par des opérateurs. Les modèles de tâche peuvent ensuite être simulés pour étudier les différents chemins d'interaction possibles. La notation est par ailleurs basée sur quatre types de tâches (les tâches abstraites, les tâches utilisateur, les tâches système et les tâches d'interaction), ainsi que sur plusieurs opérateurs temporels pour davantage de précisions. De plus, CTT a reçu beaucoup d'attention de la part de la communauté scientifique car sa partie théorique est validée et complète. Le succès de ce formalisme et la flexibilité et l'extensibilité qu'il offre via son environnement CTTE expliquent qu'il a fait l'objet de nombreuses expérimentations.

L'importance de l'environnement CTTE est de rendre les modèles de tâche utilisables par d'autres systèmes, comme le recours à la structuration XML. Celui-ci permet de rendre les

modèles exploitables dans plusieurs plateformes hétérogènes. Ces environnements rendent aussi les formalismes de modèle des tâches facilement extensibles, en ajoutant des opérateurs ou en modifiant les concepts et les opérateurs de base selon le besoin des utilisateurs. CTTE est donc le meilleur exemple à donner. Il regroupe plusieurs outils de modélisation et d'exploitation des tâches. Par exemple, (Leal, Bouchet, Langlois, & Jourde, 2009) proposent un formalisme novateur étendant les formalismes existants CTT. (Beltran & Ravat, 2009) ont étendu le formalisme CTT travaux pour répondre aux besoins de la modélisation des tâches pour les entrepôts de connaissances.

Nous avons modélisé les procédures de maintenance fournies par les experts avec l'environnement CTTE en utilisant le formalisme CTT. La figure IV.10 montre une partie d'une des procédures (*opening_FWD_AFT*) modélisée avec CTT.

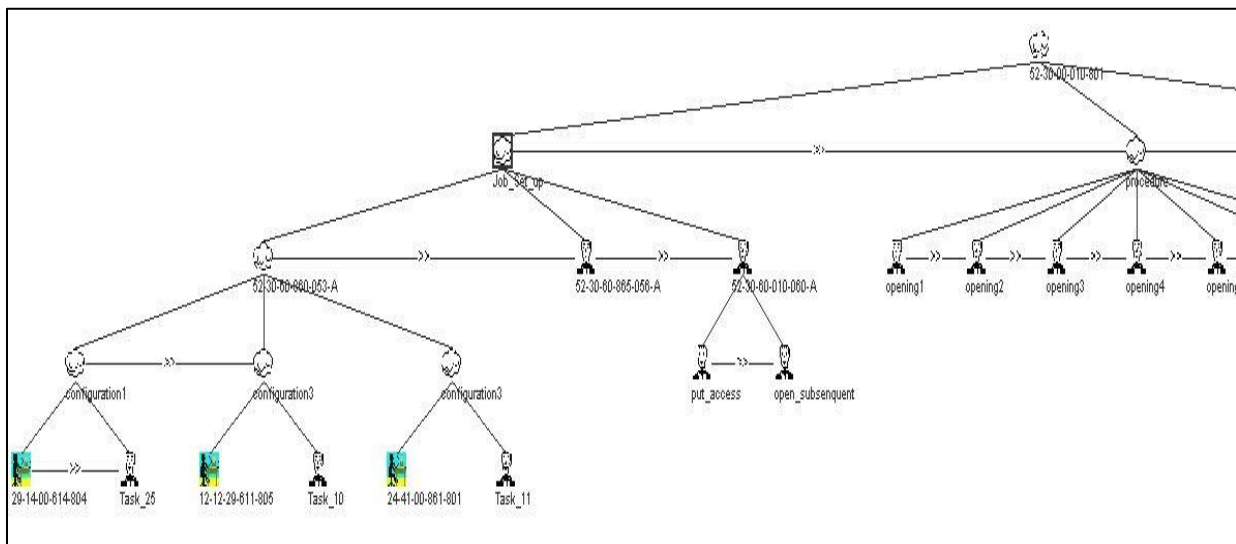


Figure IV.10. Exemple du contenu d'une tâche modélisée avec CTT

Par la suite et en utilisant l'environnement CTTE cette tâche sera convertie en XML, afin d'être utilisée par notre prototype (cf. figure IV.11).

```

<?xml version="1.0" encoding="UTF-8"?>
<TaskModel NameTaskModelID="implementer java.xml">
<Task Identifier="52-30-00-010-801" Category="Abstraction Task" Itera
<Name>opening_Fwd_Aff</Name>
<Type>MP</Type>
<Description>>null</Description>
<Precondition>>null</Precondition>
<TimePerformance>
  <Max>120</Max>
  <Min>50</Min>
  <Average>60</Average>
</TimePerformance>
<Object name="" class="" type="" access_mode="" cardinality="">
  <InputAction Description="null" From="null"/>
  <OutputAction Description="null" To="null"/>
</Object>
<Object name="" class="" type="" access_mode="" cardinality="">
  <InputAction Description="null" From="null"/>
  <OutputAction Description="null" To="null"/>
</Object>
<Object name="" class="" type="" access_mode="" cardinality="">
  <InputAction Description="null" From="null"/>
  <OutputAction Description="null" To="null"/>
</Object>
<Object name="" class="" type="" access_mode="" cardinality="">
  <InputAction Description="null" From="null"/>
  <OutputAction Description="null" To="null"/>
</Object>
<Object name="" class="" type="" access_mode="" cardinality="">
  <InputAction Description="null" From="null"/>
  <OutputAction Description="null" To="null"/>
</Object>
<Object name="" class="" type="" access_mode="" cardinality="">
  <InputAction Description="null" From="null"/>
  <OutputAction Description="null" To="null"/>
</Object>
<Object name="" class="" type="" access_mode="" cardinality="">
  <InputAction Description="null" From="null"/>
  <OutputAction Description="null" To="null"/>
</Object>
<Object name="" class="" type="" access_mode="" cardinality="">
  <InputAction Description="null" From="null"/>
  <OutputAction Description="null" To="null"/>
</Object>
<Object name="" class="" type="" access_mode="" cardinality="">
  <InputAction Description="null" From="null"/>
  <OutputAction Description="null" To="null"/>
</Object>
<Object name="" class="" type="" access_mode="" cardinality="">
  <InputAction Description="null" From="null"/>
  <OutputAction Description="null" To="null"/>
</Object>
<Object name="" class="" type="" access_mode="" cardinality="">
  <InputAction Description="null" From="null"/>
  <OutputAction Description="null" To="null"/>
</Object>

```

Figure IV.11. Exemple du contenu d'une tâche en XML

b) Modélisation des règles

Nous avons également extrait des règles (métier légal ou métier) à partir des procédures. Pour cela nous nous sommes basés sur les *Cautions* et les *Warnings* dans les procédures qui nous ont été fournies (cf. figure IV.12).

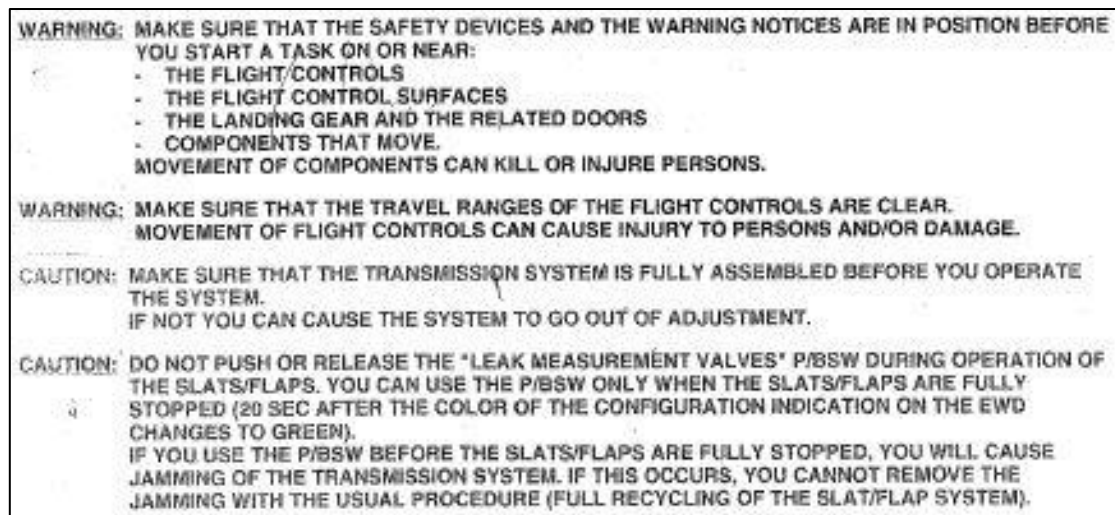


Figure IV.12. Exemple de *caution* et *warning* dans les procédures

Ainsi, dans les procédures de la maintenance un *warning* informe que des dégâts humains peuvent être causés s'il n'est pas respecté tandis qu'une *caution* informe du risque des dégâts matériels. Nous avons considéré les *warnings* et les *cautions* comme *règles métier légales* pour contrôler la validité de la situation.

