



HAL
open science

Conception d'une architecture de services d'intelligence ambiante pour l'optimisation de la qualité de service de transmission de messages en e-santé

Nachoua Guizani

► To cite this version:

Nachoua Guizani. Conception d'une architecture de services d'intelligence ambiante pour l'optimisation de la qualité de service de transmission de messages en e-santé. Intelligence artificielle [cs.AI]. Université de Lyon, 2016. Français. NNT : 2016LYSE1157 . tel-01412597

HAL Id: tel-01412597

<https://theses.hal.science/tel-01412597v1>

Submitted on 8 Dec 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



N°d'ordre NNT : 2016LYSE1157

THESE de DOCTORAT DE L'UNIVERSITE DE LYON
opérée au sein de
l'Université Claude Bernard Lyon 1

Ecole Doctorale ED 512
Ecole Doctorale Informatique et Mathématiques de Lyon
Spécialité Informatique

A soutenir publiquement le 30 septembre 2016, par :
Nachoua GUIZANI

**Conception d'une architecture de services
d'intelligence ambiante pour l'optimisation de la
qualité de service de transmission de messages
en e-santé**

Devant le jury composé de :

Marie-Christine JAULENT
Corine CAUVET
Joël COLLOC
Rémi BASTIDE
Norbert NOURY
Jocelyne FAYN
Jérôme DARMONT
Bruno GIACOMINI

Directrice de Recherche INSERM
Professeur, Université d'Aix-Marseille
Professeur, Université du Havre
Professeur, Université Champollion
Professeur, Université Lyon 1
Ingénieur de Recherche INSERM, HDR
Professeur, Université Lyon 2
Responsable R&D, AXON'Cable SAS

Présidente du jury
Rapporteur
Rapporteur
Examineur
Examineur
Directrice de thèse
Co-Directeur de thèse
Invité

UNIVERSITE CLAUDE BERNARD - LYON 1

Président de l'Université

Président du Conseil Académique

Vice-président du Conseil d'Administration

Vice-président du Conseil Formation et Vie Universitaire

Vice-président de la Commission Recherche

Directeur Général des Services

M. le Professeur Frédéric FLEURY

M. le Professeur Hamda BEN HADID

M. le Professeur Didier REVEL

M. le Professeur Philippe CHEVALIER

M. Fabrice VALLÉE

M. Alain HELLEU

COMPOSANTES SANTE

Faculté de Médecine Lyon Est – Claude Bernard

Faculté de Médecine et de Maïeutique Lyon Sud – Charles
Mérieux

Faculté d'Odontologie

Institut des Sciences Pharmaceutiques et Biologiques

Institut des Sciences et Techniques de la Réadaptation

Département de formation et Centre de Recherche en Biologie
Humaine

Directeur : M. le Professeur J. ETIENNE

Directeur : Mme la Professeure C. BURILLON

Directeur : M. le Professeur D. BOURGEOIS

Directeur : Mme la Professeure C. VINCIGUERRA

Directeur : M. le Professeur Y. MATILLON

Directeur : Mme la Professeure A-M. SCHOTT

COMPOSANTES ET DEPARTEMENTS DE SCIENCES ET TECHNOLOGIE

Faculté des Sciences et Technologies

Département Biologie

Département Chimie Biochimie

Département GEP

Département Informatique

Département Mathématiques

Département Mécanique

Département Physique

UFR Sciences et Techniques des Activités Physiques et Sportives

Observatoire des Sciences de l'Univers de Lyon

Polytech Lyon

Ecole Supérieure de Chimie Physique Electronique

Institut Universitaire de Technologie de Lyon 1

Ecole Supérieure du Professorat et de l'Education

Institut de Science Financière et d'Assurances

Directeur : M. F. DE MARCHI

Directeur : M. le Professeur F. THEVENARD

Directeur : Mme C. FELIX

Directeur : M. Hassan HAMMOURI

Directeur : M. le Professeur S. AKKOUCHE

Directeur : M. le Professeur G. TOMANOV

Directeur : M. le Professeur H. BEN HADID

Directeur : M. le Professeur J-C PLENET

Directeur : M. Y. VANPOULLE

Directeur : M. B. GUIDERDONI

Directeur : M. le Professeur E. PERRIN

Directeur : M. G. PIGNAULT

Directeur : M. le Professeur C. VITON

Directeur : M. le Professeur A. MOUGNIOTTE

Directeur : M. N. LEBOISNE

Remerciements

Ce travail de thèse a pu aboutir grâce à la contribution de différentes personnes faisant partie de la communauté des universitaires et des chercheurs et de ma famille. Je les remercie vivement pour leurs encouragements, leur soutien et leur accompagnement tout au long de ma thèse. Je tiens aussi à remercier vivement tous ceux qui m'ont aidé et encouragé de près ou de loin à réaliser ma thèse.

J'adresse de sincères remerciements à Madame Corine CAUVET, Professeur à l'Université d'Aix-Marseille, et à Monsieur Joël COLLOC, Professeur à l'Université du Havre, de m'avoir fait l'honneur de rapporter ma thèse et pour l'intérêt qu'ils ont porté à mon travail. J'associe à ces remerciements, Madame Marie-Christine JAULENT, Directrice de Recherche INSERM et Directrice de l'UMRS INSERM LIMICS « Laboratoire d'Informatique Médicale et d'Ingénierie des Connaissances en e-Santé », Monsieur Rémi BASTIDE, Professeur à l'Institut National Universitaire Champollion, Monsieur Norbert NOURY, Professeur à l'Université Lyon 1, pour avoir accepté d'examiner mon travail et d'assister à la soutenance de ma thèse.

Ayant réalisé cette thèse conjointement au sein de la SFR Santé Lyon-Est et dans le cadre d'un partenariat industriel avec l'entreprise AXON'CABLE, sous contrat CIFRE, je tiens à remercier Monsieur Joseph PUZO, Président-directeur général d'AXON'CABLE et Monsieur Bruno GIACOMINI, responsable de projets R&D, pour leur accueil chaleureux dans leur entreprise et pour m'avoir permis de mener ces travaux dans de très bonnes conditions.

Je remercie Madame Jocelyne Fayn, ma Directrice de thèse et Directrice de la plateforme eTechSanté de la SFR Santé Lyon-Est, INSERM US7. Merci Jocelyne pour vos précieux conseils, votre aide et votre manière de travailler qui sont aussi magiques que l'ingéniosité de votre encadrement et la sagesse de vos orientations scientifiques. Je vous remercie également pour votre confiance en mes capacités. Ce fut un réel plaisir et un immense honneur d'être votre étudiante.

Cette thèse a été co-dirigée par Monsieur Jérôme Darmont, Professeur à l'Université de Lyon 2 et directeur du laboratoire ERIC « Entrepôts, Représentation et Ingénierie des Connaissances ». Je tiens à le remercier pour son soutien, ses précieux conseils, sa disponibilité malgré son programme chargé. Je lui suis également reconnaissante pour le temps conséquent qu'il m'a accordé, ses qualités pédagogiques et scientifiques, sa franchise et sa sympathie. J'ai beaucoup appris à ses côtés et je lui adresse ma gratitude pour tout cela. Merci aussi à tous les membres du laboratoire ERIC.

Un immense merci à Monsieur Paul RUBEL, ex-Professeur à l'INSA de Lyon pour sa gentillesse, son écoute, son soutien et pour sa disponibilité dont il a su faire preuve tout au long de nos séances de travail. Il m'a enseigné la rigueur et la précision dans la rédaction des

articles scientifiques. Merci Monsieur Rubel pour tout le temps que vous m'avez accordé. Votre expérience et les discussions très enrichissantes que j'ai menées avec vous ont sans doute contribué nettement à l'amélioration de mes recherches.

Je remercie également mes collègues membres du laboratoire et mes amis Joël, Ghazi, Ebrahim, Nadia, Rabeb, Arij,... pour les bons moments partagés. L'ambiance de travail a été excellente et nous avons passé de bons moments lors des pauses café, repas, sorties, etc. Je dois un grand merci à Hossam qui m'a soutenu et n'a cessé de m'encourager.

Je ne saurais terminer ces deux pages sans une grande pensée à mes parents, mes frères, Mama, mes tantes, mes oncles et mes neveux. Un très grand merci à mon frère Haythem qui a toujours été à côté de moi, et sans qui je n'en serais pas là aujourd'hui.

Résumé

La gestion de l'acheminement de messages d'e-santé en environnement ubiquitaire soulève plusieurs défis majeurs liés à la diversité et à la spécificité des cas d'usage et des acteurs, à l'évolutivité des contextes médical, social, logistique, environnemental...

Nous proposons une méthode originale d'orchestration autonome et auto-adaptative de services visant à optimiser le flux des messages et à personnaliser la qualité de transmission, en les adressant aux destinataires les plus appropriés dans les délais requis. Notre solution est une architecture générique dirigée par des modèles du domaine d'information considéré et des données contextuelles, basés sur l'identification des besoins et des contraintes soulevées par notre problématique.

Notre approche consiste en la composition de services de fusion et de gestion dynamique en temps réel d'informations hétérogènes provenant des écosystèmes source, cible et message, pilotés par des méthodes d'intelligence artificielle pour l'aide à la prise de décision de routage. Le but est de garantir une communication fiable, personnalisable et sensible à l'évolution du contexte, quel que soit le scénario et le type de message (alarme, technique, etc.). Notre architecture, applicable à divers domaines, a été consolidée par une modélisation des processus métiers (BPM) explicitant le fonctionnement des services qui la composent.

Le cadrage proposé est basé sur des ontologies et est compatible avec le standard HL7 V3. L'auto-adaptation du processus décisionnel d'acheminement est assurée par un réseau bayésien dynamique et la supervision du statut des messages par une modélisation mathématique utilisant des réseaux de Petri temporels.

Mots-clés : Intelligence ambiante, Systèmes autonomes, Sensibilité au contexte, SOA, BPM, e-santé, HL7, Flux de messages.

Abstract

Routing policy management of eHealth messages in ubiquitous environment leads to address several key issues, such as taking into account the diversity and specificity of the different use cases and actors, as well as the dynamicity of the medical, social, logistic and environmental contexts.

We propose an original, autonomous and adaptive service orchestration methodology aiming at optimizing message flow and personalizing transmission quality by timely sending the messages to the appropriate recipients. Our solution consists in a generic, model-driven architecture where domain information and context models were designed according to user needs and requirements.

Our approach consists in composing, in real time, services for dynamic fusion and management of heterogeneous information from source, target and message ecosystems, driven by artificial intelligence methods for routing decision support. The aim is to ensure reliable, personalized and dynamic context-aware communication, whatever the scenario and the message type (alarm, technical, etc.). Our architecture is applicable to various domains, and has been strengthened by business process modeling (BPM) to make explicit the services operation.

The proposed framework is based on ontologies and is compatible with the HL7 V3 standard. Self-adaptation of the routing decision process is performed by means of a dynamic Bayesian network and the messages status supervision is based on timed Petri nets.

Keywords: Ambient intelligence, Autonomic computing, Context-awareness, SOA, BPM, eHealth, HL7, Workflow management.

Table des matières

Résumé	7
Abstract	8
Table des matières	9
Liste des Figures	15
Liste des Tableaux	17
Liste des Algorithmes	19
Introduction Générale.....	21
Partie I Problématique d’acheminement des messages en e-santé	29
Chapitre 1 L’e-santé et la télémédecine citoyenne.....	33
1.1 Introduction	33
1.2 Le concept de l’e-santé.....	33
1.3 Les défis du système de santé.....	34
1.4 L’e-santé et ses enjeux	34
1.4.1 Enjeux socio-économiques.....	35
1.4.2 Enjeux médicaux et sociaux.....	35
1.4.3 Enjeux scientifiques	36
1.5 L’exemple du projet européen EPI-MEDICS	37
1.6 Le Dossier Médical Personnel (DMP) et interopérabilité des systèmes d’information médicaux	40
1.6.1 Le DMP.....	40
1.6.2 Standards internationaux et protocoles de communication en santé	40
1.6.3 Nécessité d’une standardisation	41
1.6.3.1 Interfaçage des systèmes	41
1.6.3.2 Interopérabilité des systèmes d’information en Santé.....	42
1.6.4 Le Standard HL7.....	43
1.6.4.1 Définition.....	43
1.6.4.2 Les versions HL7.....	43
1.6.4.3 Le RIM	44
1.7 Conclusion.....	49
Chapitre 2 La gestion de l’acheminement des messages en e-santé. Défis et besoins	51
2.1 Introduction	51
2.2 Défis des systèmes de gestion des acheminements des messages en e-santé.....	52

2.2.1	Exemples de scénarios.....	52
2.2.1.1	Scénario d’alarme moyenne du projet EPI-MEDICS.....	52
2.2.1.2	Scénario de télé-consultation du projet national Télé-infirmière	53
2.2.2	Diversité des scénarios en e-santé	54
2.2.3	Diversité des acteurs en e-santé.....	54
2.2.4	Contraintes en environnement ubiquitaire.....	56
2.2.4.1	C1 : Evolutivité du contexte	56
2.2.4.2	C2 : Mobilité des utilisateurs.....	57
2.2.4.3	C3 : Hétérogénéité des environnements	57
2.2.4.4	C4 : Temps de réponse	57
2.3	Besoins requis en gestion des acheminements en e-santé	58
2.3.1	B1: Personnalisation.....	58
2.3.2	B2: Sensibilité au contexte	58
2.3.3	B3: Auto-Adaptation	59
2.3.4	B4: Interopérabilité.....	59
2.3.5	B5: Supervision	59
2.3.6	B6: Autonomie	60
2.3.7	B7: Sécurité et protection de la vie privée.....	60
2.3.8	B8: Intelligence	60
2.3.9	B9: Qualité de Service.....	61
2.3.9.1	Définition.....	61
2.3.9.2	Qualité de service dans la gestion d’acheminement des messages.....	61
2.4	Travaux connexes. Avantages et Limites.....	62
2.4.1	Travaux de recherche connexes.....	63
2.4.1.1	Approches tous domaines.....	63
2.4.1.2	Approches e-santé	64
2.4.2	Projets de développement connexes.....	67
2.4.2.1	Projet MSSanté.....	67
2.4.2.2	Projet ACM	68
2.4.2.3	Projet régional de télé-expertise : Expert-Assist	69
2.4.3	Synthèse des travaux connexes	71
2.5	Conclusion.....	71
Partie II Modélisation des systèmes dynamiques, intelligents, autonomes et sensibles au contexte ...		73
Chapitre 3 Les systèmes autonomes et sensibles au contexte		77
3.1	Introduction	77
3.2	Les systèmes autonomes	77

3.2.1	Définition des systèmes autonomes.....	77
3.2.2	Caractéristiques des systèmes autonomes	78
3.2.3	Architecture des systèmes autonomes	78
3.3	Les systèmes multi-agents.....	79
3.3.1	Quelques définitions.....	79
3.3.2	Intelligence artificielle distribuée.....	81
3.3.3	Applications des systèmes multi-agents.....	81
3.4	Les systèmes sensibles au contexte	82
3.4.1	Le concept “contexte”	82
3.4.2	Définition des systèmes sensibles au contexte	82
3.4.3	Adaptation des systèmes sensibles au contexte	83
3.4.4	Exemples d’intericiels sensibles au contexte	83
3.4.4.1	Exemple de CAMidO.....	83
3.4.4.2	Exemple de CA3M.....	84
3.4.4.3	Exemple de Context Toolkit.....	85
3.4.5	Modélisation du contexte	86
3.4.5.1	Approche orientée ontologie	86
3.4.5.2	Approche orientée Ingénierie Dirigée par les Modèles (IDM).....	87
3.4.5.3	Discussion	90
3.5	Apport des systèmes autonomes à la sensibilité au contexte et aux systèmes multi-agents..	91
3.6	Conclusion.....	91
Chapitre 4	Méthodes d’aide à la décision et de modélisation des systèmes dynamiques	93
4.1	Introduction	93
4.2	Méthodes d’intelligence artificielle pour l’aide à la décision.....	93
4.2.1	Les arbres de décision	94
4.2.1.1	Définition.....	94
4.2.1.2	Exemple introductif.....	94
4.2.1.3	Algorithmes d’apprentissage d’arbres de décision	96
4.2.2	Les systèmes experts	98
4.2.2.1	Définition.....	98
4.2.2.2	Composition d’un système expert	98
4.2.2.3	Types de moteur d’inférence	99
4.2.3	Les colonies de fourmis.....	102
4.2.3.1	Définition.....	102
4.2.3.2	Expérience du double pont binaire	102

4.2.3.3	Algorithme de colonies de fourmis appliqué au problème du voyageur de commerce	103
4.2.3.4	Intensification versus diversification.....	105
4.2.4	Les réseaux bayésiens	105
4.2.4.1	Définition.....	105
4.2.4.2	Notation graphique	106
4.2.4.3	Distribution de probabilité conjointe.....	107
4.2.4.4	Indépendance conditionnelle	107
4.2.4.5	Inférence dans un réseau bayésien : l'arbre de jonction.....	108
4.2.5	Les réseaux bayésiens dynamiques	110
4.2.5.1	Définition.....	110
4.2.5.2	Hypothèse Markovienne.....	111
4.2.5.3	Inférence dans un réseau bayésien dynamique.....	112
4.2.5.4	Exemples d'application des réseaux bayésiens dynamiques	114
4.3	Modélisation des systèmes dynamiques.....	115
4.3.1	Les réseaux de Petri.....	115
4.3.1.1	Sémantique des places, des transitions et des jetons	116
4.3.1.2	Représentation graphique et notation matricielle	116
4.3.1.3	Le marquage	116
4.3.1.4	Franchissement d'une transition.....	116
4.3.2	Dimension temporelle dans les réseaux de Petri	117
4.3.2.1	Réseaux de Petri temporisés	117
4.3.2.2	Réseaux de Petri temporels	118
4.4	Conclusion.....	119
Partie III Gestion de l'acheminement des messages autonome, intelligente et sensible au contexte.		121
Chapitre 5	Modèles et architecture pour la gestion autonome, intelligente et auto-adaptative de l'acheminement de messages	125
5.1	Introduction	125
5.2	Modèle de représentation de politiques d'acheminement de messages avec prise en compte du contexte multi-dimensionnel	125
5.3	Architecture orientée services pour la gestion autonome et intelligente de l'acheminement des messages	127
5.3.1	La publication du message : « <i>Message Reporter</i> »	127
5.3.2	La dissémination du message : « <i>Message Disseminator</i> ».....	128
5.3.3	La représentation des connaissances : Ecosystèmes source, cible et message	128
5.3.4	La gestion de la politique d'acheminement : <i>Routing Policy Manager</i>	128
5.3.4.1	Services d'analyse	129