IV.4. Fouille de règles basée sur des situations réelles

IV.4.1. Méthodologie

Pour des raisons de confidentialité, nous n'avons pas pu utiliser notre GSC dans de vraies conditions d'utilisation métier. Afin d'atténuer ce problème, nous nous sommes tournés vers la thèse en psychologie de Zafiharimalala (Zafiharimalala, 2011). L'auteur a travaillé dans le domaine de la maintenance aéronautique et plus exactement sur la façon dont les opérateurs accèdent à l'information. Zafiharimalala a réalisé des études de terrain en observant en détail les techniciens effectuant leurs tâches. La thèse présente aussi des expérimentations sur l'utilisation des petits et grands écrans pour la recherche et la consultation de l'information, toujours en maintenance aéronautique.

Nous avons choisi une des tâches de maintenance fortement liée à l'accès à l'information : la tâche T1 Zafiharimalala et la sous-procédure AMM 32-41-11-000-006-B. La tâche T1 a pour objectif de trouver une procédure dans la documentation **AMM 32-41-11-000-006-B** *Removal of the MLG Wheel* et consultation de la sous-procédure AMM 32-41-11-020-075-A *Preparation of the Wheel before Removal*. La tâche T1 est une tâche complexe et son temps de référence est estimé à 291 secondes.

Grace à notre collaboration avec Zafiharimalala nous avons pu récolter **23 situations réelles** (cf. annexe III). Les éléments contextuels composant ces situations sont fondés sur les variables pertinentes des trois dimensions de la maintenance présentées dans la section IV.2.

IV.4.2. Résultat de la fouille de règles

Nous avons implémenté ces situations avec notre prototype du GSC afin d'expérimenter notre processus d'extraction des règles. Nous avons par la suite comparé le résultat trouvé avec les observations et les conclusions de (Zafiharimalala, 2011).

Malgré le nombre limité de situations, nous avons pu obtenir des règles qui convergent vers le travail de Zafiharimalala. En effet, en nous fondant sur ces 23 situations récoltées, nous obtenons une règle de priorité maximale (=1) admettant que tous les opérateurs considérés **experts** ne dépassent pas le **temps de référence** de la tâche quand elle est exécutée sur un **PC**. Par contre, en utilisant un PDA, les temps d'exécution des experts varient et dépassent dans quelques cas le temps de référence. Pour les autres opérateurs novices dans le domaine (élève BTS maintenance aéronautique), ils dépassent le temps de référence dans 66 % des cas. En outre, contrairement aux experts, les opérateurs novices ne sont pas impactés par les dimensions des écrans utilisés.

L'explication de ces règles dans (Zafiharimalala, 2011) réside dans le fait que les experts ont l'habitude de travailler avec les PC pour réaliser cette tâche ce qui justifie les temps très courts de réalisation. Concernant l'utilisation des PDA, qui sont encore non employés dans le domaine de la maintenance (en phase d'expérimentation), les experts rencontrent beaucoup de difficultés à exécuter cette même tâche routinière sur un autre support (PDA). Enfin pour les novices, vu leur manque d'expérience du domaine, le temps d'exécution d'une tâche est logiquement supérieur à la normale.

Nous avons pu extraire d'autres règles qui ne peuvent pas être considérées pertinentes vu le nombre limité de situations utilisées. Parmi les règles extraites ayant une confiance égale à 1, nous pouvons citer, à titre d'exemple, celle qui admet que *les « experts » ayant une qualification « B2 » et effectuant une tâche « exigeante » dans un niveau de luminosité « moyen » ne sautent pas des étapes*. Cependant, le support de cette règle est très faible (= 6/23). Ce qui veut dire que malgré sa confiance maximale, nous ne pouvons pas juger si cette règle est pertinente ou pas en nous appuyant uniquement sur notre ensemble de situations et

plus précisément sur les 6 situations des experts. Nous aurons besoin de plus de situations dans lesquelles les usagers sont des experts pour déduire de la pertinence de la règle.

V. Conclusion

Dans ce chapitre, nous avons présenté notre GSC orienté vers SAI métier et plus spécifiquement vers le domaine de la maintenance aéronautique. Le GSC se base sur les trois dimensions considérées par la littérature comme les plus pertinentes, dans le contexte métier, de l'accès et de la recherche d'information. Notre gestionnaire de contexte est un système « intelligent » qui utilise d'une manière optimale tous les facteurs contextuels combinés à la connaissance issue de l'historique des situations. Les règles, extraites des activités réelles des usagers, permettent à notre gestionnaire de contexte d'évoluer dans le temps, de rendre le processus d'adaptation du contexte de plus en plus efficace et pertinent, et ce, dans le but de refléter encore davantage le contexte métier réel.

Nous avons également présenté dans ce chapitre l'architecture du prototype du GSC que nous avons implémenté. Ce prototype montre dans un premier temps la faisabilité technique de notre approche générique, et dans un deuxième temps la validité de nos propositions de mise en situation et de fouille de règles.

Pour des raisons de confidentialité du constructeur aéronautique, nous n'avons pas pu utiliser notre prototype sur un système d'information réel du domaine de la maintenance aéronautique dans lequel des usagers du domaine pouvaient interagir avec notre système. Cependant, pour combler cette limite nous avons collaboré avec les experts du domaine de la maintenance aéronautique pour étudier avec eux les éléments contextuels dans le domaine et pour ainsi avoir des données utiles pour nos expérimentations.

Nous avons validé avec les experts les variables à modéliser dans notre prototype. Dans notre approche, ces variables représentent les éléments contextuels des différentes dimensions (usager, tâche et environnement). Ensuite, nous nous sommes basés sur des procédures de la maintenance pour :

- les étudier avec l'aide des experts du domaine ;
- extraire d'éventuelles *règles métier légales* ;
- les modéliser en utilisant le formalisme CTT ;
- les transformer en XML exploitable par notre prototype.

En conséquence, nous avons pu simuler la création des situations avec des données issues de la maintenance aéronautique. Cependant, à part le contrôle de la validité des situations par rapport aux quelques *règles métier légales* que nous avons extraites des procédures, les situations générées ne reflètent pas l'originalité de notre démarche de création de situations. Malheureusement, pour ce faire, nous avons besoin de plus de règles (spécialement, les *règles métier* et *règles inférées*) et de l'utilisation de vrais opérateurs du domaine.

Par ailleurs, pour expérimenter le processus de fouille de règles, nous avons exploité le travail de Zafiharimalala pour extraire quelques situations réelles. Malgré le petit nombre de situations que nous avons utilisé, les résultats convergent vers les conclusions faites de Zafiharimalala. Ce dernier point est très motivant pour la réalisation d'expérimentations sur une multitude de situations réelles de maintenance pour étudier plus amplement les résultats fournis par notre prototype.

Conclusion générale

Synthèse de nos propositions

Les travaux présentés dans ce mémoire se situent dans le cadre général de la contextualisation des systèmes d'informations. Plus particulièrement, notre approche se concentre sur la fiabilité des informations contextuelles utilisées par les systèmes d'accès à l'information dans des contextes métiers. Contrairement à la plupart des travaux dans ce même cadre, nous ne nous intéressons pas à la manière avec laquelle ces systèmes adaptent leurs processus informationnels, mais plutôt au contrôle et à la gestion des informations contextuelles afin de leur garantir la photographie la plus fidèle possible du contexte à un instant donné.

Pour cela, nous avons proposé un GSC générique basé sur un ensemble de contributions qui portent sur quatre volets :

- la définition de la notion de contexte et ses composantes ;
- la définition de la notion de situation ;
- l'approche de génération des situations ;
- l'approche d'amélioration de la fiabilité du gestionnaire.

Dans un premier temps, nous avons étudié la notion de contexte parce qu'elle est la pierre angulaire de nos contributions. Pour cela, nous avons défini le contexte d'un objet comme étant toutes les dimensions contextuelles ayant un impact sur cet objet. Cette définition ne dépend pas d'un domaine en particulier, contrairement aux multitudes de définitions de la littérature. La définition que nous avons fournie est en phase avec les deux principales visions du contexte proposées par Dourish (Dourish, 2004) : la vue interactionnelle et la vue représentationnelle. Nous avons jugé que ces deux vues ne doivent pas se confronter, mais au contraire se compléter pour éloigner la définition du contexte de son aspect applicatif et ainsi obtenir sa généralité. En plus de la définition de la notion de contexte, nous avons décrit ses concepts. En effet, le contexte dans notre approche est composé d'un ensemble de dimensions contextuelles à structures hiérarchiques composées d'éléments contextuels. Ces derniers représentent les informations contextuelles élémentaires et possèdent des caractéristiques spécifiques afin de garantir leurs gestions et la fiabilité de leurs valeurs.

Nous avons aussi défini la situation comme étant une interprétation stable du contexte à un instant donné. Pour cela, nous avons montré les différences qui peuvent exister entre le contexte initial et son interprétation. En plus d'une simple interprétation, la situation que nous proposons est une conjonction originale des éléments contextuels, combinée avec un ensemble de concepts. Ces derniers permettent de garantir un contrôle de cette conjonction des informations contextuelles. La situation est le point d'entrée des systèmes d'accès à l'information afin de contextualiser leurs processus informationnels. La situation est donc la photographie la plus fidèle possible d'une interprétation du contexte.

Par conséquent, nous avons proposé un GSC fondé sur une approche afin de générer les situations. La démarche de création de la situation est basée sur notre définition du contexte. Ainsi, chaque dimension est considérée séparément et adaptée selon le reste des dimensions : cycle d'adaptation. Dans cette situation tous les éléments contextuels ont une importance et peuvent être en relation les uns avec les autres en se basant sur des connaissances contextuelles supplémentaires (les règles). Le GSC se base sur le processus MES pour contrôler dans un premier temps la validité de la conjonction des éléments contextuels en utilisant les *règles métier légales*, puis dans un deuxième temps pour générer la situation en atteignant la stabilité de la dynamique des interactions entre les différents éléments contextuels qui la composent. Pour cela, le MES applique successivement les *règles métier* et les *règles inférées* sur les dimensions contextuelles jusqu'à ce qu'un point de stabilité à un instant t soit trouvé, c'est ce que nous appelons la situation.

L'approche de génération des situations que nous proposons permet au GSC, à travers le MES, d'injecter des connaissances supplémentaires dans les situations. Pour générer les situations, le MES porte davantage sur la dynamique de l'interaction entre toutes les dimensions contextuelles que sur la façon dont l'information est représentée. Par conséquent, nous avons ajouté à notre GSC le processus d'extraction de règles permettant d'avoir un feedback sur ces situations passées. Ce processus permet à notre GSC d'évoluer au fil du temps en utilisant des approches de fouilles de connaissances pour alimenter la base des règles du GSC avec des *règles inférées*. Le processus d'extraction de règles a pour objectif principal d'extraire les nouvelles relations qui pourraient exister dans ces situations et qui permettent aux systèmes d'analyser les activités passées.

Enfin, nous avons montré dans ce travail l'efficacité de notre GSC dans le contexte métier particulier, celui de la maintenance aéronautique. Pour cela, nous avons implanté un

prototype du GSC qui permet de proposer les situations aux systèmes d'accès à la documentation de la maintenance aéronautique. Dans ce domaine, les enjeux de sûreté, de légalité, d'efficacité et d'efficience sont importants. Les erreurs humaines liées à la non-utilisation d'un document sont présentes. Nous avons montré que le GSC permet de combler les défaillances de tels systèmes dans tous les domaines où l'enjeu de fiabilité est fort, en particulier le domaine de la maintenance aéronautique.

Limites et perspectives de recherche

La première limite de ce travail est le manque de données contextuelles pertinentes utilisées dans l'expérimentation. Le prototype du GSC doit pouvoir disposer d'informations contextuelles concrètes (dimensions, règles, actions, etc.) issues des contextes métiers réels afin de générer un grand nombre de situations qui serviront à affiner l'expérimentation. Ainsi, nous pourrions analyser dans un premier temps plus en détail l'impact de nos différentes propositions dans le domaine de la maintenance aéronautique. Dans un deuxième temps, de nouvelles expérimentations pourraient être réalisées dans d'autres cadres applicatifs pour montrer la généralité de notre approche.

Ensuite, une des limites du prototype est qu'il ne permet pas encore de réaliser l'objectif de proactivité. Il reste donc à implanter la méthode de détection des situations similaires pour permettre de proposer les actions adéquates. Pour cela, un travail d'étude et de comparaison des approches et techniques de la littérature est nécessaire pour atteindre cet objectif. Certaines pistes sont déjà considérées, comme celle de raisonnement à partir de cas.

De nombreuses perspectives s'offrent à la suite de nos travaux.

La première perspective à long terme concerne l'aspect du contexte lui-même. Autrement dit, nous avons présenté un GSC dédié au contexte métier dans lequel les éléments contextuels sont connus et fournis par les applications à l'initialisation du système. Notre gestionnaire n'est pas encore adapté aux environnements ubiquitaires dans lesquels les tâches ne sont pas prescrites et l'information contextuelle change continuellement. Dans ce type d'environnement, le GSC doit être plus réactif et moins dépendant des règles métier. La problématique principale qui se pose donc est de proposer une approche qui permettra au GSC d'inférer la nature de l'activité de l'utilisateur à partir des éléments contextuels de la

situation. Pour cela, nous nous tournerons vers les travaux présentés à la fin du chapitre I.1 et qui se focalisent sur les approches de détermination automatique des situations.

Nous pouvons intégrer d'autres méthodes d'extraction de règles parmi celles proposées dans la littérature après leur étude et comparaison. Le processus d'extraction de règles pourra ainsi choisir entre les différentes approches de fouilles de règles celles qui sont adaptées aux spécificités des dimensions contextuelles composant les situations.

Dans le même cadre, nous souhaitons améliorer le calcul des priorités des règles inférées. Nous pourrions considérer l'emploi d'autres outils statistiques permettant d'évaluer la qualité des règles d'association et ainsi de définir une méthode de calcul des priorités basée sur une leur pondération.

Une autre direction de recherche peut se concentrer sur la façon d'améliorer la gestion du cycle de stabilité des situations. Plus précisément, notre approche de création de situations ne permet pas au processus MES de traiter les situations instables. Le MES doit être capable de gérer ce genre de situations et pas seulement arrêter le cycle d'adaptation après un intervalle de temps prédéfini.

À l'image de l'extraction des règles associations à partir des situations, nous aimerions intégrer celle des motifs séquentiels pour donner une dimension temporelle aux règles extraites. Une valeur d'élément contextuel (ou une action) dans une situation à l'instant t de cette situation peut avoir un impact sur la prédiction d'une valeur d'un autre élément contextuel (ou l'anticipation d'une action) dans la situation à l'instant $t + n$. L'extraction des motifs séquentiels ne doit pas se limiter seulement à l'analyse des situations finales, mais pourrait s'intéresser également aux liens qui puissent exister entre les composants des états d'une même situation.

Un autre travail de réflexion doit concerner la question du calcul de la fiabilité des éléments contextuels. Dans notre approche lors de l'application des règles par le MES, la fiabilité d'un élément contextuel devient égale à la priorité de la règle qui l'adapte. Le problème se pose quand un élément contextuel est adapté plus d'une fois lors de la création d'une seule situation. Dans ce cas, nous devons approfondir avec des outils probabilistes la méthode à appliquer pour le calcul de la fiabilité. Elle ne doit donc pas être simplement une récupération triviale de la priorité de la dernière règle appliquée.

Un autre processus d'apprentissage pourrait être intégré aux GSC pour prendre en compte l'évolution (la mise à jour) des différentes instances des dimensions contextuelles. En effet, ce processus observerait les différentes situations passées afin d'identifier et répliquer une évolution du contenu des différents modèles détenus par le GSC. Cependant, ce type de processus dépendra du contexte métier et aura besoin de règles définies a priori par les experts du domaine pour mettre à jour les valeurs des éléments contextuels. Par exemple, si nous posons la règle : si un usager u effectue n fois la tâche métier T alors son expérience augmente. Le rôle du processus d'apprentissage est donc d'incrémenter l'expérience de l'utilisateur u pour cette tâche.