5.3.4.2	Services auto-adaptatifs.....	129
5.3.5	La supervision des écosystèmes : « <i>Ecosystem supervisor</i> »	131
5.3.6	La supervision du statut des messages : « <i>Message status supervisor</i> »	131
5.4	Modélisation ontologique des écosystèmes sources, cibles et messages	131
5.5	Prise en compte des besoins requis pour l'acheminement des messages	133
5.6	Conclusion.....	135
Chapitre 6	Modélisation des processus métiers pour la gestion de l'acheminement des messages	137
6.1	Introduction.....	137
6.2	Modélisation du processus métier « Supervision des écosystèmes ».....	137
6.3	Modélisation du processus métier « Supervision du statut des messages »	138
6.4	Modélisation du processus métier « Gestion de l'acheminement des messages »	140
6.5	Modélisation du processus métier « Inférence des paramètres d'acheminement ».....	142
6.6	Modélisation du processus métier « Détermination de la destination»	142
6.7	Conclusion.....	143
Partie IV	Vers l'implémentation d'un système de gestion d'acheminement des messages. Application à l'e-santé	145
Chapitre 7	Modèles et flux d'informations pour un pilotage de l'acheminement des messages en e-santé compatible HL7 version 3	149
7.1	Introduction.....	149
7.2	Modèles d'informations appliqués au domaine de l'e-santé	149
7.2.1	Modélisation des écosystèmes source, cible et messages.....	150
7.2.1.1	Modélisation en couches des écosystèmes	151
7.2.1.2	Modélisation UML des écosystèmes.....	154
7.2.1.3	Modélisation ontologique de quelques exemples de contextes et d'écosystèmes	157
7.2.2	Raisonnement et inférence ontologique du contexte.....	163
7.3	Présentation des principaux flux d'information de notre architecture appliquée à l'e-santé.....	164
Publication du message		164
7.3.1	Acquittement de publication du message	167
7.3.2	Transmission du message vers le destinataire	167
7.3.3	Notifications du statut du message	167
7.3.4	Notifications des paramètres de gestion de l'acheminement du message.....	168
7.3.5	Notification des problèmes d'acheminement et de traitement du message	168
7.3.6	Notification de changements de données contextuelles	169
7.4	Exemple de diagrammes de séquence dans le cadre d'un scénario d'acheminement des messages pour la surveillance des personnes âgées	169

7.5	Conclusion.....	171
Chapitre 8	Réseaux bayésiens dynamiques et réseaux de Petri temporels pour la gestion de l'acheminement des messages en e-santé.....	173
8.1	Introduction.....	173
8.2	Réseaux bayésiens dynamiques pour l'inférence des paramètres d'acheminement.....	173
8.2.1	Pourquoi les réseaux bayésiens dynamiques ?.....	173
8.2.2	Construction d'un réseau bayésien dynamique pour l'inférence du paramètre d'acheminement « type de destinataire ».....	176
8.2.3	Construction d'un réseau bayésien dynamique pour l'inférence du paramètre d'acheminement « délai requis pour la lecture du message ».....	177
8.2.3.1	Choix des variables du RBD basé sur les attributs du HL7 RIM.....	178
8.2.3.2	Scénario de simulation du RBD dans le domaine de cardiologie.....	178
8.3	Les réseaux de Petri temporels pour la supervision du statut des messages.....	179
8.3.1	Pourquoi les réseaux de Petri temporels ?.....	179
8.3.2	Modélisation du statut des messages.....	180
8.3.3	Modélisation du Réseau de Petri temporel.....	181
8.4	Conclusion et discussion.....	182
Chapitre 9	Application de notre méthodologie à l'e-cardiologie et illustration de son fonctionnement. Discussion.....	183
9.1	Introduction.....	183
9.2	Instanciation de notre méthode de gestion d'acheminement d'un message d'alarme moyenne dans un contexte de « self-care » en cardiologie.....	183
9.2.1	Scénario d'alarme moyenne.....	184
9.2.2	Diagramme de collaboration appliqués à un scénario d'acheminement d'un message d'alarme dans un contexte de « <i>self-care</i> » en cardiologie.....	188
9.3	Discussions et perspectives.....	192
9.4	Conclusion.....	196
Conclusion Générale	197
Bibliographie	205

Liste des Figures

Figure 1. Limites des interfaçages de systèmes.....	41
Figure 2. Avantages d'un protocole commun pour garantir l'interopérabilité entre différents systèmes	42
Figure 3. Principe d'un document CDA d'HL7	44
Figure 4. Valeurs des attributs des classes cœurs du RIM d'HL7.....	45
Figure 5. Exemple d'instanciation des classes « <i>Entity</i> », « <i>Role</i> », « <i>Participation</i> » et « <i>Act</i> » du RIM	46
Figure 6. Représentation UML du cas d'utilisation « administration du patient ».....	46
Figure 7. Eléments du modèle d'interactions d'HL7	47
Figure 8. Modèle d'interaction pour le cas d'utilisation : « ajout d'un patient »	47
Figure 9. Formation des messages HL7 version 3	48
Figure 10. Instanciation XML d'un message HL7.....	49
Figure 11. Scénario d'une alarme moyenne dans le cadre du projet EPI-MEDICS.....	53
Figure 12. Architecture du système de télé-assistance médicale dans le cadre du projet Télé-infirmière.....	53
Figure 13. Diversité des scénarios en e-santé.....	54
Figure 14. Modèle des acteurs en e-santé.....	55
Figure 15. Le diagramme des acteurs du profil ACM.....	68
Figure 16. Architecture logicielle du système Expert-Assist	69
Figure 17. Interface pour le paramétrage des règles d'acheminement	70
Figure 18. Architecture de systèmes autonomes	79
Figure 19. Architecture CAMidO.....	84
Figure 20. Architecture de CA3M.....	85
Figure 21. Architecture Toolkit.....	85
Figure 22. Le méta-modèle CAMidO	86
Figure 23. Pyramide de modélisation de l'OMG	87
Figure 24. Méta-modèle CA3M de la sensibilité au contexte	88
Figure 25. Le méta-modèle ContextUML.....	89
Figure 26. Arbre de décision sur le fichier de données « <i>Weather</i> ».....	95
Figure 27. Organisation d'un système expert.....	98
Figure 28. Expérience du double pont binaire pour la sélection des branches les plus courtes par une colonie de fourmis	103
Figure 29. Exemple de réseau bayésien	106
Figure 30. Exemple de tables de probabilités d'un Réseau Bayésien	107
Figure 31. Trois types de connexion dans un réseau bayésien	108
Figure 32. Graphe d'un modèle de Markov observable	111
Figure 33. Exemple de modèle de Markov caché (HMM).....	112
Figure 34. Modèle d'évolution du système Diatelic TM v3.	115
Figure 35. Exemple de marquage dans un RdP.....	116
Figure 36. Principe de franchissement d'une transition dans un RdP	117
Figure 37. Schéma de principe de notre démarche méthodologique.....	124

Figure 38. Modèle de représentation de politiques d'acheminement de messages avec prise en compte de contextes	127
Figure 39. Architecture pour la gestion d'acheminement des messages	128
Figure 40. Modèle ontologique du contexte	132
Figure 41. Représentation des besoins d'acheminement des messages	133
Figure 42. Le processus de supervision des écosystèmes	138
Figure 43. Le processus supervision du statut du message	139
Figure 44. Le processus de gestion d'acheminement des messages.....	141
Figure 45. Le sous-processus d'inférence des paramètres d'acheminement	142
Figure 46. Le sous-processus de détermination de la destination en e-santé	142
Figure 47. Synoptique de représentation de notre architecture de gestion d'acheminement des messages.....	150
Figure 48. Modélisation en couches des écosystèmes source, cible et message	152
Figure 49. Diagramme de classes modélisant l'écosystème source	155
Figure 50. Diagramme de classes modélisant l'écosystème cible	156
Figure 51. Diagramme de classes modélisant une partie de l'écosystème message	156
Figure 52. Classification de l'état des signes cliniques en cardiologie	157
Figure 53. Classification de l'état des paramètres biomédicaux en cardiologie.....	158
Figure 54. Classification des facteurs de risques en cardiologie	159
Figure 55. Classification de A) Etat du trafic routier, B) Type de l'environnement, C) Etat du climat, D) Etat d'accès	160
Figure 56. Classification de l'état d'accompagnement	160
Figure 57. Classement des distances	160
Figure 58. Ontologie de l'écosystème source.....	162
Figure 59. Ontologie de l'écosystème message.....	163
Figure 60. Représentation des flux d'interaction et des exceptions	164
Figure 61. Diagramme de séquence d'acheminement d'une requête de prise en charge en cas de son acceptation par le médecin	170
Figure 62. Diagramme de séquence d'acheminement d'une requête de prise en charge en cas d'indisponibilité du professionnel de santé et de changements du contexte.	171
Figure 63 . Réseau bayésien dynamique pour l'inférence du type de staff	177
Figure 64. Simulation du RBD pour l'inférence du délai requis pour la lecture d'un message en e-santé dans le domaine de la cardiologie	179
Figure 65. Diagramme UML d'états-transitions pour la modélisation des états du message	181
Figure 66. Réseau de Petri temporel pour la modélisation du statut du message.....	182
Figure 67. Diagramme de collaboration du processus de gestion d'acheminement d'une alarme moyenne en cardiologie (alarme issue d'un CARDIAN®).....	185
Figure 68. Diagramme de collaboration du sous-processus « <i>Search for relevant healthcare professional</i> » du RPM appliqué à un scénario en cardiologie	186

Liste des Tableaux

Tableau 1. Contraintes et besoins d'un système de gestion d'acheminement des messages en e-santé.....	62
Tableau 2. Tableau comparatif des recherches étudiées.....	71
Tableau 3. Données « <i>Weather</i> ».....	95
Tableau 4. Nature des besoins satisfaits par les services de notre architecture d'acheminement des messages.....	134
Tableau 5. Valeurs possibles de l'élément « <i>Acknowledgement detail code</i> » du message d'acquittement HL7.....	167
Tableau 6. Quelques valeurs possibles de l'élément « <i>Acknowledgement type</i> » du message d'acquittement HL7.....	167
Tableau 7. Enumération des valeurs du statut du message.....	168
Tableau 8. Structure de la transaction « paramètres d'acheminement du message ».....	168
Tableau 9. Valeurs possibles de la transaction « problèmes d'acheminement et de traitement du message ».....	169
Tableau 10. Etude comparative des méthodes d'aide à la décision pour l'inférence des paramètres d'acheminement.....	174
Tableau 11. Liste préconfigurée des destinataires du patient A et leurs critères de sélection de type « <i>Background attributes</i> ».....	187
Tableau 12. Liste préconfigurée des destinataires des messages provenant du CARDIAN® du patient A et leurs critères de sélection de type « <i>Real-time attributes</i> ».....	187

Liste des Algorithmes

Algorithme 1. Algorithme d'apprentissage générique	96
Algorithme 2. Chaînage avant.....	100
Algorithme 3. Chaînage arrière	101
Algorithme 4. Chaînage mixte	102
Algorithme 5. Algorithme de fourmis de base : le « Ant System »	105
Algorithme 6. Algorithme Forward.....	113
Algorithme 7. Algorithme Backward.....	114

Introduction Générale

L'e-santé, la p-santé (e-santé personnalisée) comme la m-santé (m signifiant mobile) font déjà partie intégrante du quotidien de nombre de personnes à risque susceptibles d'être confrontées à un incident de santé pour elles-mêmes ou pour leurs proches. L'objectif est d'améliorer leur qualité de vie en accélérant, grâce à l'utilisation des technologies de l'information et de la communication, leur prise en charge médicale par les centres et professionnels de santé appropriés, à tout instant et en tout lieu, procurant ainsi un accès aux soins prompt et efficace pour tous, tout en diminuant les coûts. Le concept d'e-santé génère de fait une quantité très importante de transmissions de messages véhiculant aussi bien une demande de conseil diététique d'un patient à son médecin traitant ou d'assistance technique pour l'utilisation d'instrumentations médicales par tout type d'acteur (patient ou professionnel de santé), que le convoiement intermittent ou programmé de résultats médicaux inter-acteurs (téléassistance ou hospitalisation à domicile, partage d'informations, etc.), de demandes de télé-expertise à un confrère, jusqu'au déclenchement d'une alarme vitale. Une nouvelle génération de dispositifs personnels de santé dotés d'intelligence embarquée et de moyens de communication sans fil peuvent aussi transmettre, automatiquement ou à la demande, à un serveur dédié distant des alertes de diverses natures, à traiter en urgence ou en différé. De tels systèmes sont déjà sur le marché, et ce sera l'un des premiers secteurs industriels dans les prochaines années. Ils sont destinés à l'usage de tout citoyen, de ses aidants familiaux, des personnels paramédicaux, des équipes de secours, et de tout professionnel de santé, en environnement domotique dans le cas de maladies chroniques (insuffisants cardiaques, respiratoires, rénaux, personnes âgées, ...) et en environnement ubiquitaire (en zone rurale, en montagne, ...) pour des adultes mobiles, en activité, etc. Tous ces messages provenant de et s'adressant à divers types d'acteurs peuvent être urgents ou non, avec des délais requis de traitement très variés.

L'un des challenges sociétaux relevant du domaine des Sciences et Technologies de l'Information et de la Communication est à présent d'assurer une qualité de service intelligente, optimale et pertinente de gestion du flux de communication des messages d'alertes et de transmission d'informations, eu égard à chacun des divers scénarios d'e-santé. Tel est le défi lancé par notre sujet de thèse et par nos travaux de recherche qui visent à contribuer à répondre à cette problématique en proposant une méthodologie de conception d'intelligence ambiante de supervision de la gestion de l'acheminement de messages entre un patient, ses aidants ou un dispositif de surveillance et un professionnel de santé, ou entre professionnels de santé. Cette supervision n'a pas pour vocation de réguler automatiquement tous les flux de messages en e-santé sans possibilité d'interaction utilisateur, ni de se substituer aux technologies existantes (téléphone, e-mail), mais d'apporter un nouveau moyen, complémentaire, pour pallier autant que faire se peut les manquements fortuits de prise en charge des patients dans des délais compatibles avec leur état. En effet, une carence de soins médicaux peut être due à diverses raisons pouvant entraîner des issues fatales : indisponibilité du destinataire du message alors que le message aurait pu, si nécessaire, être automatiquement redirigé vers un confrère ou vers le SAMU, incapacité du patient à alerter les services d'urgence suite à un malaise, prise de conscience insuffisante de la gravité des symptômes ou des paramètres médicaux par un patient ou ses aidants, etc.

Dans la littérature, les travaux de recherche qui traitent de la problématique d'acheminement de messages sont nombreux mais, en l'état de nos connaissances, aucun d'entre eux ne propose une solution qui à la fois prend en compte l'aspect multidimensionnel des contextes source, cible et message pour déterminer automatiquement la destination la plus pertinente, supervise les changements du contexte, contrôle les états du message et adapte en conséquence les décisions de routage à divers cas d'utilisation.

Objectifs

Notre approche consiste à proposer une méthode d'orchestration de services intelligents visant à optimiser la qualité de transmission des flux de messages tout en tenant compte des changements de contexte qui peuvent survenir en temps réel. Les circonstances qui peuvent interférer dans le processus d'acheminement sont, par exemple, l'indisponibilité des professionnels de santé et/ou de ressources matérielles, un changement de localisation géographique du patient et/ou des professionnels de santé, etc., et des exceptions événementielles imprévues comme le dépassement du délai requis pour répondre aux demandes de prise en charge. Notre objectif est de rendre le processus décisionnel d'acheminement : (1) Personnalisable, afin de pouvoir délivrer le message au « bon » destinataire et dans le « bon » délai requis pour son traitement ; (2) Sensible au contexte, en le dotant de capacités d'observer son environnement extérieur et d'adapter son comportement décisionnel en fonction des situations. Pour satisfaire ces deux propriétés, le processus décisionnel d'acheminement doit pouvoir être : (3) autonome, c'est-à-dire capable de s'auto-adapter et de prendre ses propres décisions sans intervention humaine si nécessaire, et donc (4) Intelligent, grâce à des techniques d'Intelligence Artificielle (IA) embarquées, pour superviser et piloter le convoiement des messages à bon escient.

L'optimisation de la qualité de service de transmission et d'acheminement de messages d'alertes nécessitera la prise en compte multidimensionnelle du contexte des acteurs sources et cibles de ces messages. Les dimensions à considérer sont spatiotemporelles, afin de prendre en compte la dynamique des dimensions spatiales que représentent, par exemple en e-santé, le contexte médical, le contexte géographique et le contexte sociétal des acteurs cibles et sources. Chacun de ces contextes devra être décliné lui-même en divers types de concepts. En effet, le contexte médical source du patient résulte à la fois de la perception de ses besoins à l'instant t en fonction de son état physiologique et de ses signes cliniques, de leur gravité potentielle et de leur niveau d'urgence vitale en fonction de l'état des connaissances médicales en général et de sa propre histoire clinique. Le contexte médical cible des professionnels ou institutions de santé se caractérise par les compétences matérielles (équipements spécifiques d'exploration en imagerie, de nature interventionnelle pour la chirurgie, la thérapie, etc.) et humaines (en termes de spécialité médicale, etc.) disponibles à l'instant t . Le contexte sociétal du patient dépend notamment de l'entourage dont il dispose : y-a-t-il un médecin, un(e) infirmier(ère), de la famille, des amis, des aidants à son chevet ou à proximité ? Quelles sont ses conditions de vie ? Côté cible médicale, le temps de réponse dépendra de la qualité logistique disponible et de sa gestion (existence d'une permanence, etc.). Le contexte géographique se décline à la fois dans le temps et dans l'espace (durée de trajet, densité du trafic routier, éloignement et difficultés d'accès, moyens de transport

disponibles, etc.). L'optimisation de la qualité de service pour l'acheminement des messages d'alertes et leur prise en charge devra donc être réalisée en faisant une projection agrégée/combinée de l'ensemble de ces éléments contextuels et de leur degré d'importance relative.