Enfin, une dernière piste de recherche peut exploiter les différentes actions entreprises par un utilisateur durant ses différentes activités. Nous pouvons ainsi nous fonder sur les études existantes des analyses de traces pour mettre une modélisation de l'usage du système par les utilisateurs. Cela permettra aux systèmes d'améliorer leurs processus d'adaptation en fonction des modèles de traces des utilisateurs en analysant la chronologie de toutes les actions réalisées dans les situations.

Bibliographie générale

- Abowd, G. D., Atkeson, C. G., Hong, J., Long, S., Kooper, R., & Pinkerton, M. (1997). Cyberguide: a mobile context-aware tour guide. *Wirel. Netw.*, 3(5), 421–433. doi:10.1023/A:1019194325861
- Adomavicius, G., & Tuzhilin, A. (2005). Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE Trans. on Knowl. and Data Eng.*, 17(6), 734–749. doi:10.1109/TKDE.2005.99
- Adomavicius, G., & Tuzhilin, A. (2008). Context-aware recommender systems. *Proceedings of the 2008 ACM conference on Recommender systems*, RecSys '08 (p. 335–336). New York, NY, USA: ACM. doi:10.1145/1454008.1454068
- Agrawal, R., Imielinski, T., & Swami, A. (1993). Mining Association Rules between Sets of Items in Large Databases. *ACM SIGMOD International Conference on Management of Data* (p. 207-216).
- Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., Verkamo, A. I., & others. (1996). Fast discovery of association rules. *Advances in knowledge discovery and data mining*, 12, 307–328.
- Agrawal, R., Srikant, R., & others. (1994). Fast algorithms for mining association rules. *Proc. 20th Int. Conf. Very Large Data Bases, VLDB* (Vol. 1215, p. 487–499).
- Ailisto, H., Alahuhta, P., Haataja, V., Kyllönen, V., & Lindholm, M. (2002). Structuring context aware applications: Five-layer model and example case. *Proceedings of the Workshop on Concepts and Models for Ubiquitous Computing*.
- Allen, B. (1997). Information needs: a person-in-situation approach. *Proceedings of an international conference on Information seeking in context* (p. 111-122). Tampere, Finland: Taylor Graham Publishing.
- Amodeo, G., Amati, G., & Gambosi, G. (2011). On relevance, time and query expansion. *Proceedings of the 20th ACM international conference on Information and knowledge management* (p. 1973–1976).
- Anagnostopoulos, C. B., Tsounis, A., & Hadjiefthymiades, S. (2007). Context awareness in mobile computing environments. *Wireless Personal Communications*, 42(3), 445-464.
- Anand, S., & Mobasher, B. (2007). Contextual Recommendation. *From Web to Social Web: Discovering and Deploying User and Content Profiles* (p. 142-160). Consulté de http://dx.doi.org/10.1007/978-3-540-74951-6_8
- Annett, J. (2003). Hierarchical task analysis. *Handbook of cognitive task design* (p. 17-35). CRC.

- Annett, J., & Duncan, K. D. (1967). Task analysis and training design. *Journal of Occupational Psychology*, 41, 211-221.
- Apolinarski, W., Handte, M., Le Phuoc, D., & Marrón, P. J. (2011). A peer-based approach to privacy-preserving context management. *Proceedings of the 7th international and interdisciplinary conference on Modeling and using context*, CONTEXT'11 (p. 18–25). Berlin, Heidelberg: Springer-Verlag. Consulté de <http://dl.acm.org/citation.cfm?id=2045502.2045505>
- Arapakis, I., Jose, J. M., & Gray, P. D. (2008). Affective feedback: an investigation into the role of emotions in the information seeking process. *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval* (p. 395-402). Singapore, Singapore: ACM.
- Baeza-Yates, R., & Ribeiro, B. de A. N. (1999). *Modern information retrieval*. Pearson, Addison-Wesley.
- Bai, J., & Nie, J.-Y. (2008). Adapting information retrieval to query contexts. *Information Processing & Management*, 44(6), 1901–1922. doi:<http://dx.doi.org/10.1016/j.ipm.2008.07.006>
- Balabanović, M., & Shoham, Y. (1997). Fab: content-based, collaborative recommendation. *Communications of the ACM*, 40(3), 66–72.
- Baldauf, M., Dustdar, S., & Rosenberg, F. (2007). A survey on context-aware systems. *International Journal of Ad Hoc and Ubiquitous Computing*, 2(4), 263 - 277. doi:10.1504/IJAHUC.2007.014070
- Bazire, M., & Brézillon, P. (2005). Understanding context before using it. *Modeling and Using Context*, 29–40.
- Belkin, N. (1993). Interaction with Texts: Information Retrieval as Information-Seeking Behavior. *UNIVERSITÄTSVERLAG KONSTANZ*, 55-66.
- Belkin, N. J. (1996). Intelligent information retrieval: Whose intelligence. *ISI*, 96, 25–31.
- Belkin, N. J. (2000). The human element: helping people find what they don't know. *Commun. ACM*, 43(8), 58-61.
- Belkin, N. J. (2008). Some(what) grand challenges for information retrieval. *SIGIR Forum*, 42(1), 47-54. doi:10.1145/1394251.1394261
- Belkin, N. J., & Croft, W. B. (1992). Information filtering and information retrieval: two sides of the same coin? *Commun. ACM*, 35(12), 29–38. doi:10.1145/138859.138861

- Belkin, N. J., Muresan, G., & Zhang, X. M. (2004). Using user's context for ir personalization. *Proceedings of the ACM/SIGIR Workshop on Information Retrieval in Context*.
- Beltran, T., & Ravat, F. (2009). Entrepôt de contextes. *Congrès Informatique des Organisations et Systèmes d'Information et de Décision (INFORSID), Toulouse, 26/05/09-29/05/09* (p. 429–444). <http://inforsid.irit.fr>: INFORSID.
- Bettini, C., Brdiczka, O., Henriksen, K., Indulska, J., Nicklas, D., Ranganathan, A., & Riboni, D. (2010). A survey of context modelling and reasoning techniques. *Pervasive and Mobile Computing*, 6(2), 161-180. doi:10.1016/j.pmcj.2009.06.002
- Bolchini, C., Curino, C. A., Quintarelli, E., Schreiber, F. A., & Tanca, L. (2007). A data-oriented survey of context models. *ACM SIGMOD Record*, 36(4), 19–26.
- Borlund, P. (2003). The concept of relevance in IR. *J. Am. Soc. Inf. Sci. Technol.*, 54(10), 913–925. doi:10.1002/asi.10286
- Brézillon, P., Pasquier, L., & Pomerol, J. C. (2002). Reasoning with contextual graphs. *European Journal of Operational Research*, 136(2), 290–298.
- Brin, S., Motwani, R., Ullman, J. D., & Tsur, S. (1997). Dynamic itemset counting and implication rules for market basket data. *ACM SIGMOD Record* (Vol. 26, p. 255–264).
- Brown, P. J., & Jones, G. J. F. (2002). Exploiting contextual change in context-aware retrieval. *Proceedings of the 2002 ACM symposium on Applied computing, SAC '02* (p. 650–656). New York, NY, USA: ACM. doi:10.1145/508791.508917
- Bruegger, P., Lalanne, D., Lisowska, A., & Hirsbrunner, B. (2009). Tools for designing and prototyping activity-based pervasive applications. *Proceedings of the 7th International Conference on Advances in Mobile Computing and Multimedia* (p. 129–136).
- Brusilovsky, P. (1996). Methods and Techniques of Adaptive Hypermedia. Consulté de <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.53.8848>
- Brusilovsky, P., & Millán, E. (2007). User Models for Adaptive Hypermedia and Adaptive Educational Systems. *The Adaptive Web* (p. 3-53).
- Budzik, Jay, & Hammond, K. J. (2000). User interactions with everyday applications as context for just-in-time information access (p. 44-51). New Orleans, Louisiana, United States: ACM. doi:10.1145/325737.325776
- Budzik, J., Hammond, K. J., & Birnbaum, L. (2001). Information access in context. *Knowledge-based systems*, 14(1-2), 37–53.

- Burke, R. (2002). Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*, 12(4), 331–370. doi:10.1023/A:1021240730564
- Byström, K. (2002). Information and information sources in tasks of varying complexity. *J. Am. Soc. Inf. Sci. Technol.*, 53(7), 581–591. doi:http://dx.doi.org/10.1002/asi.10064
- Byström, K., & Hansen, P. (2005). Conceptual framework for tasks in information studies: Book Reviews. *J. Am. Soc. Inf. Sci. Technol.*, 56(10), 1050–1061. doi:http://dx.doi.org/10.1002/asi.v56:10
- Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction*. Erlbaum.
- Chaker, H., Chevalier, M., Soulé-Dupuy, C., & Tricot, A. (2010a). Gestionnaire de contexte métier : application à la recherche d'information. *Colloque Veille Stratégique Scientifique et Technologique (VSST 2010)*. Toulouse.
- Chaker, H., Chevalier, M., Soulé-Dupuy, C., & Tricot, A. (2010b). Adapter le contexte métier pour améliorer le système de recherche d'information. *Prise en Compte de l'Usager dans les Systèmes d'Information (INFORSID) (Atelier PeCUSI 2010)*. Marseille.
- Chaker, H., Chevalier, M., Soulé-Dupuy, C., & Tricot, A. (2010c). Système de recherche d'information pour les tâches métier. *CONFérence en Recherche d'Infomations et Applications* (p. 301-312). Sousse, Tunisia.
- Chaker, H., Chevalier, M., Soulé-Dupuy, C., & Tricot, A. (2010d). Improving information retrieval by modelling business context (p. 117–122). Présenté à Advances in Human-Oriented and Personalized Mechanisms, Technologies and Services (CENTRIC), 2010 Third International Conference on, Nice, France.
- Chaker, H., Chevalier, M., Soulé-Dupuy, C., & Tricot, A. (2011a). Business Context Information Manager: An Approach to Improve Information Systems (Vol. LNAI 6967, p. 67-70). Présenté à Context 2011, Karlsruhe.
- Chaker, H., Chevalier, M., Soulé-Dupuy, C., & Tricot, A. (2011b). Business Context Information Manager: application to Information Retrieval (p. 67-70). Présenté à International Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services (CENTRIC 2011), Barcelone, Espagne.
- Chen, H., Perich, F., Finin, T., & Joshi, A. (2004). SOUPA: Standard ontology for ubiquitous and pervasive applications.

- Cheverst, K., Davies, N., Mitchell, K., Friday, A., & Efstratiou, C. (2000). Developing a context-aware electronic tourist guide: some issues and experiences. *Proceedings of the SIGCHI conference on Human factors in computing systems* (p. 17–24).
- Civil Aviation Authority (2002). *Human Factors in Aircraft Maintenance and Inspection*. (Report No. CAP 718) London : The Safety Regulation Group.
- Conan, D., Rouvoy, R., & Seinturier, L. (2007). Scalable processing of context information with cosmos. *Distributed Applications and Interoperable Systems* (p. 210–224).
- Cool, C. (2001). The concept of situation in information science. *Annual review of information science and technology*, 35, 5-42.
- Cool, C., & Spink, A. (2002). Issues of context in information retrieval (IR): an introduction to the special issue. *Information Processing & Management*, 38(5), 605-611. doi:10.1016/S0306-4573(01)00054-1
- Costa, P. D., Almeida, J. P. A., Pires, L. F., & Van Sinderen, M. (2007). Situation specification and realization in rule-based context-aware applications. *Proceedings of the 7th IFIP WG 6.1 international conference on Distributed applications and interoperable systems* (p. 32–47).
- Coutaz, J., Crowley, J. L., Dobson, S., & Garlan, D. (2005). The Disappearing Computer-Context is Key. *Communications of the ACM-Association for Computing Machinery-CACM*, 48(3), 49–53.
- Crestani, F., & Ruthven, I. (2007). Introduction to special issue on contextual information retrieval systems. *Inf. Retr.*, 10(2), 111–113. doi:10.1007/s10791-007-9022-z
- De Virgilio, R., & Torlone, R. (2005). A general methodology for context-aware data access. *Proceedings of the 4th ACM international workshop on Data engineering for wireless and mobile access, MobiDE '05* (p. 9–15). New York, NY, USA: ACM. doi:10.1145/1065870.1065873
- Dey, A. K. (2001). Understanding and Using Context. *Personal Ubiquitous Comput.*, 5(1), 4-7.
- Dey, A. K., Abowd, G. D., & Salber, D. (2001). A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Human-Computer Interaction*, 16, 97–166. doi:http://dx.doi.org/10.1207/S15327051HCI16234_02
- Diaper, D., & Stanton, N. (2004). *The handbook of task analysis for human-computer interaction*. Lawrence Erlbaum Associates.

- Dourish, P. (2004). What we talk about when we talk about context. *Personal Ubiquitous Comput.*, 8(1), 19-30.
- Fahy, P., & Clarke, S. (2004). CASS—a middleware for mobile context-aware applications. *Workshop on Context Awareness, MobiSys*.
- Fayyd, U. M., Shapiro, G. P., & Smyth, P. (1996). From data mining to knowledge discovery: an overview. *Advances in knowledge discovery and data mining* (p. 1-34).
- Freund, L. (2008). Exploiting task-document relations in support of information retrieval in the workplace. *SIGIR Forum*, 42(2), 107–107. doi:10.1145/1480506.1480529
- Freund, L., & Toms, E. G. (2005). Contextual search: from information behaviour to information retrieval. *Proceedings of the Annual Conference of the Canadian Association for Information Science*.
- Freund, L., Toms, E. G., & Waterhouse, J. (2005). Modeling the information behaviour of software engineers using a work - task framework. *Proceedings of the 68th annual meeting of the American Society for Information Science and Technology*, 42(1), NA. doi:10.1002/meet.14504201181
- Fuhr, N. (2000). *Information retrieval: introduction and survey. post-graduate course on information retrieval, university of Duisburg-Essen. Germany*.
- Göker, A., Myrhaug, H., & Bierig, R. (2009). Context and information retrieval. *Information Retrieval: Searching in the 21st Century. John Wiley and Sons, Ltd, Chichester, UK*.
- Göker, Ayşe, & Myrhaug, H. (2008). Evaluation of a mobile information system in context. *Inf. Process. Manage.*, 44(1), 39–65. doi:10.1016/j.ipm.2007.03.011
- Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12), 61–70.
- Gwizdka, J., & Lopatovska, I. (2009). The role of subjective factors in the information search process. *J. Am. Soc. Inf. Sci. Technol.*, 60(12), 2452–2464. doi:10.1002/asi.v60:12
- Halder, S., Roy, A., & Chakraborty, P. (2010). The influence of personality traits on information seeking behaviour of students. *Malaysian Journal of Library & Information Science*, 15(1), 41–53.
- Han, J., Kamber, M., & Pei, J. (2011). *Data mining: concepts and techniques*. Morgan Kaufmann Pub.
- Henricksen, K., Indulska, J., & Rakotonirainy, A. (2003). Generating context management infrastructure from high-level context models. *Proceedings of the 4th International Conference on Mobile Data Management (MDM'03)* (p. 1–6).

- Hirsh, S. G. (2000). Information Needs, Information Seeking, and Communication in an Industrial R&D Environment. *Proceedings of the ASIS Annual Meeting*, 37, 473-86.
- Hobbs, A. (2000). Maintenance error, lessons from the BASI survey. *Flight Safety Australia*, 4, 36-37.
- Hofer, T., Schwinger, W., Pichler, M., Leonhartsberger, G., Altmann, J., & Retschitzegger, W. (2003). Context-awareness on mobile devices-the hydrogen approach. *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on* (p. 10–pp).
- Hyldegård, J. (2009). Personality traits and group-based information behaviour: an exploratory study. *Information Research*, 14(2), paper 402.
- Indulska, J., & Sutton, P. (2003). Location management in pervasive systems. *Proceedings of the Australasian information security workshop conference on ACSW frontiers 2003-Volume 21* (p. 143–151).
- Ingwersen, P. (1992). *Information retrieval interaction*. Taylor Graham London.
- Ingwersen, P., & Järvelin, K. (2005a). Information retrieval in context: IRiX. *SIGIR Forum*, 39(2), 31-39. doi:10.1145/1113343.1113351
- Ingwersen, P., & Järvelin, K. (2005b). *The Turn: Integration of Information Seeking and Retrieval in Context (The Information Retrieval Series)*. Springer-Verlag New York, Inc.
- Jameson, A. (2001). Modelling both the Context and the User. *Personal Ubiquitous Comput.*, 5(1), 29-33.
- Johnson, H., & Johnson, P. (1991). Task knowledge structures: Psychological basis and integration into system design. *Acta Psychologica*, 78(1–3), 3-26. doi:10.1016/0001-6918(91)90003-I
- Johnson, J. D. (2003). On contexts of information seeking. *Information Processing & Management*, 39(5), 735-760. doi:10.1016/S0306-4573(02)00030-4
- Jones, G. J. F. (2005). Challenges and Opportunities of Context-Aware Information Access. *Proceedings of the International Workshop on Ubiquitous Data Management, UDM '05* (p. 53–62). Washington, DC, USA: IEEE Computer Society. doi:10.1109/UDM.2005.5
- Kavitha, V. (2009). Context aware approach for smart homes. *Proceedings of the 8th WSEAS international conference on Data networks, communications, computers* (p. 225–230).