Verrous scientifiques à lever

Notre modèle d'architecture vise à être le plus générique possible pour répondre aux divers scénarios d'usage, notamment en e-santé. Un message en e-santé peut être de plusieurs types : (1) informatif : envoyé suite par exemple à un protocole de soin préétabli requérant un envoi régulier des paramètres biomédicaux surtout en cas de maladies chroniques, (2) alarme, (3) technique, etc. Le message peut être généré automatiquement par un dispositif médical intelligent ou bien déclenché suite à une intervention humaine (patient, aide-soignant). Cette large diversité de scénarios doit être prise en compte lors de la mise en place d'une stratégie d'acheminement. L'un des verrous majeur à lever est de concevoir un système décisionnel d'acheminement des messages qui prend en compte la dynamique et le changement continu du contexte. La localisation de la source et de la cible, la disponibilité des ressources matérielles et humaines, ..., etc. sont des données contextuelles qui reflètent la dynamique de l'environnement externe auxquels une politique d'acheminement est très sensible. La question qui se pose à cet égard est la suivante : Comment adapter le processus décisionnel d'acheminement des messages en fonction du contexte social, médical, et environnemental de la source, en fonction de la connaissance et de l'expérience de la cible et en fonction d'autres événements imprévus comme le dépassement du délai requis pour la prise en compte du message ? Pour assurer une telle adaptation, le défi est triple : modéliser l'espace des informations contextuelles, identifier les contraintes ou bien les situations d'adaptation auxquelles la décision d'acheminement est fortement sensible, et enfin adapter le comportement de la politique d'acheminement en fonction du changement de contexte. Le challenge des années 90 était en effet d'accéder à l'information qu'il faut, où il faut et quand il faut ! Le challenge actuel est non plus seulement de donner les moyens d'accéder à l'information, mais de « fournir automatiquement par anticipation les bonnes informations en tout lieu et à tout moment », dont l'intérêt est de la plus haute importance lorsqu'il s'agit de la prise en charge de la santé des citoyens.

Approches développées

Notre première approche consiste à précisément identifier et modéliser les différents défis et besoins qui découlent de la problématique de la gestion de l'acheminement en e-santé en environnement ubiquitaire. Pour illustrer ces défis, nous nous sommes basés notamment sur deux exemples de scénarios réels issus de deux projets de notre équipe : Le projet européen EPI-MEDICS [1] et le projet national Télé-infirmière [2]. Nous nous intéressons dans ce travail de thèse aux apports de la conception des systèmes autonomes. Nous nous appuyons, en particulier sur les concepts de sensibilité au contexte et d'auto-adaptation, deux propriétés fondamentales qui sont à la base des systèmes autonomes. Pour rendre le processus décisionnel d'acheminement intelligent et prendre en compte la dynamique du contexte dans le temps et dans l'espace, nous explorons les potentialités de quelques-unes des méthodes

d'intelligence artificielle pour l'aide à la décision (arbres de décision, systèmes experts, colonies de fourmis, réseaux bayésiens) et de modélisation dynamique stochastique (réseaux bayésiens dynamiques) et mathématique (Réseau de Petri temporel). Puis nous proposons une approche entièrement originale, qui consiste en la fusion de modèles d'information et de connaissances contextuelles et de modèles d'intelligence basés eux-mêmes sur la fusion de méthodes d'intelligence artificielle ou d'informatique décisionnelle invoquant les divers types de paramètres impliqués à l'instant t . Nous proposons ensuite une architecture orientée services (SOA) et dirigée par des modèles (MDA), dont le but est de servir de support à la mise en œuvre de stratégies d'acheminement personnalisable, intelligente, sensible au contexte et auto-adaptative. La ligne conductrice de ces travaux a été d'orchestrer des systèmes d'information de trois écosystèmes, que nous avons appelés source, cible et message, et une architecture de services intelligents. Les services à déployer interagissent entre eux tout en exécutant un ensemble de processus métiers. Pour expliciter la collaboration entre les différents processus impliqués dans la problématique de pilotage des messages ainsi que les différentes transactions métiers, nous avons adopté une approche BPM (*Business Process Management*) [3]. Nous modélisons ainsi des processus métiers intégrant la gestion des exceptions au cours de la transmission des messages. Une autre approche que nous appréhendons dans ce mémoire est l'adaptation de notre architecture aux spécificités du domaine de l'e-santé et la proposition d'un synoptique de notre architecture intégrant des éléments d'information qui la rendent compatible aux standards internationaux les plus usités pour l'échange informatisé de données cliniques tels que *Health Level Seven International* (HL7) [4]. Nous proposons trois types de modélisation des écosystèmes source, cible et message auxquelles une politique d'acheminement peut être sensible : modélisation en couches, UML et ontologique. La modélisation en couches nous permet d'avoir une vue globale sur le contenu des écosystèmes en fonction duquel une politique d'acheminement peut adapter son comportement. La modélisation UML nous donne une vision plus détaillée sur les relations entre les différents objets du contexte. La modélisation ontologique nous permet de bénéficier de l'aspect sémantique et des techniques d'interprétation qu'offre une représentation ontologique de nos trois écosystèmes.

Nous abordons les aspects implémentation en présentant les méthodes que nous préconisons pour implémenter les services de l'architecture présentée, tout en justifiant nos choix méthodologiques.

Afin de concrétiser nos solutions, nous proposons un diagramme de collaboration BPMN qui illustre le déroulement de l'acheminement d'un exemple de message d'alarme dans le cadre du projet EPI-MEDICS. Cette illustration permet de suivre l'évolution du message dans notre système depuis sa génération par le CARDIAN® jusqu'à son traitement, de montrer comment les modules de notre architecture collaborent ensemble pour gérer l'acheminement du message, et quels sont les flux de messages échangés entre eux.

Enfin, avant de conclure notre mémoire, nous présentons une large discussion portant à la fois sur les solutions technologiques de déploiement des services de notre architecture, sur les

aspects à prendre en compte lors de l'implémentation, et sur les aspects éthiques et légaux inhérents à la mise en routine de notre système.

Structuration du mémoire

Notre mémoire est subdivisé en quatre parties.

La **première partie** de ce mémoire présente le contexte de nos travaux de thèse, ainsi que les différents aspects et axes de recherche qui s'apparentent à notre problématique. Elle se compose de deux chapitres qui sont les suivants.

- Le **chapitre 1** de ce mémoire concerne le contexte de notre travail : l'e-santé et la télémédecine citoyenne, leurs enjeux sur les plans socio-économique, médical et social, les standards internationaux d'échange de données cliniques, financière et administrative (HL7, DICOM, SCP-ECG, ...) et les protocoles de communication en e-santé.
- Le **chapitre 2** est consacré à la présentation de notre problématique et des différents défis et contraintes que nous avons identifiés dans le domaine de la gestion d'acheminement des messages en e-santé. De chaque contrainte, nous avons déduit les besoins que doivent remplir un système de gestion d'acheminement de messages. Nous discutons les avantages et les inconvénients des travaux de recherche et des projets de développement connexes à notre problématique de recherche. Notre revue de la littérature porte sur des approches tous domaines et des approches spécifiques au domaine de l'e-santé. Dans les approches e-santé, nous présentons un état de l'art sur les différents types d'acheminement : acheminement des messages en cas d'urgence, acheminement sensible au contexte, acheminement intelligent des messages.

La **deuxième partie** présente l'état de l'art sur la modélisation des systèmes dynamiques, intelligents, autonomes et sensibles au contexte.

- Le **chapitre 3** présente une synthèse non exhaustive des travaux de recherche relatifs à trois aspects afférents à notre problématique : les systèmes autonomes, les systèmes multi-agents et la sensibilité au contexte. Nous étudions les systèmes sensibles au contexte selon deux axes : un premier axe concerne la gestion du contexte et le développement des intergiciels sensibles au contexte ; un second axe étudie la modélisation du contexte. Dans les travaux de recherche sur les intergiciels, nous distinguons deux approches : l'approche ingénierie dirigée par les modèles (IDM) et l'approche ontologie.
- Le **chapitre 4** présente un état de l'art sur quelques méthodes d'aide à la décision et de modélisation des systèmes dynamiques afférentes à notre problématique.

La **troisième partie** présente l'approche générique que nous proposons pour la gestion d'acheminement des messages, indépendante du domaine d'application.

- Le **chapitre 5** est consacré à la présentation de notre modèle de représentation de politiques d’acheminement de messages avec prise en compte du contexte multidimensionnel, et de notre architecture générique appliquée à tous domaines. Nous explicitons comment notre architecture prend en compte l’ensemble des besoins identifiés dans le chapitre 2.
- Le **chapitre 6** est dédié à la modélisation des processus métiers BPM de composition des services de notre architecture

La **quatrième partie** présente l’approche spécifique et applicative au domaine de l’e-santé que nous proposons pour l’implémentation d’un système de gestion d’acheminement des messages. Elle est subdivisée en 2 chapitres.

- Le **chapitre 7** présente une vue applicative de notre architecture au domaine de l’e-santé et les extensions réalisées pour la rendre compatible HL7 RIM version 3. Nous présentons également les principaux flux d’information de notre architecture appliquée à l’e-santé.
- Le **chapitre 8** présente les méthodes que nous avons choisies pour implémenter quelques services de notre architecture : des réseaux bayésiens dynamiques pour l’inférence des paramètres d’acheminement et des réseaux de Petri temporels pour la détection des impasses lors du routage des messages. Nous présentons et commentons également l’étude comparative que nous avons effectuée pour justifier nos choix méthodologiques.
- Le **chapitre 9** est consacré à la concrétisation de notre solution à travers un scénario réel d’acheminement d’un message d’alarme moyenne dans le domaine de la cardiologie. Les difficultés et les perspectives de mise en œuvre de notre système sont ensuite exposées, et discutées dans ce même chapitre.

Nous terminons ce manuscrit par une synthèse des approches mises en œuvre, puis nous proposons quelques perspectives et de nouvelles pistes de recherche.

Partie I
Problématique d'acheminement des
messages en e-santé

Nous présentons dans cette première partie notre problématique de recherche qui concerne l'acheminement des messages en e-santé. Cette partie est composée de deux chapitres. Dans le premier chapitre, nous présentons le domaine d'application de notre recherche, l'e-santé. Une revue de la littérature relative aux enjeux de l'e-santé, aux différents projets européens et aux standards internationaux liés à l'échange informatisé de données cliniques est fournie. Le deuxième chapitre est consacré à la présentation des différents défis et besoins requis en gestion de routage des messages en e-santé. Dans ce chapitre, nous adoptons une approche *bottom-up*. Nous commençons par présenter et analyser les différents scénarios d'usage extraits de la revue de la littérature en e-santé (prise en charge médicale, télé-expertise, etc.) et qui requièrent un échange de messages, pour déduire les différents problèmes et besoins relatifs à la problématique d'acheminement. Nous étudions les avantages et les inconvénients des travaux connexes sous une vision générique, c'est-à-dire tous domaines, puis ceux qui sont plus particulièrement appliqués au domaine de l'e-santé.

Chapitre 1

L'e-santé et la télémédecine citoyenne

1.1 Introduction

L'e-santé se définit par l'Organisation Mondiale de la Santé (OMS) comme « les services du numérique au service du bien-être de la personne » [5]. Elle est également définie comme « l'utilisation des outils de production, de transmission, de gestion et de partage d'informations numérisées au bénéfice des pratiques tant médicales que médico-sociales » [6]. L'objectif affiché par l'e-santé est de fournir une meilleure prise en charge des patients et garantir un meilleur accès aux soins basés sur les technologies de l'information et de la communication (TIC). Le soin à distance permet de contrecarrer de grandes distances, de compenser la décroissance continue de la démographie médicale et de réduire les dépenses hospitalières. L'e-santé ouvre ainsi de nouvelles perspectives et constitue un atout majeur pour promouvoir le bien-être des citoyens. Elle favorise en outre l'autonomie du patient, qui devient acteur responsable dans la prise en charge de sa propre santé. Les données médicales échangées doivent se soumettre à des standards internationaux pour garantir l'interopérabilité à la fois sémantique et syntaxique entre des systèmes d'informations hétérogènes. Dans la première partie de ce chapitre, nous présentons l'e-santé, un concept cœur de notre travail, ses enjeux sur les plans socio-économiques, médicaux et sociaux, et scientifiques. Nous décrivons par la suite l'exemple du projet européen EPI-MEDICS [7] qui a été piloté par notre équipe et autour duquel s'articule notre recherche. Nous mettons l'accent dans la dernière partie de ce chapitre sur le Dossier Médical Personnel (DMP) [8] et le concept d'interopérabilité des systèmes d'information et de standardisation. Un standard très connu en matière d'échange de données numériques médicales, HL7, sur lequel nous basons notre travail, est exposé à la fin de ce chapitre.

1.2 Le concept de l'e-santé

Le développement des TIC et l'évolution des systèmes d'information en santé constituent un levier majeur pour améliorer la qualité des services de santé et moderniser le système de soins. L'e-santé est une forme de pratique médicale à distance utilisant les technologies de l'information et de la communication. Selon le décret no 2010-1229 du 19 octobre 2010 [9], l'e-santé regroupe cinq actes médicaux.

- « La téléconsultation, qui a pour objet de permettre à un professionnel médical de donner une consultation à distance à un patient. Un professionnel de santé peut être

présent auprès du patient et, le cas échéant, assister le professionnel médical au cours de la téléconsultation ».

- « La téléexpertise, qui a pour objet de permettre à un professionnel médical de solliciter à distance l'avis d'un ou de plusieurs professionnels médicaux en raison de leurs formations ou de leurs compétences particulières, sur la base des informations médicales liées à la prise en charge d'un patient ».
- « La télésurveillance médicale, qui a pour objet de permettre à un professionnel médical d'interpréter à distance les données nécessaires au suivi médical d'un patient et, le cas échéant, de prendre des décisions relatives à la prise en charge de ce patient. L'enregistrement et la transmission des données peuvent être automatisés ou réalisés par le patient lui-même ou par un professionnel de santé ».
- « La téléassistance médicale, qui a pour objet de permettre à un professionnel médical d'assister à distance un autre professionnel de santé au cours de la réalisation d'un acte ».
- « La réponse médicale qui est apportée dans le cadre de la régulation médicale mentionnée à l'article L. 6311-2 et au troisième alinéa de l'article L. 6314-1 ».

1.3 Les défis du système de santé

Le système de santé doit faire face à un panorama de défis pour favoriser le développement des systèmes d'information de santé et promouvoir ainsi la santé et le bien-être [10]. Michel Gagneux, président du conseil d'administration de l'Agence des Systèmes d'Information Partagés de Santé (ASIP Santé) [11], souligne que le développement de la pratique de santé en France vers l'e-santé doit répondre à un ensemble de défis majeurs, liés notamment au vieillissement de la population, à l'augmentation accrue des maladies chroniques, à l'évolution de la démographie médicale, à l'accroissement des coûts de soins, etc. La Direction de la Recherche, des Etudes, de l'Evaluation et des Statistiques (DREES) [12] indique que d'ici dix ans, le nombre de médecins en France devrait baisser de 20 000 tandis que la population devrait croître de 10 %, et la consommation des soins devrait augmenter. Des tendances analogues sont observées un peu partout dans le monde. Pour éviter l'hospitalisation, des outils de maintien à domicile, par exemple, sont ainsi nécessaires. La pratique médicale doit aussi permettre l'accès aux soins pour tous et couvrir d'une façon homogène toute les zones géographiques.

1.4 L'e-santé et ses enjeux

L'e-santé apporte des éléments de réponse aux différents défis et comporte à la fois des enjeux sociétaux, économiques et scientifiques importants. Elle représente un marché en plein essor et un vecteur de croissance économique et de création d'emploi. Forte de son potentiel d'amélioration de la qualité de soins, l'e-santé a fait l'objet d'une attention croissante de la

part des industriels et des chercheurs scientifiques dans le monde. Nous détaillerons dans les sous-sections suivantes les enjeux majeurs de l'e-santé.

1.4.1 Enjeux socio-économiques

Depuis plus de vingt ans, l'Union Européenne [13] encourage et finance des projets de recherche coopératifs pour le développement d'une société d'information numérique au bénéfice des professionnels de santé et de tout citoyen, afin de faciliter et d'accélérer l'accès et l'échange des données consignées dans des dossiers médicaux répartis géographiquement en tout lieu, et de réduire la réplication d'examen inutiles et coûteux. Nombre d'initiatives ont ainsi été proposées, depuis le développement d'architectures intégratives et de standards fondés sur les premiers concepts d'« autoroutes de l'information » des années 90 appliqués à la télémédecine, jusqu'à la cyber-médecine et le *self-care* basés sur l'essor technologique de l'informatique ubiquitaire [14] [15]. A l'instar et grâce à l'expérience du département ICT (*Information and Communications Technology*) for Health de l'Union Européenne [16], l'Organisation Mondiale de la Santé (OMS/WHO) créa un observatoire en e-santé dès les années 2005 pour un renforcement et un développement extensif de l'usage des TIC pour la santé [5]. L'intérêt socio-économique de l'e-santé est à présent unanimement admis, bien qu'il soit difficile à quantifier de façon précise [17]. Le développement industriel des TIC pour la santé est devenu l'un des secteurs clés de l'économie de marché et de création d'emploi, et par voie de conséquence dans de nouveaux domaines d'activité tels que les services de soins à domicile, les offres d'hébergement de données, etc. Selon le réseau de télémédecine Medcom à l'hôpital universitaire d'Odense au Danemark, les bénéfices enregistrés grâce à l'utilisation du réseau de communication de télésanté au niveau national sont un gain de 50 minutes par jour et par médecin, une diminution de 66 % des appels téléphoniques de patients et une économie de 2,3 millions d'euros d'économie par transaction, soit 60 millions d'euros par an [18]. Dans le même contexte, une étude a été menée au Mali pour évaluer l'impact de l'e-santé dans les pays en développement sur l'utilisation, la qualité et les coûts des soins. « Les patients des sites du projet ont économisé en moyenne 12 380 Francs CFA ou 25 \$US, et un maximum de 35 000 Francs CFA ou 70 \$US, comparativement aux patients des établissements voisins qui ont dû se rendre dans la capitale pour obtenir les mêmes soins » [19].

1.4.2 Enjeux médicaux et sociaux

Le vieillissement de la population et l'augmentation des maladies chroniques placent aujourd'hui les architectures ou systèmes de télésurveillance à domicile et les systèmes pervasifs personnels de santé parmi les solutions de TIC à mettre en œuvre prioritairement. La surveillance médicale à distance des grossesses à risques, des personnes âgées, etc., devient ainsi un enjeu sociétal majeur visant à préserver le confort de vie privative et à assurer une prise en charge prompte et efficace tout en évitant des déplacements des patients inutiles, voire à risques. Elle permet également de réduire les coûts de prise en charge, de créer de nouveaux services à la personne et de réduire le risque de maladies handicapantes. Parallèlement au constat de vieillissement de la population, les maladies cardio-vasculaires, par exemple, deviennent aujourd'hui, d'après les statistiques de l'OMS, l'une des premières

causes de mortalité dans le monde. D'après le Prof. Gabriel Steg [20] : « On compte environ 120 000 infarctus du myocarde par an en France. Environ 10 % des victimes décèdent dans l'heure qui suit et le taux de mortalité à un an est de 15 % ». L'un des challenges majeurs en cardiologie est aujourd'hui de réduire le délai entre le début des symptômes et le début des traitements [21] [22]. L'espérance de vie des patients atteints d'une maladie cardiaque est très bonne à condition que les patients soient pris en charge en moins d'une heure par des équipes médicales spécialisées compétentes. De nouvelles stratégies sont donc nécessaires pour réduire ces délais. C'est ainsi que le développement des technologies sans fil et de l'informatique embarquée et ubiquitaire ont donné naissance à une nouvelle génération d'instrumentation médicale comme les systèmes personnels de santé [23].

1.4.3 Enjeux scientifiques

De nombreux projets ont été menés en Europe pour favoriser le développement des services numériques en santé. Les premiers projets en télémédecine ont été lancés en 1989 dans le cadre du programme *Advanced Informatics in Medicine* (AIM) dans le but d'améliorer la qualité et l'efficacité de la prise en charge des patients [24]. Ces travaux de recherche ont couvert plusieurs lignes d'action à savoir la sécurité et la confidentialité des systèmes d'information, la dématérialisation des données médicales et la standardisation de leurs échanges, leur stockage, les dossiers électroniques médicaux, l'aide à la décision médicale, etc.

Ces actions se sont poursuivies avec les programmes *Health Telematics et Information Society Technology* (IST) pour stimuler le développement des technologies de la santé. Plusieurs projets de recherche ont été menés dans le cadre du programme IST, qui se sont intéressés plutôt au développement des dispositifs médicaux personnels, embarqués, miniaturisés et intelligents. Nous citons à titre d'exemple le projet développé par notre équipe et soulevant des problématiques connexes à notre travail de recherche, le projet EPI-MEDICS [25] [1]. Une description plus détaillée de ce projet sera fournie Section 1.5.

En France, le GdR/ STIC-Santé CNRS/ INSERM 2647 a été créé en 2003 afin de favoriser et d'initier de nouvelles recherches à la croisée des domaines de la Santé et des TIC [26]. Ce GdR regroupe des recherches multidisciplinaires en e-santé. Il comporte cinq thèmes différents : A : Modélisation et simulation appliquées à la santé, B : Modélisation et simulation, C : Signaux et image en Santé. Systèmes d'information médicaux et bases de données, D : Capteurs, vêtements, habitats et réseaux intelligents en santé, E : Analyse, modèles et outils pour les handicaps : approche pluridisciplinaire, F : Apprentissage et assistance aux gestes médico-chirurgicaux, et G : e-Santé [27].

Plus récemment, le programme de l'union européenne pour la recherche et l'innovation, H 2020 a été mis en œuvre pour garantir l'excellence scientifique pour la période 2014-2020. L'une des priorités sur laquelle H 2020 a recentré de grands financements concerne les défis sociétaux. Parmi ces défis on trouve le renforcement de la prise en compte de la santé et du bien-être.

Les recherches en TIC-Santé ont fait appel à différents domaines de l'informatique. Nous citons ici deux domaines autour desquels s'articule notre problématique de recherche.

- Les systèmes d'information : L'objectif des systèmes d'information de santé actuels n'est plus seulement de stocker les informations mais aussi d'en faciliter le traitement et le partage. Héberger des données de santé à caractère personnel soulève plusieurs questionnements en termes de sécurité et de confidentialité. A cet effet, tout un référentiel d'hébergement des données de santé a été précisé dans le décret n° 2006-6 du 4 janvier 2006 [28]. Nombreuses sont les recherches publiées dévoilant l'importance des systèmes d'information dans la fiabilisation des décisions médicales et l'assurance des meilleures coordinations entre professionnels de santé au profit de la prise en charge du patient. Le traitement de l'information constitue un enjeu majeur dans le développement des pratiques médicales et l'amélioration de la qualité des soins. De fait, les systèmes d'information demeurent un moyen fondamental pour faciliter la prise de décision en améliorant l'accès aux données pertinentes et leur gestion [29] [30].
- L'intelligence artificielle : Connue sous l'acronyme IA, elle est définie par l'un des ses créateurs Marvin Lee Minsky comme « la construction de programmes informatiques qui s'adonnent à des tâches qui sont, pour l'instant, accomplies de façon plus satisfaisante par des êtres humains car elles demandent des processus mentaux de haut niveau tels que : l'apprentissage perceptuel, l'organisation de la mémoire et le raisonnement critique » [31]. L'IA permet de résoudre, avec une machine de Turing, des problèmes imitant le comportement humain, à savoir la compréhension des langues naturelles, l'analyse de la parole, l'interprétation des images, etc. L'e-santé a notamment besoin des technologies de l'IA. Les méthodes d'aide à la décision sont utilisées pour le diagnostic des états cliniques du patient, le suivi automatique des personnes âgées à domicile, la génération des alertes basée sur les méthodes d'apprentissage automatique, etc. Les systèmes experts rendent possible la reproduction des mécanismes cognitifs des experts en santé. Le plus connu fut peut être MYCIN, qui a été développé dans les années 70 par Shortliffe et Buchanan à l'université de Stanford [32]. Ce système expert dispose d'une base de connaissances et d'un moteur d'inférence pour permettre le diagnostic d'une maladie du sang, identifier des bactéries à l'origine d'infections graves et prescrire des médicaments. Une évaluation de MYCIN a démontré son efficacité et sa grande capacité à fournir des prescriptions d'antibiotiques antimicrobiens plus adéquates que celles données par des prescripteurs humains [33].

1.5 L'exemple du projet européen EPI-MEDICS

Nous présentons dans cette section l'exemple du projet EPI-MEDICS pour la simple raison qu'il soulève des problématiques de gestion d'acheminement des messages connexes à notre travail de recherche (Chapitre 2). Le projet EPI-MEDICS (*Enhanced Personal Intelligent & Mobile System for Detection & Interpretation of Cardiovascular Syndromes*) a

pour objectif de répondre aux besoins médicaux et sociaux décrits dans la section 1.4.2. EPI-MEDICS propose la création d'un nouveau système permettant la détection précoce des arythmies et des ischémies, le PEM (Personnel ECG Monitor) [34]. Le PEM est un système personnel, portable et intelligent de recueil d'ECG à la demande par le patient lui-même, d'analyse automatique des ECGs enregistrés et, si nécessaire, de notification d'alerte avec convoiement du dossier médical embarqué à divers types de destinataires cibles selon le degré de gravité de l'incident détecté. Il est doté de technologies sans fil, d'une carte à mémoire de type SD card pour le stockage du dossier médical et des ECG, et d'un serveur web embarqué. Les ECG sont encodés et stockés au format SCP-ECG, qui est un standard ISO d'échange des signaux ECG numériques [35]. Le PEM a toutes les caractéristiques requises d'un électrocardiographe professionnel selon les recommandations et standards en vigueur. C'est un petit boîtier compact de faible encombrement, qui peut tenir dans une main. L'ECG est enregistré selon un système de positions d'électrodes original qui a été élaboré dans notre équipe pour être facile à appliquer par tout individu néophyte, mais qui présente trois atouts majeurs par rapport aux solutions existantes [34] : 1) c'est un système d'enregistrement 3D des signaux ECG permettant de recueillir la quasi-totalité de l'information spatiotemporelle de l'activité électrique cardiaque, ce qui est indispensable pour la détection des maladies coronariennes de type infarctus ou ischémie [36]; 2) il est basé sur les dérives standards qui sont habituellement utilisées par les médecins, ce qui facilite et rend plus fiable le contrôle de qualité humain, si nécessaire a posteriori, des signaux recueillis ; 3) les trois dérives recueillies constituent un sous-ensemble des dérives ECG standard : l'une (V2) classique pour tout type d'ECG et fondamentale pour l'évaluation des maladies coronariennes, les deux autres (I et II en position Mason-Likar) étant conformes aux dérives périphériques utilisées de façon standard en électrocardiographie d'effort, afin de s'affranchir des effets de bruit induits par les mouvements des membres, ce qui est fondamental lorsque l'ECG (dit de repos) doit être recueilli par le patient ou l'utilisateur lui-même. Ce système d'enregistrement permet en outre la reconstruction fiable d'un ECG standard utilisé habituellement par les médecins [37]. Le PEM embarque ainsi des composants métiers de synthèse de l'ECG, selon deux stratégies : l'une générique applicable à tout individu, l'autre personnalisée à chaque individu qui dispose d'un ECG antérieur standard de référence. Cette dernière stratégie a été conçue et évaluée à grande échelle dans le cadre d'une thèse de doctorat au sein de l'équipe de recherche [38]. Elle est basée sur une architecture de comité de cent réseaux de neurones [37], dont la phase d'apprentissage pour la détermination des paramètres de transformation, à télécharger dans le PEM, peut être réalisée via un service web, utilisant des ressources de grille de calculs permettant alors de fournir les résultats en quasi-temps-réel.

L'intelligence embarquée dans le PEM comprend deux phases : 1) la phase d'analyse se décomposant en une phase de traitement des signaux à l'instar des systèmes ECG de type professionnel (correction de la ligne de base, filtrage, reconnaissance et classification des ondes ECG, calcul de la fréquence cardiaque, détermination d'un battement type, délimitation des ondes P-QRS-T, calcul des paramètres ECG) [39], et qui a été certifiée par l'action internationale CSE [40] ; 2) la phase d'aide à la décision comprenant l'analyse et l'interprétation du rythme, l'analyse des formes d'ondes dans le temps et dans l'espace et le calcul d'une probabilité diagnostique du risque d'infarctus ou d'ischémie. Cette probabilité

résulte de l'agrégation/composition par logique floue d'une probabilité du risque cardiaque obtenue à partir d'un réseau bayésien fusionnant des facteurs de risque (cholestérol, diabète, fumeur, obésité, etc.) issus des antécédents archivés dans le dossier médical embarqué, et de la probabilité de risque déterminée par un comité de réseaux de neurones basé sur un ensemble de descripteurs ECG [41]. Cet ensemble résulte à son tour d'une analyse en composantes principales des paramètres calculés sur le dernier ECG recueilli et de leurs variations avec les paramètres calculés sur un ECG de référence antérieur, s'il existe. Une évaluation médicale, sur une base de données d'environ 3700 patients, de la qualité diagnostique de l'intelligence embarquée a montré une performance analogue, voire supérieure, à celle d'un cardiologue expérimenté [38].

Scénarios d'alarmes

Le modèle d'information et de communication embarqué supporte différents scénarios d'intelligence ambiante. Selon la probabilité diagnostique de survenue d'un événement cardiovasculaire à risque pour le patient, obtenue à l'issue du processus décisionnel lors de l'acquisition d'un ECG, quatre niveaux d'actions sont envisagés [1] [34]. Dans le cas où aucune suspicion de l'apparition d'une quelconque pathologie n'est décelée, un message réconfortant, directement affiché sur l'écran, est fourni au patient. Si une anomalie de degré mineur est suspectée, nécessitant l'avis d'un médecin mais sans présenter un caractère d'urgence, des recommandations seront fournies au patient pour qu'il prenne un rendez-vous de consultation avec son médecin traitant. Ce dernier pourra alors consulter le dossier embarqué et les ECG collectés par le patient grâce à une connexion Bluetooth, un navigateur et une application de visualisation d'ECG compatible avec le standard SCP-ECG [35]. Si au contraire, une pathologie sévère est suspectée ou détectée, un message est automatiquement envoyé via GPRS à un serveur d'alarme. Ce dernier reçoit alors successivement au format XML une description de la cause de l'alerte, les données permettant de localiser et d'identifier le patient, puis le dossier médical du patient et, au format SCP-ECG, l'ECG ayant déclenché l'alerte et, s'il mémorisé dans la SD card, le dernier ou l'avant dernier ECG du patient et éventuellement un ECG antérieur qualifié de tracé de référence. Deux situations peuvent alors se produire selon la gravité et le degré de certitude de l'événement détecté et selon les indications qui auront été mentionnées dans le dossier patient [25] [42]. Dans le cas d'une alarme majeure, le message pourra être transmis directement à un service de régulation des urgences vitales, tel que le SAMU en France qui pourra immédiatement consulter le dossier sur un serveur dédié et agir en conséquence. Dans le cas où l'occurrence d'une pathologie est suspectée de façon incertaine mais qui doit être confirmée assez rapidement, le message pourra être transmis à un serveur d'alarmes hébergé par un centre d'appel dont les coordonnées ont été documentées dans le dossier embarqué du patient. Ce serveur envoie alors un SMS au médecin traitant du patient, cardiologue ou généraliste, dont les coordonnées sont référencées dans le dossier transmis. Ce SMS indique l'objet du message et les coordonnées du patient concerné, l'adresse du serveur d'alarmes, un identifiant et un mot de passe générés aléatoirement permettant au médecin de se connecter à ce serveur et de consulter le dossier correspondant via Internet, afin de pouvoir décider des actions à entreprendre pour la prise en charge de ce patient [2].

L'architecture logicielle embarquée a été conçue comme un système multi-agents dirigé par des modèles qui invoque les différents composants de façon dynamique en fonction des données disponibles. C'est une architecture modulaire multicouche, afin de faciliter la mise à jour éventuelle des composants.

1.6 Le Dossier Médical Personnel (DMP) et interopérabilité des systèmes d'information médicaux

Le besoin de partager les données médicales entre patients et professionnels de santé ou entre professionnels de santé augmente et s'impose. Le Dossier Médical Personnel (DMP) est une réponse efficace au besoin de partage et d'échange de l'information médicale. Il représente un enjeu majeur pour améliorer la qualité des soins et favoriser leur coordination. Il permet de mettre le patient au cœur du système de santé. Le patient devient maître de son dossier. Il décide de sa création et de sa fermeture, de la gestion d'autorisations d'accès, etc. Néanmoins, l'étude de la littérature médicale a prouvé la nécessité d'un DMP unique et commun, afin que son contenu soit interprété à l'identique par les différents acteurs du système de santé. Ce besoin de partage de DMP d'une façon structurée a abouti à l'adoption de standards internationaux tels que HL7. Dans ce contexte, nous présentons plus en détail ci-après le DMP et ses enjeux. Nous mettons l'accent sur la nécessité de standardisation et nous présentons par la suite le standard HL7.

1.6.1 Le DMP

Créé par la loi du 4 mars 2002 [43] et rebaptisé en 2015 dossier médical partagé, le dossier Médical Personnel (DMP) comporte un ensemble de données informatisées (antécédents médicaux, coordonnées personnelles du patient, résultats de laboratoires d'analyse, d'imagerie, etc.). Ces données sont accessibles par Internet avec l'accord préalable du patient. Le DMP est un projet catalyseur qui a permis la mise en place du cadre d'interopérabilité en santé et a suscité l'émergence de toute une série de services « modernes » en santé. L'une des vocations de la recherche en e-santé est de dématérialiser les données cliniques et de rendre le DMP unique, personnel, partagé et réparti. A cet égard, une définition des exigences en matière de sécurité des données médicales du DMP est nécessaire pour créer un espace de confiance et de confidentialité pour l'échange et le partage des données entre différents systèmes d'information de santé (SIS).

1.6.2 Standards internationaux et protocoles de communication en santé

Principal frein au développement du secteur de l'e-santé, l'absence d'un ensemble de normes clairement identifiées pour l'échange de données médicales crée une situation de perte de consensus, de cohérence et d'interopérabilité tant sur le plan sémantique que sur le plan syntaxique entre les différents acteurs de santé.

Plusieurs organismes de standardisation dans le domaine de l'e-santé ont produit des standards pour assurer l'échange, le partage et l'exploitation des données de santé. Nous citons à titre d'exemple HL7 pour les échanges informatisés de données cliniques, financières

et administratives [4], DICOM (*Digital Imaging and Communications in Medicine*) pour la gestion des images médicales [44], SCP-ECG (*Standard Communications Protocol for computer assisted electrocardiography*) qui est un format d'enregistrement numérique des électrocardiogrammes (ECG) et des métadonnées associées [35].

1.6.3 Nécessité d'une standardisation

1.6.3.1 Interfaçage des systèmes

Demander un conseil à un médecin traitant, demander l'avis médical d'un expert à distance, consulter les résultats d'examens d'imagerie, etc., toutes ces tâches sollicitent des systèmes d'informations hétérogènes (système de radiologie, système de laboratoire, systèmes de soins infirmiers, système d'acquisition d'images, système administratif, système d'information clinique, etc.) qui doivent communiquer pour échanger des données médicales concernant un patient. Néanmoins, pour faire communiquer n systèmes entre eux et assurer la totalité des échanges envisageables, il faut maintenir $n \times (n-1)$ interfaces (Figure 1).

Aujourd'hui, construire un système d'information hospitalier (SIH) complet dans un établissement de santé nécessite de 30 à 40 applications différentes, soit de 870 à 1560 interfaces. On conçoit ainsi que la difficulté d'intégration des systèmes hétérogènes s'accroît au fur et à mesure que le nombre de systèmes croît.

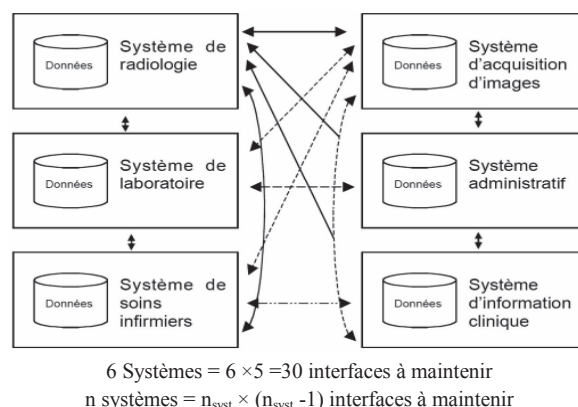
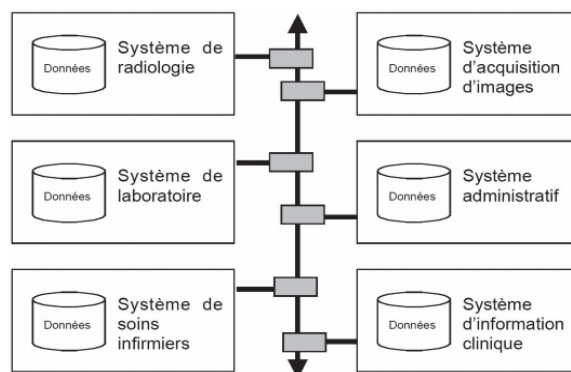


Figure 1. Limites des interfaçages de systèmes [45]

Un protocole commun qui garantit l'interopérabilité des systèmes d'information en réduisant la complexité de communication et le nombre d'interfaces à maintenir est ainsi nécessaire (Figure 2). Avec un protocole commun, les n systèmes devront communiquer avec seulement n interfaces.