- Kelly, D. (2006). Measuring online information seeking context, Part 1: Background and method. *J. Am. Soc. Inf. Sci. Technol.*, 57(13), 1729-1739.
- Kim, K.-S. (2008). Effects of emotion control and task on Web searching behavior. *Information Processing & Management*, 44(1), 373–385. doi:10.1016/j.ipm.2006.11.008
- Kiss, C. (2006). Composite capability/preference profiles (cc/pp): Structure and vocabularies 2.0. *W3C Working Draft*, 8.
- Kobsa, A. (2001). Generic User Modeling Systems. *User Modeling and User-Adapted Interaction*, 11(1), 49–63.
- Kobsa, A. (2007). Generic User Modeling Systems (p. 136-154).
- Kostadinov, D., Bouzeghoub, M., & Lopes, S. (2007). Accès personnalisé à des sources de données multiples: évaluation de deux approches de reformulation de requêtes (p. 73-88).
- Koutrika, G., & Ioannidis, Y. (2005). Personalized Queries under a Generalized Preference Model. *Proceedings of the 21st International Conference on Data Engineering, ICDE '05* (p. 841–852). Washington, DC, USA: IEEE Computer Society. doi:10.1109/ICDE.2005.106
- Krüger, A., Baus, J., Heckmann, D., Kruppa, M., & Wasinger, R. (2007). Adaptive mobile guides. *The adaptive web* (p. 521–549).
- Kumaran, G., & Allan, J. (2008). Adapting information retrieval systems to user queries. *Information Processing & Management*, 44(6), 1838-1862. doi:10.1016/j.ipm.2007.12.006
- Lang, K. (1995). Newsweeder: Learning to filter netnews. *In Proceedings of the Twelfth International Conference on Machine Learning*.
- Leal, A., Bouchet, J., Langlois, G., & Jourde, F. (2009). Conception de l'interaction homme-machine et partage d'autorité: application aux systèmes de drones. *Proceedings of the 21st International Conference on Association Francophone d'Interaction Homme-Machine* (p. 283-290). Grenoble, France: ACM. doi:10.1145/1629826.1629872
- Li, Y. (2009). Exploring the relationships between work task and search task in information search. *Journal of the American Society for Information Science and Technology*, 60(2).

- Li, Y., & Belkin, N. J. (2008). A faceted approach to conceptualizing tasks in information seeking. *Information Processing & Management*, 44(6), 1822–1837. doi:<http://dx.doi.org/10.1016/j.ipm.2008.07.005>
- Li, Y., & Landay, J. A. (2008). Activity-based prototyping of ubicomp applications for long-lived, everyday human activities. *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems* (p. 1303–1312).
- Lieberman, H., & Selker, T. (2000). Out of context: computer systems that adapt to, and learn from, context. *IBM Syst. J.*, 39(3-4), 617-632.
- Liu, B. (2007). *Web data mining*. Springer.
- Liu, J., & Belkin, N. J. (2010a). Personalizing information retrieval for multi-session tasks: the roles of task stage and task type. *Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval* (p. 26-33). Geneva, Switzerland: ACM. doi:10.1145/1835449.1835457
- Liu, J., & Belkin, N. J. (2010b). Personalizing information retrieval for people with different levels of topic knowledge. *Proceedings of the 10th annual joint conference on Digital libraries* (p. 383-384). Gold Coast, Queensland, Australia: ACM. doi:10.1145/1816123.1816191
- Loke, S. W. (2004). Representing and reasoning with situations for context-aware pervasive computing: a logic programming perspective. *The Knowledge Engineering Review*, 19(03), 213–233.
- Marques, O., Barenholtz, E., & Charvillat, V. (2011). Context modeling in computer vision: techniques, implications, and applications. *Multimedia Tools and Applications*, 1–37.
- McCarthy, J., & Buvac, S. (1998). Formalizing context (expanded notes). *Computing natural language*, 81, 13–50.
- Melucci, M. (2005). Context modeling and discovery using vector space bases. *Proceedings of the 14th ACM international conference on Information and knowledge management, CIKM '05* (p. 808–815). New York, NY, USA: ACM. doi:10.1145/1099554.1099745
- Montaner, M., López, B., & De La Rosa, J. L. (2003). A Taxonomy of Recommender Agents on the Internet. *Artif. Intell. Rev.*, 19(4), 285–330. doi:10.1023/A:1022850703159
- Öztürk, P., & Aamodt, A. (1997). Towards a model of context for case-based diagnostic problem solving. *Context-97; Proceedings of the interdisciplinary conference on modeling and using context* (p. 198–208).

- Park, T. K. (1994). Toward a theory of user-based relevance: a call for a new paradigm of inquiry. *J. Am. Soc. Inf. Sci.*, 45(3), 135–141. doi:10.1002/(SICI)1097-4571(199404)45:3<135::AID-ASI3>3.0.CO;2-1
- Pascoe, J. (1998). Adding generic contextual capabilities to wearable computers. *Proceedings of the 2nd International Symposium on Wearable Computers* (p. 92-99).
- Paterno, F. (2000). Model-based design of interactive applications. *intelligence*, 11(4), 26-38.
- Pazzani, M., & Billsus, D. (1997). Learning and revising user profiles: The identification of interesting web sites. *Machine learning*, 27(3), 313–331.
- Pomerol, J. C., & Brézillon, P. (2001). About some relationships between Knowledge and Context. *Modeling and Using Context*, 461–464.
- Reason, J., & Hobbs, A. 2003. Managing maintenance error: a practical guide. Aldershot: Ashgate.
- Reason, J., & Hobbs, A. (2006). Managing maintenance error: a practical guide. Aldershot: Ashgate.
- Reichenbacher, T. (2007). Adaptation in mobile and ubiquitous cartography (p. 383-397).
- Rey, G. (2005). *Contexte en interaction homme-machine: le contexteur*. Université Joseph Fourier - Grenoble I.
- Ricci, F., Rokach, L., Shapira, B., & Kantor, P. B. (Éd.). (2011). *Recommender Systems Handbook*. Boston, MA: Springer US. Consulté de <http://www.springerlink.com/content/v24503855k56g500/>
- Rijsbergen, C. J. V. (1979). *Information Retrieval* (2nd éd.). Newton, MA, USA: Butterworth-Heinemann.
- Robertson, S. E., & Jones, K. S. (1976). Relevance weighting of search terms. *Journal of the American Society for Information science*, 27(3), 129–146.
- Ruthven, I. (2008). The context of the interface (p. 3-5). London, United Kingdom: ACM. doi:10.1145/1414694.1414697
- Salton, G. (1971). *The SMART Retrieval System-Experiments in Automatic Document Processing*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc.
- Salton, Gerard, & McGill, M. J. (1986). *Introduction to Modern Information Retrieval*. New York, NY, USA: McGraw-Hill, Inc.
- Scapin, D., & Pierret-Golbreich, C. (1990). Towards a method for task description: MAD. *Work with display units*, 89, 371–380.