6 Systèmes = 6 interfaces à maintenir ; n systèmes = n interfaces à maintenir

Figure 2. Avantages d'un protocole commun pour garantir l'interopérabilité entre différents systèmes [45]

1.6.3.2 Interopérabilité des systèmes d'information en Santé

L'un des grands challenges en e-santé est de garantir l'interopérabilité des systèmes d'information en santé. L'interopérabilité des systèmes est la capacité d'échanger et de partager de l'information entre eux tout en préservant le sens de l'information dans l'espace et dans le temps. En effet, l'interopérabilité doit garantir la sémantique, c'est-à-dire que l'information échangée doit être interprétée à l'identique entre un SI cible et un autre initiateur. L'ASIP a établi le référentiel Cadre d'Interopérabilité des Systèmes d'Information de Santé (CI-SIS) [46] qui constitue un ensemble de règles et de spécifications ayant pour vocation particulière d'améliorer la qualité des soins, de développer la coordination entre les SIS et l'aide à la décision médicale, tout en préservant la sécurité des données personnelles de santé. Le CI-SIS couvre deux types d'interopérabilité : l'interopérabilité technique et l'interopérabilité des contenus métier. L'interopérabilité technique couvre le transport des données médicales et les services qui permettent l'échange et le partage de ces données. L'interopérabilité des contenus métiers s'adresse plutôt au contenu et à la sémantique du contenu échangé entre les systèmes d'information. D'autre part, l'IHE (*Integrating the Healthcare Enterprise*) a aussi opté pour l'amélioration du partage et de l'échange des données médicales en assurant l'interopérabilité entre les différents SIS. L'IHE a pour objectif principal de définir des profils de standards, appelés profils d'intégration pour la dématérialisation, l'échange et le partage de données de santé. Ces profils regroupent des informations concernant tous les acteurs et les transactions nécessaires pour réaliser de grandes fonctions tels que prescription, réalisation d'examen radiologique et partage des données associées, recherche de l'identité d'un patient ou de sa situation courante dans l'établissement, etc.

En France, l'association Française Interop'Santé a œuvré pour le déploiement des normes d'interopérabilité et la promotion des échanges d'information de santé. Elle s'appuie aujourd'hui sur plusieurs organisations de standardisation des échanges informatiques dans le domaine de la santé tels que HL7, HPRIM (Harmoniser et Promouvoir l'Informatique Médicale).

1.6.4 Le Standard HL7

Le principe de standardisation, comme nous l'avons vu, est nécessaire pour pallier les limites d'interfaçages entre les systèmes d'information de santé. Les standards en santé sont nombreux (CEN/TC 251, ISO/TC 215) et méritent l'attention mais ne sont pas tous examinés ici. Nous présentons dans cette section le standard HL7 sur lequel nous appuyons notre travail.

1.6.4.1 Définition

Fondé en 1987, *Health Level Seven* (HL7) est un standard qui définit un ensemble de spécifications pour l'échange informatisé des données cliniques, financières et administratives entre systèmes d'information hospitaliers (SIH) [4]. En 1994, HL7 a été désigné comme une organisation de développement des standards (SDO) accrédité par l'ANSI (*American National Standards Institute*). Initialement américain, HL7 tend à s'imposer en tant que standard international reconnu par l'organisation internationale de normalisation (ISO). Le chiffre 7 d'HL7 indique que ce standard fonctionne au niveau de la couche 7 du modèle OSI (*Open Systems Interconnection*), la couche application.

1.6.4.2 Les versions HL7

Trois versions successives d'HL7 ont été élaborées. La version 1 (1987) était un prototype peu implanté. La série des versions 2 dont la plus récente est la version 2.6 (2006), définit la structure d'un message en HL7. En fait, un message HL7 version 2 est composé d'un ensemble de segments, eux même subdivisés en champs. Ces messages peuvent avoir pour objet la transmission d'un avis médical, de résultats de laboratoire, d'une prescription, de données relatives à la gestion administrative des patients, d'un compte rendu d'hospitalisation, etc. Néanmoins, ces versions ne reposent pas sur des modèles sous-jacents. L'objectif de conception et le fondement même de la conception des messages sont en grande partie absents dans la version 2 de HL7. Le contenu du message est basé sur un modèle d'information implicite et non explicite. En outre, cette version ne fournit pas de documentation pour la spécification des vocabulaires pour décrire la conformité des concepts. Les messages HL7 versions 2 peuvent contenir des champs de données optionnels ce qui fait que leur interprétation peut être différente.

Dans l'intention d'atténuer les problèmes des anciennes versions, le consortium HL7 lance, en 1997, le développement de la version 3. La version 3 d'HL7 s'appuie sur des fondements conceptuels et méthodologiques solides garantissant sa cohérence globale. Elle repose sur le langage XML et une modélisation orientée objet pour la représentation des messages. En effet, cette représentation découle d'une réflexion autour d'un modèle d'information cohérent et partagé, c'est le modèle de référence (*Reference Information Model* « RIM »). Le RIM est exprimé en diagrammes de classe UML (*Unified Modeling Language*). Il constitue la source des données à partir desquelles tous les autres modèles d'information et les spécifications des messages HL7 sont dérivés. Une description plus détaillée du RIM sera exposée dans la section suivante. HL7 a établi un ensemble de spécifications des types de données (norme

ISO/HL7 21090 : 2009) qu'il associe au RIM. Ces spécifications définissent le format structurel des données des attributs du RIM. HL7 associe également au RIM des spécifications du vocabulaire qui définissent un ensemble de concepts considérés valides dans l'instanciation d'un attribut ou d'une propriété de type de données. HL7 a conçu une architecture commune pour les documents cliniques, nommée « *Clinical Document Architecture* » (CDA) (Figure 3). Cette architecture définit la structure et la sémantique d'un document clinique. Un document CDA est organisé en deux parties : un entête et un corps. L'entête définit l'identifiant du document, son titre, son auteur, etc., le corps représente le contenu textuel qui peut contenir du texte, une image, etc. Ce contenu est sémantiquement structuré et se réfère aux spécifications du RIM et aux référentiels terminologiques admis, comme *Logical Observation Identifiers Names and Codes* (LOINC). Un document CDA est encodé en XML.

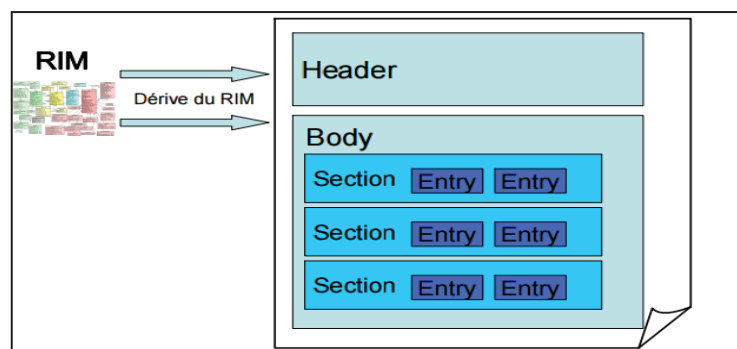


Figure 3. Principe d'un document CDA d'HL7 [4]

1.6.4.3 Le RIM

Comme évoqué précédemment, le HL7 RIM représente la source du contenu de tous les messages HL7 version 3. Le RIM est représenté par un diagramme de classe, dont les attributs de chaque classe constituent une clause dans la structure du message. HL7 fournit des spécifications du vocabulaire pour associer à chaque attribut des valeurs valides. Pour plus de détails, ces valeurs sont téléchargeables, visualisées et gérées avec l'outil *HL7 Rose TreeTool* [4].

1.6.4.3.1 Les éléments du RIM

Une étude de l'organisation HL7 en avril 1996 indique que le RIM est composé de 114 classes, 536 attributs, 159 classes association, 27 relations de spécialisation/généralisation, 2 relations d'agrégation et 37 *Subject Areas*. Un extrait d'HL7 RIM est disponible dans [47]. Le RIM comporte six classes fondamentales :

- *Entity* : C'est un objet physique ou une organisation ou un groupe d'objets physiques capables de participer dans des actes. *Entity* peut être une personne, un matériel, un dispositif, etc.
- *Act* : C'est une action exécutée dans un contexte médical. Les ordonnances médicales, les plans de soins représentent des exemples d'actes.

- *Act relationship* : est une association entre deux actes, telle qu'une relation de succession/précédence ou cause/effet, par exemple.
- *Role* : C'est la fonction jouée par l'entité impliquée dans l'acte. Un rôle peut être : employé, patient, médecin. Une entité peut jouer plusieurs rôles (*Employee, provider, practitioner, specimen, patient, etc.*) et un rôle est associé à une entité.
- *Participation* : est une association n-aire entre les classes « *Role* » et « *Act* » représentant la fonction assumée par le rôle dans le contexte d'un acte.
- *Role link* est une association entre deux instances différentes de rôles.

Un *subject area* représente une collection de classes liées à une classe mère. Par exemple, le regroupement des classes fils de la classe « *Entity* » forme un « *Entity subject area* ». La Figure 4 [48] illustre une instantiation de quelques valeurs des attributs des classes cœurs du HL7 RIM.

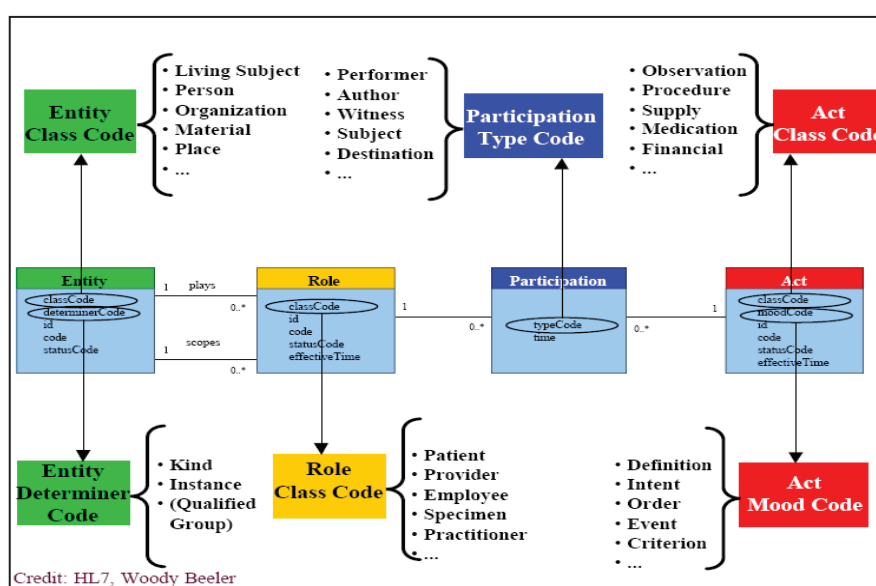


Figure 4. Valeurs des attributs des classes cœurs du RIM d'HL7 [4]

Nous donnons ci-après deux exemples d'instanciation des classes du RIM : *entity*, *role*, *participation*, *act* (Figure 5).

- L'entité « *Personne A* » joue le rôle d'un « *Médecin* » en participant en tant que « *Réalisateur* » dans l'acte « *Auscultation* ».
- L'entité « *Personne B* » joue le rôle d'un « *Patient* » et participe en tant que « *Objet* » dans l'acte « *Auscultation* ».

Un exemple de message HL7 version 3 illustrant l'échange de résultats de laboratoire est fourni dans le Chapitre 7.

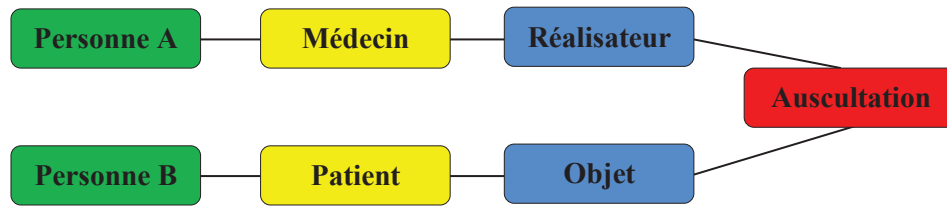


Figure 5. Exemple d’instanciation des classes « Entity », « Role », « Participation » et « Act » du RIM

Le RIM fournit deux sujets appelés « *subject areas* » [4], qui traitent d’un objet cœur dans notre problématique, le message. Ces « *subject areas* » sont : « *message control subject area* » et « *Query Control subject area* ». Le « *message control subject area* » contient toutes les classes impliquées dans le contrôle, la communication et l’acquiescement des messages. Quant au « *Query Control subject area* », il contient les classes nécessaires pour formuler les messages, communiquer et répondre aux messages de requête. Dans ce modèle, nous trouvons les classes « *message* », « *acknowledgement* », « *transmission* », « *query spec* », « *communication function* », etc. « *Query spec* » définit les critères qui seront appliqués sur la requête et sur le message de réponse tel que : mode de réponse (différé/immédiat). L’entité qui participe à la transmission d’un message est appelée *Communication Function*. *Communication Function* peut prendre trois valeurs (*sender*, *receiver*, ou *respond to*).

1.6.4.3.2 Processus de développement d’un message HL7 version 3

Le processus de développement d’un message HL7 version 3 se compose de 5 étapes décrites ci-après.

L’analyse des besoins : Elle permet aux utilisateurs, analystes et développeurs de définir et de représenter formellement les frontières et responsabilités du système étudié. Cette étape délivre un modèle basé sur l’analyse de cas d’utilisation réalistes appelé *Use case Model*. Un exemple simplifié d’analyse de cas d’utilisation : administration du patient, est illustré Figure 6. La démarche d’analyse des besoins est un processus itératif. Elle aide à identifier les besoins fonctionnels, à élaborer les plans de test et la documentation utilisateur.

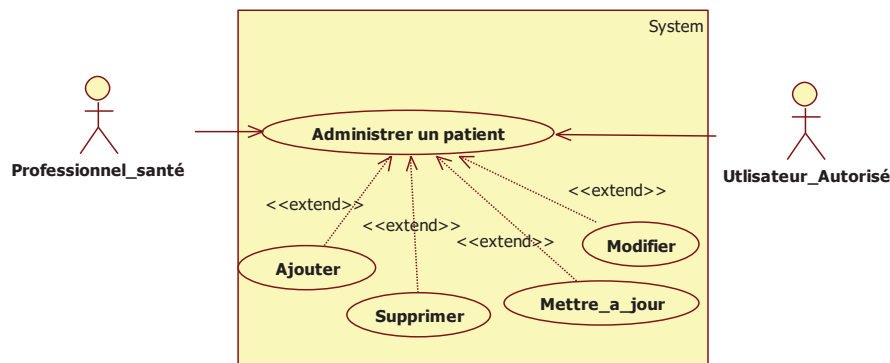


Figure 6. Représentation UML du cas d’utilisation « administration du patient »

L’analyse du domaine : Elle délivre un modèle d’information du domaine « *Domain information model* » (DIM). Ce modèle représente le contenu informationnel requis pour un

domaine donné dans HL7 (Pharmacie, Laboratoire, Administration du patient, etc.). Il est aussi utilisé pour établir des changements dans la conception du RIM.

La définition des structures des interactions : Une fois le DIM défini, l'élaboration du modèle d'interactions (*interaction model*) peut être entreprise. Le modèle d'interactions est équivalent à un diagramme de séquence en UML. Il permet d'illustrer, selon un ordre chronologique, l'interaction entre l'émetteur et le récepteur d'un message HL7 dans le cadre d'un scénario spécifique. Il permet d'identifier les événements déclencheurs de la transmission du message (*Trigger event*), les rôles de participation à l'échange (*Application role*), les besoins qui doivent être satisfaits pour le déclenchement de l'événement, et les responsabilités des destinataires du message (Figure 7).

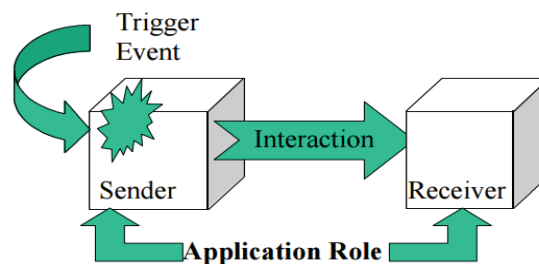


Figure 7. Eléments du modèle d'interactions d'HL7 [4]

Nous donnons Figure 8 un exemple simplifié de modèle d'interaction pour le cas d'utilisation : « ajout d'un patient ».

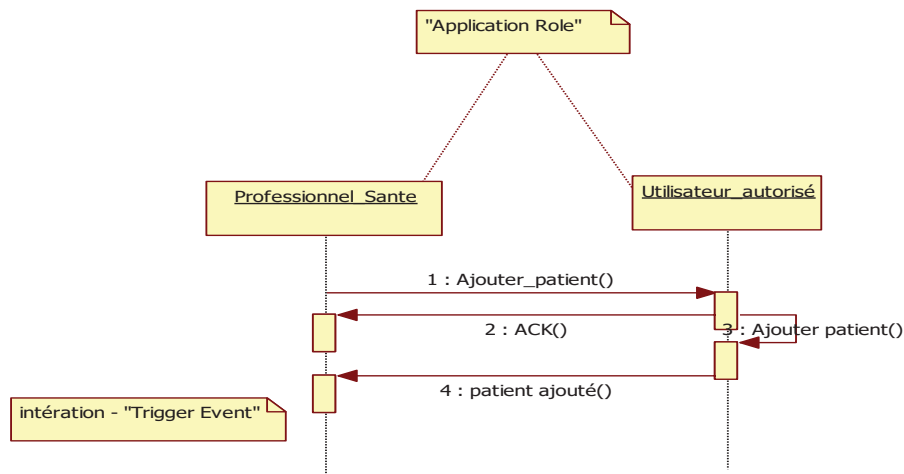


Figure 8. Modèle d'interaction pour le cas d'utilisation : « ajout d'un patient »

La définition des structures de messages : De cette étape résulte le modèle « *Hierarchical Message Description* » (HMD). Comme son nom l'indique, ce modèle décrit la structure du message et définit l'ensemble des attributs inclus dans chaque type de message.

Le vote est la dernière étape du processus. Il s'agit de soumettre les modèles précédemment délivrés au vote des membres HL7.

1.6.4.3.3 Cadriciel de développement d'un message HL7

En plus du RIM, HL7 définit trois autres types de modèles d'information : D-MIM (*Domain Message Information Model*), R-MIM (*Refined Message Information Model*) et HDD (*Hierarchical Message description*) (Figure 9). Le D-MIM est un dérivé du RIM. Il inclut un ensemble de classes, d'attributs et de relations utilisées pour la création des messages dans un domaine particulier. Le R-MIM dérive du D-MIM et constitue un sous-ensemble de classes spécifiant le contenu informationnel des messages dans un domaine spécifique. Enfin, le HDD intervient sur un niveau plus opérationnel dans le processus de spécification du message. Il décrit la hiérarchie du message qui sera utilisable par les industriels. Le choix des classes et des attributs du RIM ainsi que leurs instanciations diffèrent d'un domaine à un autre selon les règles et les conditions organisationnelles qu'exige chaque domaine.

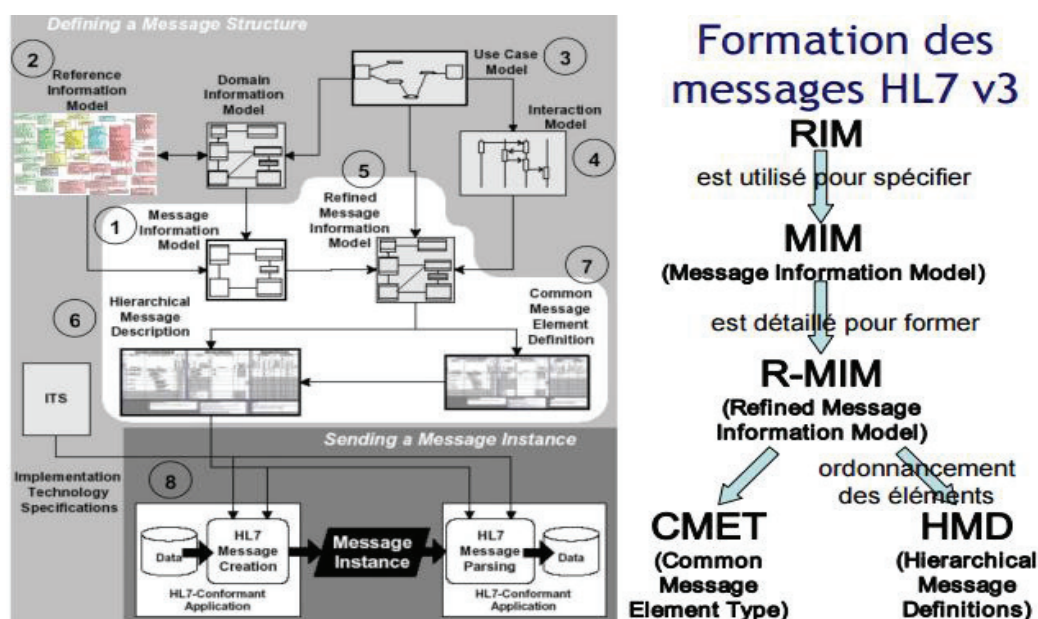


Figure 9. Formation des messages HL7 version 3 [4]

A la fin nous aurons un message au format XML prêt à être échangé. L'exemple présenté Figure 10 montre comment les éléments du RIM se traduisent en balises XML dans un message HL7. L'acte « prescription » est ici lié à deux participations « auteur » et « sujet ». Le sujet est ici un patient qui est de type « personne » et est caractérisé par un identifiant, possède une adresse, etc. Ces données sont définies comme des balises XML structurant le message HL7.

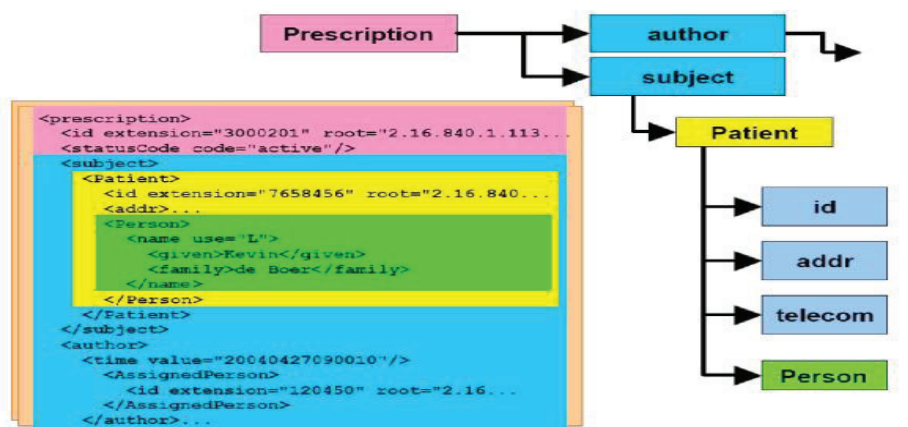


Figure 10. Instanciation XML d'un message HL7 [4]

Le standard HL7 version 3 regroupe toute les « spécialités » et les différents types de messages en santé. Un package HL7 version 3 est organisé en sections et sous-sections. Chaque sous-section est structurée en sous domaines. Par exemple, la section « *Health Clinical Management (HM)* » est organisée en trois sous-sections « *Operations* », « *Reasoning et records* ». Sous la section « *Operation* », on trouve par exemple les domaines : « *laboratory* », « *pharmacy* », « *imaging integration* », etc.

1.7 Conclusion

Nous avons présenté dans ce chapitre le contexte de notre travail qui s'inscrit dans le domaine de l'e-santé. L'e-santé est une pratique médicale nouvelle et porteuse d'espoir face à l'explosion démographique, au vieillissement de la population et à la croissance des maladies chroniques dans le monde. L'e-santé représente un moyen puissant pour améliorer la qualité de prise en charge voire la qualité de vie des patients/citoyens. Ces nouvelles pratiques s'appuient aussi bien sur des innovations technologiques, que sur des valeurs humaines. L'e-santé permet ainsi de créer un environnement de coordination et de collaboration non seulement entre professionnels de santé mais aussi entre professionnels de santé et patient. Les services de soins évoluent vers une prise en charge pluridisciplinaire qui rend le patient au centre des préoccupations et donc davantage acteur responsable dans la prise en charge de sa santé. Outre ces enjeux, l'e-santé permet notamment de réduire les coûts dans un processus de soins en termes de temps et de dépenses hospitalières. Elle permet en effet d'une part d'éviter les déplacements des patients et des professionnels de santé et de supprimer les consultations ou les hospitalisations non nécessaires, et d'autre part de mieux organiser le temps de travail des professionnels de santé. Cette vision demande l'échange et le partage des données médicales entre différents systèmes d'information. Pour garantir l'interopérabilité et établir un consensus syntaxique et sémantique entre ces systèmes d'information, plusieurs consortiums ont développé des standards, à savoir HL7, DICOM, SCP-ECG, etc. Dans ce contexte, une définition des exigences en matière de sécurité doit être mise en œuvre pour favoriser un espace de confidentialité et de respect de la vie privée des patients. Le progrès des technologies de l'information et de la communication ont conduit à la mise en place d'un univers intelligent via notamment le développement des dispositifs mobiles et intelligents qui détectent des problèmes de santé et envoient automatiquement des messages d'alerte ou à titre

informatif aux professionnels de santé. La gestion d'acheminement de ces messages doit ainsi prendre en considération les changements contextuels continus à savoir la mobilité aussi bien du patient ou du professionnel de santé tout en couvrant tous les domaines métiers en e-santé et sans pour autant nuire à la sécurité des systèmes d'information.

Le chapitre suivant sera ainsi consacré à évoquer la problématique de gestion d'acheminement des messages en Santé ainsi que l'ensemble des défis à surmonter. Parallèlement, nous montrerons la nécessité d'adaptation intelligente en temps réel de la stratégie de transmission des messages en e-santé face à la complexité des cas d'usage.

Chapitre 2

La gestion de l'acheminement des messages en e-santé. Défis et besoins

2.1 Introduction

L'acheminement des messages est la circulation de l'information entre un acteur source et un ou des acteurs cibles. Cela paraît simple, mais dans l'univers de la santé, la problématique de l'échange des données médicales entre différents systèmes d'information revêt un caractère complexe lié à de nombreux facteurs : la diversité et la complexité des processus métiers, le respect des spécificités des standardisations internationales, l'évolutivité des systèmes d'information pervasifs, la dynamique du contexte, la sécurité des données, etc. L'enjeu est pourtant énorme car il s'agit de garantir au sein d'un établissement de santé, voire au sein d'un système de soins dans son ensemble, une bonne qualité de service d'acheminement des messages qui, sur un plan opérationnel, s'adapte automatiquement aux changements permanents du contexte, aux évolutions en temps réel, liés à l'indisponibilité soudaine de professionnels de santé, à la mobilité du patient, à l'aggravation de l'état clinique du patient, etc. En e-santé, on assiste de plus en plus à une augmentation du taux de morbidité et de mortalité liée à un mauvais pilotage d'acheminement des messages. Dans ce contexte, plusieurs questions se posent : Quelle décision doit être prise si par exemple un message d'alarme n'a pas été traité par un professionnel de santé ? Faut-il rediriger le message ? Si oui, vers quel type de professionnel de santé ? Quelle stratégie d'acheminement doit être prise si une aggravation des signes cliniques est survenue après le transfert du message ?

Dans ce chapitre, nous présentons dans un premier temps les défis d'un système de gestion d'acheminement des messages en e-santé. Dans les défis, nous présentons les contraintes levées par la nature ubiquitaire de l'environnement. Pour illustrer ces contraintes, nous nous sommes basés sur deux exemples de scénarios différents et réels issus de deux projets de notre équipe: Le projet européen EPI-MEDICS [1] détaillé dans le chapitre 1 et le projet national Télé-infirmière [2]. Nous présentons ensuite l'ensemble des requis du système. Nous terminons ce chapitre par une analyse des travaux connexes à notre problématique de recherche en identifiant leurs avantages et limitations.

2.2 Défis des systèmes de gestion des acheminements des messages en e-santé

2.2.1 Exemples de scénarios

Nous allons présenter deux exemples de scénarios d'échanges d'information médicales en cardiologie dans le cadre de deux projets auxquels notre laboratoire a participé : le projet européen EPI-MEDICS présenté dans le chapitre 1 et le projet national RNTS¹ Télé-infirmière [2]. Ils permettront, dans le cadre de ce chapitre, d'illustrer les problèmes et les besoins d'un système de gestion d'acheminement des messages en environnement ubiquitaire.

2.2.1.1 Scénario d'alarme moyenne du projet EPI-MEDICS

Un patient : Monsieur X a des antécédents de maladie cardiovasculaire. Il dispose du PEM, un dispositif médical portable et intelligent qui permet la détection précoce des arythmies et des ischémies (Figure 11). Ce système embarque le dossier médical de Monsieur X. Le dossier comporte, à titre d'exemple :

- Antécédents.
- allergies.
- Commentaires.
- Traitements.
- Résultats d'examens effectués (ECG de repos, ECG-HR, pression artérielle, échocardiographies, radiographies, examens de laboratoires, coronarographie, IRM)
- Angioplastie (acte réalisé, résultats).
- Les facteurs de risque cardiovasculaires (hypertension, cholestérol, hygiène de vie).
- Nom et adresse du médecin traitant.

Le suivi du dossier médical de Mr X s'effectue sous la responsabilité de son médecin traitant Y. En faisant du ski, le patient X a senti des douleurs thoraciques. Un message d'alarme moyen est ainsi envoyé à son médecin traitant Y.

Le protocole de détection d'un événement et de transmission d'alarme moyenne est le suivant :

- Le système recueille les 12 dérivations pseudo-orthogonales de l'électrocardiogramme ECG,
- Le système analyse l'ECG nouvellement acquis et quantifie ses différences par rapport à un ECG de référence stocké,
- Il génère les messages d'alarmes et transfère l'ensemble des informations au serveur d'alarme.

¹RNTS (Réseau National sur les Technologies pour la Santé soutenu conjointement par les ministères de la santé et de la recherche).

- Dès réception, le serveur d'alarme envoie un SMS au médecin référent (cardiologue ou généraliste). Ce SMS indique le niveau et la raison de l'alarme, l'adresse URL du serveur d'alarme, le numéro de téléphone du patient, un identifiant et un mot de passe générés aléatoirement.
- Le médecin traitant Y se connecte sur le serveur d'alarme via une connexion internet, consulte les informations disponibles pour le patient et prend les décisions qui s'imposent.

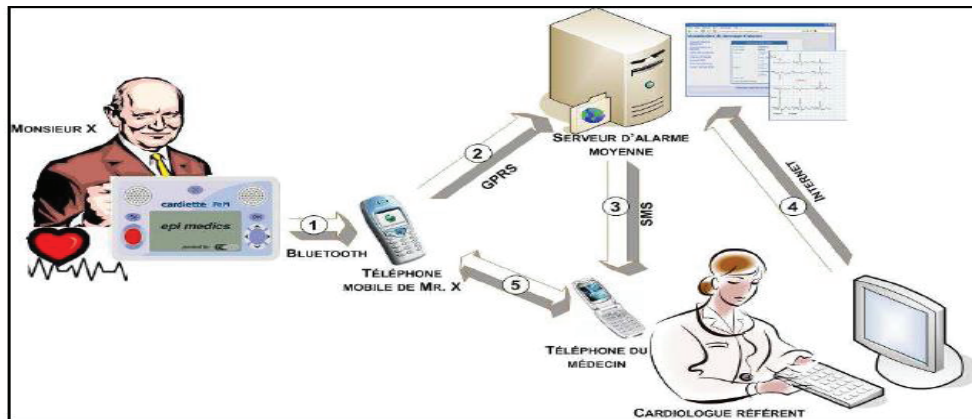


Figure 11. Scénario d'une alarme moyenne dans le cadre du projet EPI-MEDICS [2]

2.2.1.2 Scénario de télé-consultation du projet national Télé-infirmière

Dans une maison de retraite pour personnes âgées est installé un système de téléassistance médicale aux personnels infirmiers (Figure 12). Supposons qu'une infirmière A collecte des données cliniques du patient B (pression artérielle, ECG, taux d'oxygène) et détecte un problème. Elle veut se renseigner auprès d'un médecin généraliste C. Elle transmet l'ECG ainsi que les différents paramètres biomédicaux sur un serveur par l'intermédiaire d'un chariot médical. Le généraliste C consulte les données enregistrées en accédant au serveur depuis la station. Il décide à distance d'une conduite à tenir qui sera appliquée par l'infirmière.

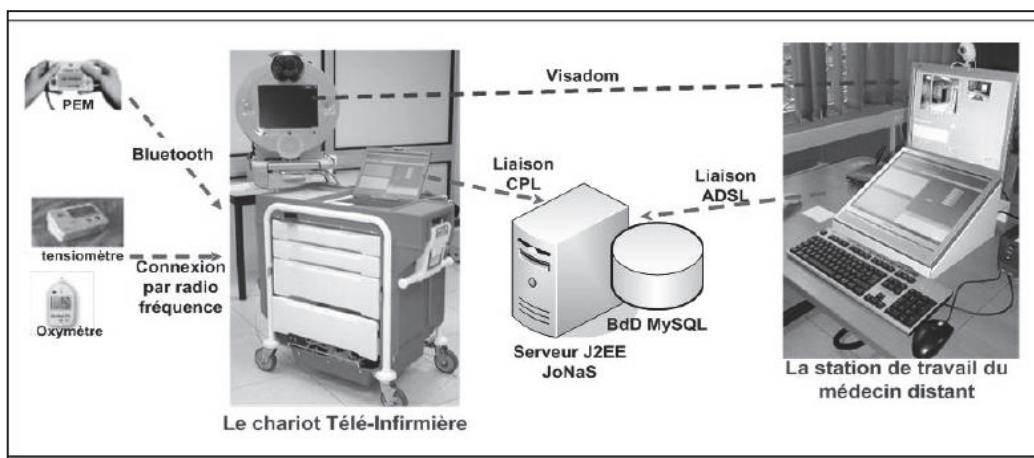


Figure 12. Architecture du système de télé-assistance médicale dans le cadre du projet Télé-infirmière [2]

2.2.2 Diversité des scénarios en e-santé

Dans le domaine de l'e-santé, les scénarios d'usage (télé-assistance à domicile, télé-consultation, télé-expertise, etc.) sont multiples. Un système qui gère l'acheminement des messages doit permettre l'adéquation à divers scénarios cliniques : multi-acteurs, multi-spécialités, etc. Dans chaque scénario, il y a différents types de messages échangés entre de multiples acteurs (Figure 13). Un message peut être une alarme. Il peut aussi faire l'objet d'un transfert de paramètres biomédicaux avec contrôle automatique de valeurs (suivi à domicile de patients dialysés, de maladies chroniques, etc.) ou sans contrôle (résultats d'analyse, compte rendu, etc.), d'une demande ponctuelle de rendez-vous, de prescription, d'assistance, etc. Une demande ponctuelle peut être d'avis médical (avis diagnostique, avis thérapeutique, avis de transfert), d'assistance technique ou de prescription, médicamenteuse, diététique. Une stratégie d'acheminement d'un message de télé-expertise peut ne pas être la même que celle d'un message de télétransmission d'informations par exemple. Pour le même message, plusieurs scénarios d'acheminement sont envisageables. L'un des principaux défis est de fournir une gestion générique du pilotage des messages qui s'applique aux différents scénarios d'usage et qui s'adapte aux différents acteurs.

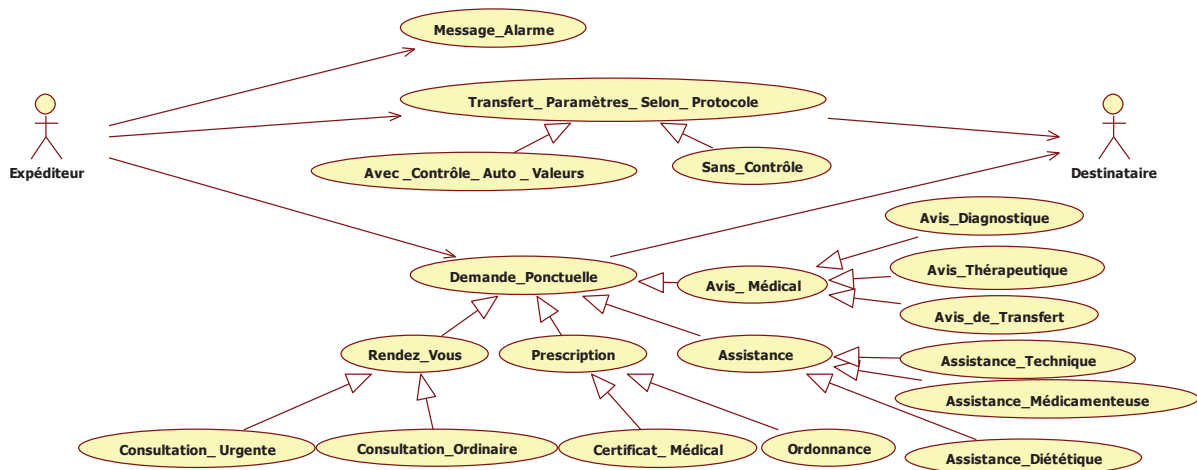


Figure 13. Diversité des scénarios en e-santé

2.2.3 Diversité des acteurs en e-santé

Les systèmes d'information en santé mettent en jeu plusieurs acteurs, professionnels de santé, hébergeurs de données médicales, assurances maladies, prestataires de soins, etc., et bien évidemment le patient qui représente l'acteur principal dans tout le processus médical. Les professionnels de santé sont des professionnels médicaux (médecin, chirurgien, etc.) et paramédicaux (aide-soignant(e), ambulancier(ère), infirmier(ère), etc.) dont le droit d'exercice et les actes sont réglementés. La numérisation des données médicales et l'émergence des technologies d'information de communication ont favorisé l'échange de données médicales informatisées entre différents types d'acteurs expéditeurs et destinataires des messages. Le diagramme de classe (Figure 14) illustre la diversité des acteurs en e-santé. L'acteur source ou bien l'expéditeur du message en e-santé peut être un patient, un aide-soignant, un secouriste, un pompier, un médecin de travail, un dispositif médical de surveillance en continu ou

actionné sur demande, un dispositif d'alerte, etc. Le destinataire peut être un médecin traitant, un spécialiste, un centre d'appel, un service d'urgence (SAMU), etc.