- Schafer, J. B., Konstan, J., & Riedi, J. (1999). Recommender systems in e-commerce. *Proceedings of the 1st ACM conference on Electronic commerce* (p. 158–166).
- Schilit, B. N., Adams, N., Gold, R., Tso, M. M., & Want, R. (1993). The PARCTAB mobile computing system. *Workstation Operating Systems, 1993. Proceedings., Fourth Workshop on* (p. 34–39).
- Schilit, B. N., Adams, N., & Want, R. (1994). Context-aware computing applications. *First Workshop on Mobile Computing Systems and Applications* (p. 85-90).
- Schmidt, A. (2000). Implicit human computer interaction through context. *Personal and Ubiquitous Computing*, 4(2), 191–199.
- Schmidt, A., Aidoo, K. A., Takaluoma, A., Tuomela, U., Van Laerhoven, K., & Van de Velde, W. (1999). Advanced interaction in context. *Handheld and Ubiquitous Computing: First International Symposium, Huc'99, Karlsruhe, Germany, September 27-29, 1999, Proceedings* (p. 89). Springer.
- Schmidt, A., Beigl, M., & Gellersen, H. W. (1999). There is more to context than location. *Computers & Graphics*, 23(6), 893-901. doi:10.1016/S0097-8493(99)00120-X
- Schroeder, B. A. (1995). On-line monitoring: A tutorial. *Computer*, 28(6), 72–78.
- Shardanand, U., & Maes, P. (1995). Social information filtering: algorithms for automating « word of mouth ». *Proceedings of the SIGCHI conference on Human factors in computing systems* (p. 210–217).
- Sheng, Q. Z., & Benatallah, B. (2005). ContextUML: a UML-based modeling language for model-driven development of context-aware web services. *Mobile Business, 2005. ICMB 2005. International Conference on* (p. 206–212).
- Soule-Dupuy, C. (2001). *Bases d'informations textuelles : des modèles aux applications*. (Habilitation à diriger des recherches). Université Paul Sabatier.
- Spink, A., & Cole, C. (2001). Introduction to the special issue: Everyday life information-seeking research. *Library & Information Science Research*, 23(4), 301-304. doi:10.1016/S0740-8188(01)00090-1
- Spink, A., Jansen, B. J., Wolfram, D., & Saracevic, T. (2002). From E-Sex to E-Commerce: Web Search Changes. *Computer*, 35(3), 107-109.
- Stefanidis, K., & Pitoura, E. (2008). Fast contextual preference scoring of database tuples. *Proceedings of the 11th international conference on Extending database technology: Advances in database technology, EDBT '08* (p. 344–355). New York, NY, USA: ACM. doi:10.1145/1353343.1353387

- Stojanovic, N. (2005). On the role of a user's knowledge gap in an information retrieval process. *Proceedings of the 3rd international conference on Knowledge capture, K-CAP '05*, 83–90. doi:10.1145/1088622.1088638
- Strang, T., & Linnhoff-Popien, C. (2004). A context modeling survey. *Workshop on Advanced Context Modelling Reasoning and Management as part of UbiComp* (p. 1–8).
- Tamine-Lechani, L., Boughanem, M., & Daoud, M. (2010). Evaluation of contextual information retrieval effectiveness: overview of issues and research. *Knowl. Inf. Syst.*, 24(1), 1–34. doi:10.1007/s10115-009-0231-1
- Tarby, J. C., & Barthet, M. F. (1996). the Diane+ method. *Computer-Aided Design of User Interfaces* (p. 5–7).
- Taylor, A. R., Cool, C., Belkin, N. J., & Amadio, W. J. (2007). Relationships between categories of relevance criteria and stage in task completion. *Inf. Process. Manage.*, 43(4), 1071–1084. doi:10.1016/j.ipm.2006.09.008
- Taylor, R. S. (1991). Information use environments. *Progress in communication sciences* (Vol. 10, p. 217, 255). Ablex Publishing Corporation.
- Tenopir, C., Wang, P., Zhang, Y., Simmons, B., & Pollard, R. (2008). Academic users' interactions with ScienceDirect in search tasks: Affective and cognitive behaviors. *Inf. Process. Manage.*, 44(1), 105–121. doi:10.1016/j.ipm.2006.10.007
- Thomson, G., Terzis, S., & Nixon, P. (2006). Situation determination with reusable situation specifications. *Pervasive Computing and Communications Workshops, 2006. PerCom Workshops 2006. Fourth Annual IEEE International Conference on* (p. 4–pp).
- Tricot, A., & Nanard, J. (1998). Un point sur la modélisation des tâches de recherche d'informations dans le domaine des hypermédias. *Les hypermédias, approches cognitives et ergonomiques*, 3(4), 35-56.
- Truong, H. L., & Dustdar, S. (2009). A survey on context-aware web service systems. *International Journal of Web Information Systems*, 5(1), 5–31.
- Van Der Veer, G. C., Lenting, B. F., & Bergevoet, B. A. . (1996). GTA: Groupware task analysis—Modeling complexity. *Acta Psychologica*, 91(3), 297–322.
- Vilar, P., & Žumer, M. (2008). Perceptions and importance of user friendliness of IR systems according to users' individual characteristics and academic discipline. *Journal of the American Society for Information Science and Technology*, 59(12), 1995–2007.

- Vildjiounaite, E., Kocsis, O., Kyllönen, V., & Kladis, B. (2009). Context-dependent user modelling for smart homes. *User Modeling 2007*, 345–349.
- Watters, C., & Amoudi, G. (2003). GeoSearcher: location-based ranking of search engine results. *J. Am. Soc. Inf. Sci. Technol.*, 54(2), 140–151. doi:10.1002/asi.10191
- Wen, J.-R., Lao, N., & Ma, W.-Y. (2004). Probabilistic model for contextual retrieval. *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04 (p. 57–63). New York, NY, USA: ACM. doi:10.1145/1008992.1009005
- Weyhrauch, R. (1979). Prolegomena to a Theory of Mechanized Formal Reasoning. *Artificial Intelligence*, 13(1--2), 170, 133.
- White, R. W., Bailey, P., & Chen, L. (2009). Predicting user interests from contextual information. *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval* (p. 363–370).
- Xie, H. (2006). Understanding human-work domain interaction: Implications for the design of a corporate digital library. *Journal of the American Society for Information Science and Technology(Print)*, 57(1), 128–143.
- Xie, H. I. (2000). Shifts of interactive intentions and information-seeking strategies in interactive information retrieval. *J. Am. Soc. Inf. Sci.*, 51(9), 841–857. doi:http://dx.doi.org/10.1002/(SICI)1097-4571(2000)51:9<841::AID-ASI70>3.0.CO;2-0
- Yau, S. S., Huang, D., Gong, H., & Yao, Y. (2006). Support for situation awareness in trustworthy ubiquitous computing application software. *Software: Practice and Experience*, 36(9), 893–921.
- Ye, J., Dobson, S., & McKeever, S. (2011). Situation identification techniques in pervasive computing: A review. *Pervasive and Mobile Computing*.
- Zafiharimalala, H. (2011). *Etude ergonomique pour la consultation sur écran de petite taille de la documentation de maintenance aéronautique* (Thèse de doctorat en Psychologie). Université Toulouse II – Le Mirail, Toulouse.

Annexe

I. Exemple de contenu d'une procédure de maintenance aéronautique

** ON A/C ALL

TASK 52-30-00-010-801-A

Opening of the FWD and AFT Cargo-Compartment Doors with the Yellow Electric Pump

WARNING: MAKE SURE THAT THE TRAVEL RANGES OF THE FLIGHT CONTROL SURFACES ARE CLEAR BEFORE YOU PRESSURIZE/DEPRESSURIZE A HYDRAULIC SYSTEM. MOVEMENT OF THE FLIGHT CONTROL SURFACES CAN BE DANGEROUS AND/OR CAUSE DAMAGE.

CAUTION: DO NOT OPEN THE DOOR IF THE WIND SPEED IS MORE THAN 40 KNOTS TO PREVENT DAMAGE TO THE DOOR OR TO THE AIRCRAFT STRUCTURE. IF THE AIRCRAFT IS PUT INTO THE WIND THE MAXIMUM WIND SPEED CAN BE 60 KNOTS.

CAUTION: YOU MUST CLOSE THE DOOR BEFORE THE WIND SPEED IS MORE THAN 60 KNOTS TO PREVENT DAMAGE TO THE DOOR OR TO THE AIRCRAFT STRUCTURE.

CAUTION: WHEN YOU OPEN AND CLOSE THE CARGO DOORS, MAKE SURE THAT THE ACCESS PLATFORM IS AT THE CORRECT HEIGHT. IF IT IS TOO HIGH, IT WILL PREVENT FREE MOVEMENT OF THE DOORS.