L'un des défis d'un système de gestion d'acheminement automatisé des messages réside désormais dans sa capacité de transmettre un message vers le « bon » destinataire et dans le « bon » délai, de garantir une meilleure adéquation de l'état du patient à sa prise en charge. Le défi n'est pas simple vu la multiplicité des systèmes d'information à prendre en compte et la diversité des acteurs impliqués. Les messages échangés entre les acteurs de santé quel que soit leurs raisons, pour une demande de prise en charge du patient, une demande de conseil auprès d'un professionnel de santé ou une demande d'expertise, etc., doivent être transmis de façon personnalisable en fonction de la situation des acteurs source et cible du message.

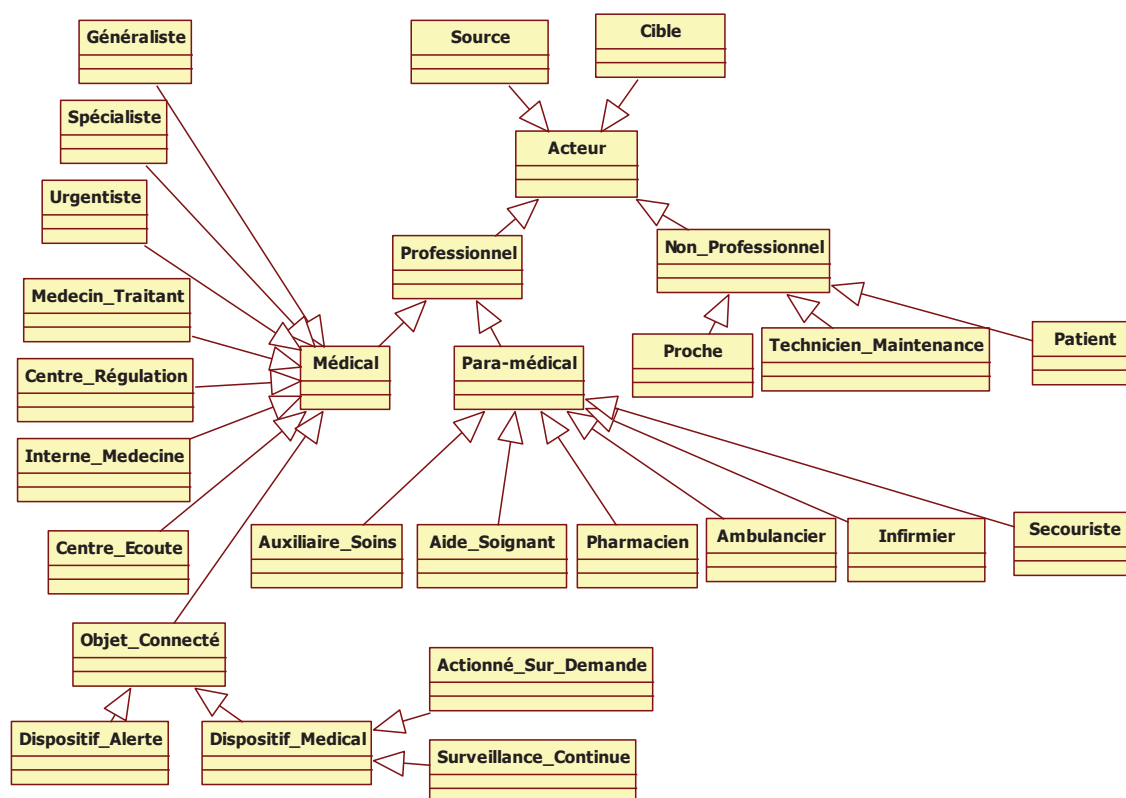


Figure 14. Modèle des acteurs en e-santé

En effet, la destination par exemple d'un message de prise en charge issu d'un patient qui vit tout seul et ayant un problème d'Alzheimer peut ne pas être la même que celle d'un patient qui vit en famille et ne souffre d'aucun handicap mental. Un message issu d'un patient ayant un problème d'Alzheimer peut être envoyé à la centrale d'écoute pour vérifier l'exactitude de la demande avant d'être transmis à son médecin traitant par exemple. Un message de prise en charge issu d'un patient se trouvant dans un environnement hostile (hautes montagnes par exemple) est supposé être envoyé à un fournisseur de soins possédant les ressources matérielles requises pour accéder à ce type d'environnement (hélicoptère par exemple).

2.2.4 Contraintes en environnement ubiquitaire

Les contraintes d'acheminement des messages en e-santé s'accroissent en environnement ubiquitaire, avec l'informatique pervasive. « *Pervasive* » est un terme anglais qui signifie omniprésent. En informatique, ce paradigme a été utilisé pour désigner un monde où l'information est disponible n'importe où et n'importe quand. Il a suscité différents axes de recherche plus au moins équivalents : les systèmes pervasifs, l'informatique ubiquitaire et l'intelligence ambiante [49] [50] [51].

Smartphones, tablettes, PDA et les technologies de réseaux sans fils bouleversent de plus en plus notre vie et deviennent une partie intégrante de notre quotidien. Avec l'essor des dispositifs miniaturisés communicants et intelligents, et de réseaux de communication sans fils, nous entrons dans l'ère de l'informatique ubiquitaire ou pervasive. L'e-santé n'échappe pas à cette évolution, plusieurs dispositifs portables et intelligents ont été développés et ont été utilisés dans différentes pratiques médicales: le suivi à domicile, la télé-expertise, la téléconsultation, etc. [52]. Ainsi, en e-santé, on assiste de plus en plus à un accroissement du nombre de messages transférés depuis et vers tout type d'environnement. Néanmoins, cela soulève plusieurs contraintes et problèmes liés notamment à la mobilité des utilisateurs, à l'hétérogénéité de l'information et des ressources, à l'évolution du contexte, etc. Un système de gestion d'acheminement des messages en e-santé doit prendre en considération toutes ces contraintes que nous détaillons ci-après.

2.2.4.1 C1 : Evolutivité du contexte

L'acheminement des messages en e-santé est très sensible au contexte à la fois de l'expéditeur et du destinataire du message. Ces contextes sont très dynamiques et changeants à tout moment, voire en temps réel au moment du transfert du message. Dès les années soixante, le terme contexte a été étudié et utilisé dans plusieurs travaux de recherches en informatique ubiquitaire en faisant l'objet de plusieurs définitions [53] [54]. Nous considérons le contexte comme étant un ensemble d'informations et de situations reliées à une entité et auxquelles une application peut être sensible, c'est-à-dire qu'elle peut changer son comportement en conséquence. Cette entité peut être un objet, une personne, un événement, etc. Une étude plus détaillée sur le concept « contexte » sera fournie dans le chapitre suivant de la thèse.

L'informatique ambiante impose la contrainte de la dynamique du contexte. Il est primordial, dans un système de gestion d'acheminement des messages de considérer cette dynamique et de savoir s'y adapter automatiquement avec la moindre intervention humaine. En e-santé, la prise en charge d'un patient par exemple en environnement ubiquitaire exige l'intervention et l'interaction de plusieurs entités à savoir le patient, un ou des professionnels de santé, des dispositifs personnels médicaux. Ces entités peuvent être n'importe où, dans un établissement hospitalier, à l'extérieur, etc. En plus du contexte géographique des entités, d'autres informations contextuelles peuvent évoluer n'importe quand. Illustrons la contrainte d'évolutivité du contexte dans le premier scénario. Le médecin traitant Y peut devenir indisponible après le transfert du message pour raison d'intervention dans un autre acte

médical. Des contraintes d'énergie ou une panne du dispositif médical du patient X peuvent également survenir. Le contexte médical du patient X à savoir ses signes cliniques, peuvent aussi évoluer dans le temps. Les douleurs thoraciques peuvent s'aggraver après la transmission du message et l'alarme moyenne devient ainsi une alarme majeure.

2.2.4.2 C2 : Mobilité des utilisateurs

La mobilité est l'une des propriétés qui caractérisent les systèmes pervasifs et est une contrainte que doit prendre en compte un système de gestion d'acheminement en e-santé. Les équipements et dispositifs médicaux peuvent être transportés par leurs utilisateurs, patients ou professionnels de santé ou autre. Ces acteurs peuvent changer de localisation géographique à tout moment, ce qui peut influencer le choix de l'itinéraire du message. Considérons le premier scénario. Vu le déclenchement d'une alarme moyenne, le déplacement du médecin traitant vers le patient X devient envisageable. Au cours du transfert du message d'alarme, le médecin traitant Y peut changer de localisation et s'éloigner de son patient X. Dans ce cas, le système qui gère l'acheminement des messages en e-santé doit tenir compte de la contrainte de mobilité et peut ainsi choisir une autre destination du message qui correspond mieux à la situation géographique et médicale du patient. La mise en place d'un tel système nécessite une capacité d'adaptation automatique en fonction de la mobilité non seulement du destinataire mais aussi de l'expéditeur du message.

2.2.4.3 C3 : Hétérogénéité des environnements

Les systèmes pervasifs en e-santé sont composés de plusieurs objets communicants hétérogènes aux niveaux matériels (dispositifs médicaux portables, PC, etc.), logiciels et protocoles de communication. Les équipements intègrent différentes technologies et fonctionnent avec divers systèmes d'exploitation. Les systèmes de santé sont très communicants mais hétérogènes. Chacun possède un jargon propre à sa discipline (radiologie, laboratoire d'analyse, soins infirmiers, etc.). Pour améliorer le parcours de soins, les systèmes doivent se comprendre et les échanges doivent se faire quelque soient les disparités culturelles et linguistiques des professionnels et des organisations. Dans le deuxième scénario par exemple, le message de téléconsultation transmis doit garder le même sens chez les deux interlocuteurs: Médecin généraliste C et l'infirmière A, de même qu'en télé-expertise.

2.2.4.4 C4 : Temps de réponse

Des décès peuvent survenir à cause de non réponse d'un médecin sur une demande de prise en charge urgente. Un système de pilotage d'acheminement des messages en e-santé doit pleinement tenir compte de la contrainte « temps de réponse ». En effet, chaque message et selon son degré de priorité a un délai limité pour son traitement. Le dépassement de ce délai peut répercuter une adaptation dans la stratégie d'acheminement. Le destinataire peut être indisponible à tout moment, comme il peut lire la demande et refuser de la traiter. Reprenons le scénario de télé-infirmière. Supposons que le généraliste C a reçu le message. Il l'a lu mais il n'a pas répondu à l'infirmière. L'infirmière a attendu 2 heures mais sans recevoir de réponse. Elle est ainsi obligée de relancer manuellement la demande vers un autre

professionnel de santé. Le problème d'un tel système est comme nous le constatons est de ne pas pouvoir fournir des redirections automatiques et intelligentes en fonction de la disponibilité du destinataire et en fonction du temps d'attente maximum requis par l'expéditeur de la demande. Les adresses des destinataires sont fixées et ne peuvent pas être changées selon des règles de décision.

2.3 Besoins requis en gestion des acheminements en e-santé

Dans cette section, nous allons répondre à la question suivante : De quelles propriétés un système de gestion d'acheminement des messages en e-santé doit-il être doté afin qu'il soit capable de surmonter les contraintes précédemment présentées ?

2.3.1 B1: Personnalisation

Devant la contrainte « diversité des acteurs en e-santé », un système de gestion d'acheminement doit être personnalisable, afin de délivrer le message au « bon » destinataire et dans le « bon » délai requis pour son traitement. Par exemple, un message de Télé-expertise de demande d'aide sur l'interprétation d'un ECG doit être transmis au professionnel de santé le plus approprié en termes de compétence, de profil professionnel, à un cardiologue par exemple. Un message d'alarme doit être envoyé au professionnel de santé le plus pertinent en termes de disponibilité et de l'historique de ses temps de réponses aux requêtes de prise en charge antérieures, etc. Le choix du destinataire dépend également de la situation médicale, géographique, sociale du patient ou plus généralement de l'expéditeur du message.

Ce problème de personnalisation se pose souvent en ingénierie des services métiers où les services doivent proposer la solution la plus adéquate au profil du consommateur et au contexte de la requête. Dans ce contexte, pour promouvoir des services personnalisés, [55] a proposé une méthode orientée objectifs. Cette approche s'appuie sur la capture des données contextuelles pour permettre de développer des services fortement adaptables et centrés sur les objectifs du consommateur.

2.3.2 B2: Sensibilité au contexte

Un système de gestion d'acheminement des messages doit être « sensible au contexte » afin de prendre en compte toute évolution en temps réel des contextes source et cible. Selon [56] [57], la sensibilité au contexte ou *context awareness* est la capacité d'un système à réagir en fonction du comportement de l'environnement où il se trouve. Ryan et al [58] définissent la sensibilité au contexte par l'aptitude de capturer, interpréter et répondre aux aspects de l'environnement local de l'utilisateur et du terminal. Ainsi, le défi qui se pose aux systèmes de gestion d'acheminement sensibles au contexte est triple : identifier le contexte pertinent auquel une prise de décision d'acheminement est sensible, surveiller le changement du contexte et s'adapter dynamiquement au contexte.

2.3.3 B3: Auto-Adaptation

Un système auto-adaptatif, est à notre sens, capable de modifier dynamiquement, en temps réel et sans intervention humaine, son comportement en fonction de changements contextuels dans le but de fournir un service plus approprié. Les décisions d'acheminement en e-santé sont, comme nous avons vu, très sensibles au contexte du message. De notre point de vue, un système qui gère l'acheminement des messages doit être adaptatif voire auto-adaptatif relativement à l'environnement dans lequel il est plongé et avec lesquels il interagit. Il doit être capable d'adapter automatiquement son processus décisionnel de routage des messages en fonction du contexte et sans aucune intervention humaine. L'adaptation du système peut se répercuter au niveau du processus décisionnel d'acheminement, des services à invoquer, des paramètres d'acheminement, etc. Des décisions de redirection des messages en e-santé peuvent ainsi être prises en temps réel.

2.3.4 B4: Interopérabilité

Un système qui gère l'acheminement des messages doit prendre en considération la contrainte hétérogénéité et préserver ainsi l'interopérabilité des systèmes d'information de santé. L'interopérabilité en matière de santé s'intéresse à l'échange de façon cohérente de documents cliniques (qui peuvent porter sur les aspects médicaux, administratifs, financiers et juridiques pour un hôpital) entre des systèmes hétérogènes d'information hospitalier. Selon Bouhaddou [59], l'interopérabilité peut être définie comme la capacité de deux ou plusieurs systèmes d'échanger des informations et de les utiliser. [60] a distingué deux perspectives d'utilisation de l'interopérabilité : l'une interne et l'autre externe. Le point de vue interne se produit au sein de la même organisation, entre les différents services au sein du même hôpital (service d'urgence, pharmacie, laboratoire, etc.) par exemple. L'interopérabilité en externe implique l'échange de données entre différentes organisations, par exemple entre hôpital et centrale d'écoute, entre le SAMU et un centre d'appel, etc. L'hétérogénéité est une réalité intrinsèque aux applications pervasives en e-santé. Cette hétérogénéité a fait de l'interopérabilité une nécessité pour garantir l'efficacité des systèmes de soins. Une des difficultés de la mise en œuvre d'un système de gestion d'acheminement des messages est de préserver l'interopérabilité entre les différents systèmes d'information de santé. Il est important qu'un tel système respecte les standards et les spécifications internationales connus pour l'échange de données numériques médicales, afin de permettre leur réutilisation, et de façon fiable.

2.3.5 B5: Supervision

Afin de surmonter la contrainte temps de réponse, un système de gestion d'acheminement de messages doit superviser l'état du message tout au long de son cycle de vie et adapter ses décisions en conséquence. Cette supervision va permettre ainsi de détecter les dépassements des délais requis pour le traitement du message chez chaque destinataire. Ceci peut optimiser le temps de traitement du message et améliorer ainsi la qualité de prise en charge du patient.

2.3.6 B6: Autonomie

Pour gérer la complexité et la diversité des scénarios en e-santé, un système de gestion d'acheminement des messages doit se doter d'une certaine autonomie lui permettant de se gérer et de s'adapter automatiquement aux changements contextuels. Un système autonome en informatique est un système qui est capable de s'autogérer, c'est-à-dire sans intervention extérieure de l'Homme [61]. Le fonctionnement des systèmes autonomes s'inspire largement de celui du système nerveux du corps humain, notamment en tendant de reproduire certains comportements qui lui permettent de se maintenir en bonne condition. Par exemple, le rythme cardiaque, la distribution de l'oxygène dans les muscles se régulent naturellement en fonction de l'effort subi par le corps humain. Ces ajustements se font de manière automatique. A l'instar du corps humain, les systèmes autonomes autogèrent plusieurs tâches sans être conduits par un opérateur humain et devraient à terme pouvoir offrir des propriétés telles que : l'auto-configuration, l'auto-optimisation, l'auto-protection, l'auto-guérison. Nous détaillerons ces diverses propriétés dans le chapitre suivant.

Dans notre contexte, en e-santé, la capacité humaine toute seule, n'est plus capable de gérer l'acheminement des messages vu la multiplicité, la diversité, la complexité des scénarios et la constante évolution du contexte à la fois de la source et de la cible du message. En effet, il est très difficile de prévoir tous les scénarios et les contextes possibles, et d'établir les stratégies d'acheminement correspondantes. Introduire des capacités d'autonomie dans un système d'acheminement de messages en e-santé rend la gestion de routage des messages plus automatisée et moins dépendante des opérateurs humains.

2.3.7 B7: Sécurité et protection de la vie privée

En e-santé, l'échange des données médicales doit être réalisé d'une façon sécurisée [62] [63]. Une définition des exigences en matière de sécurité des systèmes d'information de santé est nécessaire pour créer un espace de confiance favorable au partage et à l'échange des données de santé. Si le système de gestion de routage des messages en e-santé est adaptatif, il doit recueillir des données contextuelles des acteurs sources et cibles auxquelles il est sensible. Ces données peuvent être des données personnelles, sociales, médicales du patient, des données de localisation du patient ou de professionnels de santé, etc. La collecte de ces données doit être faite d'une manière sécurisée sans porter atteinte à la vie privée des acteurs impliqués dans le processus d'échange.