1. Reason for the Job

Self Explanatory

2. Job Set-up Information

A. Fixtures, Tools, Test and Support Equipment

REFERENCE	QTY	DESIGNATION
No specific	AP	ACCESS PLATFORM 3M (10 FT)- ADJUSTABLE

B. Work Zones and Access Panels

ZONE/ACCESS	ZONE DESCRIPTION
122CR, 152NR	

C. Referenced Information

REFERENCE	DESIGNATION
12-12-29-811-805-A	Fill the Yellow Hydraulic Reservoir with a Hydraulic Service Cart
24-41-00-861-801-A	Energize the Aircraft Electrical Circuits from the External Power A
24-41-00-861-801-A-01	Energize the Aircraft Electrical Circuits from the APU (APU Started with APU BAT)
24-41-00-861-801-A-03	Energize the Aircraft Electrical Circuits from the External Power A and B
24-41-00-861-801-A-04	Energize the Aircraft Electrical Circuits from the APU (APU started with External power A)
24-41-00-861-801-A-05	Energize the Aircraft Electrical Circuits from the External Power B
24-41-00-862-801-A	De-energize the Aircraft Electrical Circuits from the External Power A
24-41-00-862-801-A-01	De-energize the Aircraft Electrical Circuits from the APU
24-41-00-862-801-A-03	De-energize the Aircraft Electrical Circuits from the External Power A and B
24-41-00-862-801-A-04	De-energize the Aircraft Electrical Circuits from the External Power B
29-14-00-814-804-A	Pressurization of the Hydraulic Reservoirs with external bench
29-14-00-814-804-A-01	Pressurization of the Hydraulic Reservoirs with the APU
52-30-00-680-801-A	Installation of the Safety Support Equipment
Normal Operation of the FWD Cargo-Compartment Door SHEET 1	
Normal Operation of the AFT Cargo-Compartment Door SHEET 1	

3. Job Set-up

Subtask 52-30-00-860-053-A

A. Aircraft Maintenance Configuration

- (1) Make sure that the reservoir of the Yellow hydraulic system is pressurized. (Ref. AMM TASK 29-14-00-814-804)
- (2) Make sure that the level of the hydraulic fluid in the reservoir is correct. (Ref. AMM TASK 12-12-29-811-805)
- (3) Energize the aircraft electrical circuits. (Ref. AMM TASK 24-41-00-861-801)

Subtask 52-30-00-885-058-A

II. Décomposition des tâches en deux groupes en se basant sur leur nature

Level 2&3 TNA Process

1) Classify and Group Maintenance Tasks

LEVEL 2 AMM Family Tasks							
MP	C	S	I/C	D/R	R/I	T	MMEL (o)
Maintenance Practices	Cleaning	Servicing	Inspection /Check	Deactivation /Reactivation	Removal/ Installation	Test	
LEVEL 3 AMM Family Tasks							
MP	C/P	S	I/C	R/I	A/T	AR	
Maintenance Practices	Cleaning /Painting	Servicing	Inspection /Check	Removal/ Installation	Adjustment /Test	Approved Repairs	

III. Tableaux des différentes valeurs des éléments composant les vingt situations récoltées avec l'aide des psycho-cogniticien

	Situation1	Situation2	Situation3	Situation4	Situation5	Situation6
Identifiant	Expert1	Expert11	Expert12	Expert13	Expert14	Expert17
Rôle métier	Ingénieur expert fonction maintenance	Point fixeur	Point fixeur	Instructeur	instructeur	maintenance line et hangar A320/A330, essais OMS A380
Expertise	Expert	Expert	Expert	Expert	Expert	Expert
Qualification	B2	B2	B2	B2	B2	B2
Météo		Ciel clair	Ciel clair			
Vent (Km/h)		85				
Heure/Date	09h 42 17/12/09	14h18 02/02/10	15h00 02/02/10	14h23 07/04/10	13h53 07/04/10	09h18 19/05/10
Niveau sonore	Bas	Moyen	Moyen	Bas	Bas	Bas
Milieu		?				
Ecran	PC	PDA	PDA	PDA	PDA	PC
Niveau de luminosité	Moyen	Moyen	Moyen	Moyen	Moyen	Moyen
Nom de la tâche	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal
Type	non programmée	non programmée	non programmée	non programmée	non programmée	non programmée
Nature	R/I	R/I	R/I	R/I	R/I	R/I
Fréquence	Fréquente	Fréquente	Fréquente	Fréquente	Fréquente	Fréquente
Temps d'exécution (s)	147	297	283	144	396	198
Exigence	Exigeante	Exigeante	Exigeante	Exigeante	Exigeante	Exigeante
Saute des étapes	Non	Non	Non	Non	Non	Non
Durée de réalisation par rapport à la durée de référence (=291s)	Inférieure	Supérieure	Inférieur	Inférieur	Supérieure	Inférieure

	Situation7	Situation8	Situation9	Situation10	Situation11	Situation12
Identifiant	Nov9	Nov10	Nov11	Nov12	Nov13	Nov14
Rôle métier	élève BTS maintenance aéronautique	élève BTS maintenance aéronautique	élève BTS maintenance aéronautique	élève BTS maintenance aéronautique	élève BTS maintenance aéronautique	élève BTS maintenance aéronautique
Expertise	Novice	Novice	Novice	Novice	Novice	Novice
Qualification						
Météo						
Vent (Km/h)						
Heure/Date	20/11/10	20/11/10	20/11/10	20/11/10	20/11/10	20/11/10
Niveau sonore	Bas	Bas	Bas	Bas	Bas	Bas
Milieu						
Ecran	PC	PC	PC	PDA	PDA	PDA
Niveau de luminosité	Moyen	Moyen	Moyen	Moyen	Moyen	Moyen
Nom de la tâche	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal
Type	non programmée	non programmée	non programmée	non programmée	non programmée	non programmée
Nature	R/I	R/I	R/I	R/I	R/I	R/I
Fréquence	Fréquente	Fréquente	Fréquente	Fréquente	Fréquente	Fréquente
Temps d'exécution (s)	188	1050	2864	769	263	588
Exigence	Exigeante	Exigeante	Exigeante	Exigeante	Exigeante	Exigeante
Saute des étapes	Non	Non	Oui	Oui	Non	Oui
Durée de réalisation par rapport à la durée de référence (=291s)	inférieure	supérieure	supérieure	supérieure	inférieure	supérieure

	Situation13	Situation14	Situation15	Situation16	Situation17	Situation18
Identifiant	Nov15	Nov16	Nov17	Nov18	Nov19	Nov20
Rôle métier	élève BTS maintenance aéronautique	élève BTS maintenance aéronautique	élève BTS maintenance aéronautique	élève BTS maintenance aéronautique	élève BTS maintenance aéronautique	élève BTS maintenance aéronautique
Expertise	Novice	Novice	Novice	Novice	Novice	Novice
Qualification						
Météo						
Vent (Km/h)						
Heure/Date	20/11/10	20/11/10	30/11/10	30/11/10	02/12/10	02/12/10
Niveau sonore	Bas	Bas	Bas	Bas	Bas	Bas
Milieu						
Ecran	PDA	PC	PDA	PDA	PDA	PDA
Niveau de luminosité	Moyen	Moyen	Moyen	Moyen	Moyen	Moyen
Nom de la tâche	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal
Type	non programmée	non programmée	non programmée	non programmée	non programmée	non programmée
Nature	R/I	R/I	R/I	R/I	R/I	R/I
Fréquence	Fréquente	Fréquente	Fréquente	Fréquente	Fréquente	Fréquente
Temps d'exécution (s)	1434	3033	229	165	1128	1241
Exigence	Exigeante	Exigeante	Exigeante	Exigeante	Exigeante	Exigeante
Saute des étapes	Non	Non	Oui	Non	Oui	Non
Durée de réalisation par rapport à la durée de référence (=291s)	supérieure	supérieure	inférieure	inférieure	supérieure	supérieure

	Situation19	Situation20	Situation21	Situation22	Situation23
Identifiant	Nov21	Nov22	Nov23	Nov24	Nov25
Rôle métier	élève BTS maintenance aéronautique	élève BTS maintenance aéronautique	élève BTS maintenance aéronautique	élève BTS maintenance aéronautique	élève BTS maintenance aéronautique
Expertise	Novice	Novice	Novice	Novice	Novice
Qualification					
Météo					
Vent (Km/h)					
Heure/Date	02/12/10	02/12/10	02/12/10	02/12/10	02/12/10
Niveau sonore	Bas	Bas	Bas	Bas	Bas
Milieu					
Ecran	PDA	PDA	PC	PC	PC
Niveau de luminosité	Moyen	Moyen	Moyen	Moyen	Moyen
Nom de la tâche	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal	Procédure AMM 32-41-11-000-006-B Removal of the MLG Wheel et sous-tâche AMM 32-41-11-020-075-A Preparation of the Wheel before Removal
Type	non programmée	non programmée	non programmée	non programmée	non programmée
Nature	R/I	R/I	R/I	R/I	R/I
Fréquence	Fréquente	Fréquente	Fréquente	Fréquente	Fréquente
Temps d'exécution (s)	1346	565	858	653	196
Exigence	Exigeante	Exigeante	Exigeante	Exigeante	Exigeante
Saute des étapes	Non	Non	Non	Non	Non
Durée de réalisation par rapport à la durée de référence (=291s)	supérieure	supérieure	supérieure	supérieure	inférieure