2.3.8 B8: Intelligence

Afin de répondre aux besoins liés à l'acheminement de messages en e-santé, une gestion plus intelligente de cet acheminement est nécessaire. Comme nous avons vu précédemment, il est important que la prise de décision d'acheminement soit réalisée en fonction du contexte. Or, il est difficile de prédire toutes les configurations possibles du contexte. L'utilisation des méthodes classiques à base des règles prédéfinies n'est pas donc envisageable. Il est difficile à la capacité humaine de prévoir toutes les situations d'adaptation possibles. Nous avons vu également que le « choix » du destinataire est ardu vu la multiplicité des acteurs et des

scénarios en e-santé. Pour ce faire, nous recourons à l'intelligence artificielle pour embarquer de l'intelligence dans le processus décisionnel d'acheminement. Notre objectif est ainsi de garantir une meilleure qualité de service, paradigme que nous traitons dans la section suivante.

2.3.9 B9: Qualité de Service

2.3.9.1 Définition

Il existe un standard X.902 défini par l'*International Telecommunication Union* [64] qui décrit la Qualité de Service (QoS) comme un ensemble d'exigences concernant le comportement collectif d'un ou de plusieurs. La QoS est souvent utilisée en réseau pour évaluer la performance des transmissions selon des critères qualitatifs tels que la connexion, le délai, la disponibilité, la gigue, le taux d'erreur, le taux de perte de paquets. Ces critères sont plutôt non fonctionnels et ne concernent pas l'évaluation de l'application elle-même. A présent et avec la multiplication des applications multimédia, il n'est plus possible d'évaluer la QoS des applications uniquement en termes de réseaux et de matériels [65] [66]. A cet égard, [67] définit la QoS comme étant l'ensemble des caractéristiques quantitatives et qualitatives d'un système multi-média distribué qui permet d'atteindre la fonctionnalité requise pour une application. Souil et al, dans leur article [68], confirment que les critères d'évaluation du concept QoS dans une application en e-santé sont beaucoup plus nombreux et exigeants que ceux dans un autre type d'application vu la sensibilité du domaine de l'e-santé. Selon lui, les critères de la QoS au sein d'un réseau de capteurs en e-santé sont : l'exactitude et la qualité des données collectées, la rapidité pour délivrer les données collectées dans le délai approprié, la fiabilité, la sensibilité au contexte, l'interopérabilité et la bande passante.

2.3.9.2 Qualité de service dans la gestion d'acheminement des messages

Alors que la plupart des travaux se concentrent sur les critères non fonctionnels de la qualité de service de l'acheminement des messages, nous souhaitons intervenir au niveau fonctionnel de l'application elle-même. Nous définissons la QoS du système de gestion d'acheminement des messages comme étant la capacité de fournir des services surmontant les contraintes présentées : Mobilité, changements de contexte source et cible, hétérogénéité et temps. Nous souhaitons garantir une meilleure gestion d'acheminement des messages en e-santé, sensible au contexte, autonome, intelligente, personnalisable, auto-adaptative et surtout générique qui s'applique aux différents scénarios en e-santé.

Notre point de vue est qu'un système de gestion d'acheminement des messages doit être automatisé, capable d'observer le contexte à la fois de l'expéditeur et du destinataire des messages et d'adapter ses décisions de routage en conséquence. Il doit transférer le message d'une façon personnalisable vers le bon destinataire et dans le délai requis eu égard à la situation médicale du patient, tout en tenant compte de diverses situations médicales, sociales, spatiotemporelles, etc. L'objectif reste de promouvoir le bien être des patients n'importe où et n'importe quand. La QoS devrait donc satisfaire l'ensemble des besoins que nous avons identifiés.

Résumé

Les scénarios d'acheminement des messages en e-santé sont multiples. Nous avons présenté dans cette partie la diversité des scénarios et des acteurs, illustrée par deux exemples, et mis en évidence des contraintes à surmonter par un système de gestion de routage. De chaque contrainte, nous avons déduit les besoins à remplir pour un meilleur acheminement des messages. Nous résumons l'ensemble des contraintes et les besoins correspondants dans le tableau suivant.

Contraintes	Besoins
- Diversité des scénarios en e-santé - Diversité des acteurs en e-santé	- Personnalisation
- Evolutivité du contexte - Mobilité des utilisateurs	- Sensibilité au contexte - Auto-adaptation - Intelligence - Autonomie - Sécurité
- Hétérogénéité des environnements	- Interopérabilité
- Temps	- Supervision - Intelligence - Autonomie

Tableau 1. Contraintes et besoins d'un système de gestion d'acheminement des messages en e-santé

Si nous avons distingué les besoins de la gestion d'acheminement des messages, cela ne signifie pas qu'ils sont indépendants. Au contraire, ces critères présentent un chevauchement. Dans la littérature, par exemple, l'approche sensibilité au contexte a été adoptée dans la conception des systèmes autonomes et auto-adaptatifs [69]. Un système autonome peut faire appel aux technologies de l'intelligence artificielle. Un système autonome et sensible au contexte peut être de plus sécurisé.

2.4 Travaux connexes. Avantages et Limites

L'objectif de cette section est de présenter une synthèse non exhaustive de la littérature et des projets relatifs à notre problématique de recherche : l'acheminement des messages. Nous commençons dans un premier temps par présenter les approches tous domaines. Puis, plus spécifiquement, nous présentons des approches adoptées en e-santé. Dans cette dernière approche, nous classons les travaux de recherche étudiés en rapport avec notre travail en trois volets : l'acheminement des messages en cas d'urgence, l'acheminement sensible au contexte des messages et l'acheminement intelligent. Parallèlement, nous mettons en lumière les avantages et les limites des travaux présentés. Un résumé sera fourni dans la dernière partie de cette section.

2.4.1 Travaux de recherche connexes

2.4.1.1 Approches tous domaines

Plusieurs travaux de recherche abordent la problématique de gestion d'acheminement des messages et du traitement des exceptions des flux de messages. [70] propose un modèle de contrôle prédictif et proactif qui permet de prévenir les violations et les ruptures possibles entre des applications communicantes. Cette étude a démontré la capacité de SLA (*Service Level agreement*) de promouvoir la qualité des services et d'améliorer la dynamique dans l'acheminement des messages en temps-réel. [71], quant à lui, a recouru au paradigme « sensibilité au contexte » pour résoudre les problèmes de routage dans un réseau local sans fils. Néanmoins, cette étude restreint le contexte à l'énergie consommée par les dispositifs mobiles du réseau. La même approche a été abordée par [72] qui propose un algorithme, basé sur celui des colonies de fourmis pour le routage adaptatif et sensible au contexte « énergie » dans un réseau de capteurs sans fils. D'autres travaux se sont plutôt appuyés sur les algorithmes évolutionnistes et plus particulièrement sur l'algorithme génétique pour optimiser la QoS que ce soit dans un réseau générique [73] ou bien dans un réseau particulier, comme un réseau de capteurs sans fils [74]. Ke et al [74] distinguent plusieurs types de protocoles d'acheminement : protocole basé localisation, protocole respectant la contrainte de latence, protocole d'acheminement hiérarchique qui minimise la consommation de l'énergie en classant les nœuds dans des clusters, etc. Dans ce même contexte, [75] a défini quelques critères techniques nécessaires pour l'assurance de la QoS dans un réseau de capteurs, tels que : la latence, la fiabilité, la bande passante, etc. [76] a développé le Framework ADOME pour la gestion des exceptions des flux de messages. Ce Framework est doté d'un gestionnaire des exceptions adaptatif et basé sur une interface web pour permettre la réutilisation des résolutions et leurs automatisations.

Certains aspects de l'approche basée sur l'algorithme de colonies de fourmis qui est abordée dans [72] pourraient être réutilisés dans notre problématique pour la recherche du chemin le plus court. Le principe de base de l'optimisation par colonies de fourmis (*Ant Colony Optimization /ACO*) est de modéliser le problème à résoudre sous la forme d'un chemin de coût minimal dans un graphe, et d'utiliser des fourmis artificielles pour rechercher les chemins optimaux. Le comportement des fourmis artificielles est inspiré de celui des fourmis réelles : (1) elles déposent des traces de phéromones sur les composants du graphe, (2) chaque fourmi choisit son chemin en fonction de probabilités qui dépendent des traces précédemment déposées, (3) ces traces sont progressivement diminuées par évaporation. Dans le cadre de notre problématique d'acheminement, nous pouvons imaginer les correspondances suivantes : La fourmi \rightarrow c'est le message, la source de nourriture (nœud du graphe) \rightarrow c'est l'acteur source, le nid des fourmis (nœud du graphe) \rightarrow c'est l'acteur cible. Ainsi, la probabilité du choix du chemin du message va dépendre de deux facteurs : la quantité de phéromones basée sur l'expérience passée que nous expliciterons ci-après et une fonction heuristique dont les paramètres peuvent être les caractéristiques du contexte (favorable ou non) telle que la distance géographique séparant l'expéditeur du destinataire, etc. Nous pouvons prendre avantage de cette approche pour permettre le choix du bon chemin du message, et donc satisfaire le besoin de « personnalisation ». En effet, la quantité de

phéromones déposés dans le chemin entre le patient et le professionnel de santé peut nous servir à déterminer un « niveau de confiance » pour le professionnel de santé en fonction des expériences passées, du nombre de fois de non acquittement, de la fréquence des dépassements du délai requis pour la réponse à une prise en charge. Toutefois, cette approche présente aussi des inconvénients vis-à-vis de notre problématique. En effet, la probabilité de transmission de la fourmi (message) d'un nœud (acteur) à un autre, se base fortement sur l'expérience passée (quantité de phéromones précédemment déposées). Or, ce n'est pas toujours le cas dans notre travail. En effet, si par exemple, le message est urgent, il sera directement transmis à un service d'urgence dont le contexte est souvent favorable, et donc sans avoir besoin de calculer des probabilités. En outre, dans l'approche des colonies de fourmis, une probabilité doit être calculée à chaque fois au niveau de chaque nœud avant de choisir le bon chemin à suivre par la fourmi (message). Or, dans notre cas, d'autres données contextuelles peuvent changer après le transfert du message, et donc, des possibilités de redirection et de recalcul des probabilités doivent être réalisées en temps réel, ce qui n'est pas fourni dans l'algorithme de colonies de fourmis.

2.4.1.2 Approches e-santé

2.4.1.2.1 Acheminement des messages en cas d'urgence

Dans la chaîne de processus de soins des patients, les demandes de prise en charge urgentes et critiques doivent être délivrées et traitées le plus rapidement possible. A cet effet, plusieurs travaux de recherches se sont intéressés à l'aspect routage des messages en cas d'urgence. [77] a proposé un système de gestion des alertes (AMS) basé sur les services web. Ce système fournit un ensemble de services permettant d'appliquer les stratégies d'acheminement correspondantes aux différentes situations d'urgence. L'architecture du système proposé comporte trois modules importants pour la gestion des alertes sortantes: (1) le module « Définition des stratégies d'urgence » spécifie les politiques à suivre si l'alerte n'a pas été acquittée dans le délai, (2) le module « *Matching role* » est responsable d'identifier les fournisseurs de services vers lesquels l'alerte sera transférée, (3) le module « Gestion de fournisseurs de service » est responsable d'appliquer les stratégies déjà définies dans le module « Définition des stratégies d'urgence ». Pour chaque réponse reçue, le module « Gestion de fournisseur de service » met à jour le statut de l'alerte, par exemple « alerte traitée ». Si le message d'alerte est envoyé à plusieurs fournisseurs de service, le premier service qui confirme la réception est assigné à la tâche et le reste des destinataires recevront un message d'annulation. Pour chaque type d'alerte, le module vérifie la table de la stratégie, exécute l'action associée et met à jour le statut du message d'alerte. La table de stratégie contient les niveaux de priorité des messages et les actions correspondantes. Un exemple de stratégie peut être le suivant : le message est par défaut une alerte urgente. Si la priorité devient égale à « très urgent » car il n'y a pas de réponse, le AMS crée un autre message d'alerte pour notifier le fournisseur du service sur la fin imminente du délai. S'il n'y a pas de réponse, le AMS essaie un autre service. En cas de non réponse, le niveau de priorité s'incrémente vers la valeur « très critique » et tous les fournisseurs de services recevront ainsi l'alerte.

Comme nous pouvons le constater, la contrainte temps a été fortement considérée dans ce travail. Des possibilités de redirection des messages ont été également déployées en cas d'indisponibilité d'un fournisseur de service. En outre, le système proposé présente l'avantage de satisfaire le besoin de « personnalisation » d'acheminement évoqué dans le chapitre précédent grâce au module « *Matching role* ». En effet, ce module s'appuie sur un algorithme qui permet de sélectionner le service le plus adéquat selon une fonction de coût (il peut se calculer en fonction du temps requis pour le service, la distance qui va être parcourue, les charges du service, etc.). Cet algorithme permet également le passage à l'échelle dans le niveau de priorité et dans les rôles des fournisseurs de services (demander à un spécialiste au lieu d'un généraliste) en cas d'indisponibilité des services. Néanmoins, cette étude reste restreinte et spécifique seulement au type de message d'alerte. En effet, un message peut être aussi de type informatif, technique, etc. Il peut faire l'objet de la transmission d'une demande de conseil sur la prise de médicaments, de réglage technique d'un dispositif médical, etc. Ces différents messages méritent également l'attention et leur acheminement doit être géré d'une manière prompte et efficace. En outre, les stratégies d'acheminement sont prédéfinies par un ensemble de règles fixées d'avance. Elles ne sont pas auto-adaptatives de façon dynamique en fonction du contexte.

Pour garantir une supervision sur l'acheminement des messages de santé urgente, Dilmaghani et Rao [78] [79] ont proposé une nouvelle approche pilotée par les événements et basée sur une modélisation mathématique : les réseaux de Petri coloriés. A cet effet, un superviseur a été développé pour évaluer la chaîne de soins des patients au sein d'un département hospitalier, réduire les dépendances et prévoir les impasses (par exemple si le nombre des messages est beaucoup plus important que le nombre de professionnels de santé disponibles).

L'avantage de cette approche est double. D'une part, elle s'appuie sur une modélisation qui prévoit à l'avance les problèmes de communication avant qu'ils apparaissent, et d'autre part elle permet la supervision de l'acheminement des réponses urgentes pour réduire les dépendances et prévenir les impasses. Nous pouvons prendre avantage de cette modélisation mathématique à savoir les réseaux de Petri, de par leur capacité de représenter l'aspect dynamique du système, un aspect très intéressant dans notre problématique. En effet, le statut du message est très dynamique et peut changer à tout moment et en temps réel. Nous pouvons utiliser les réseaux de Petri pour modéliser l'évolution des statuts des messages tout au long de leurs cycles de vie. Ceci va nous permettre par la suite de superviser l'acheminement des différents messages, de détecter les impasses si elles existent et de pouvoir ainsi prendre les bonnes décisions de routage. Toutefois, l'approche proposée ne modélise pas la dimension temporelle dont on nous avons besoin pour détecter, par exemple, les non traitements des messages dans le temps. Les réseaux de Petri temporels ou temporisés peuvent être une piste très prometteuse à explorer dans le chapitre suivant puisqu'ils permettent de modéliser les systèmes dynamiques à contraintes temporelles. De surcroît, l'approche adoptée n'étudie pas un concept très important dans l'acheminement qui est la personnalisation. En cas de détection d'une impasse de communication par exemple, le patient ne doit pas être redirigé

vers n'importe quel médecin disponible mais vers le plus adéquat à la fois aux contextes du patient et du médecin.

2.4.1.2.2 Acheminement sensible au contexte des messages

De nombreuses recherches ont abordé la problématique d'acheminement des flux médicaux en e-santé en tenant compte de l'aspect sensibilité aux différents types de contexte. [80] a proposé un cadre pour modéliser les flux médicaux et mettre en lumière leurs sensibilités aux allocations des ressources matérielles (équipement, médecin, infirmier, lits, etc.). La même approche a été adoptée par [81] mais en intégrant l'aspect modélisation des réseaux de Petri colorés. D'autres travaux se sont plutôt focalisés sur le contexte mobilité des utilisateurs [82] [83] [84]. [82] a proposé un système d'informations médicales sensible au contexte pour accéder aux ressources telles que le dossier médical du patient. Ce système est basé sur l'utilisation des réseaux de Neurones pour la localisation du médecin destinataire de la demande de prise en charge. Le logiciel intelligent de l'hôpital est un prototype proposé à l'université à Cambridge [83]. Ce prototype permet la localisation du personnel médical et l'initiation de conférences audio-video à partir du point le plus proche. Les cliniciens sont localisés et notifiés à la réception d'un appel et peuvent acquitter cet appel via leur badge. Stanford [84] a proposé un système de badge communicateur appelé Vocera, pour les utilisateurs mobiles. C'est un badge portable avec un bouton d'appel, un écran, reconnaissance vocale, etc. Il délivre les informations directement aux utilisateurs et les évitent d'aller vers un dispositif distant tels qu'un PC ou un téléphone.

Les travaux précédemment présentés étudient le contexte sous une vision unidimensionnelle restreinte aux critères : mobilité et ressources. Toutefois, la problématique d'acheminement des messages doit étudier le côté multidimensionnel que dégage le contexte. En effet, le pilotage du routage des messages en e-santé doit tenir compte de plusieurs types d'informations contextuelles auxquels il peut être sensible et en fonction desquels il peut adapter son comportement. Par ailleurs, une stratégie d'acheminement des messages dépend non seulement de la disponibilité des ressources matérielles et humaines du côté cible mais aussi du contexte géographique, climatique, social, etc., à la fois de l'expéditeur et du destinataire.

2.4.1.2.3 Acheminement intelligent des messages

Afin d'améliorer la qualité des services dans des applications ubiquitaires en e-santé, [85] a proposé un cadre ONOF-PAS (*ONtology Oriented Framework for Pervasive Applications and Services*). Ce cadre, bien qu'il ne s'intéresse pas vraiment à la problématique d'acheminement des messages, il fournit des services qui répondent bien aux deux caractéristiques requises pour la gestion de l'acheminement : intelligence et personnalisation. En effet, ONOF-PAS s'appuie sur un raisonnement à base des règles pour inférer les ressources disponibles et les plus adéquates aux tâches cliniques tout en tenant compte des conditions médicales des patients.

L'un des avantages de ce travail se décèle dans le fait qu'il prend pleinement en compte la contrainte de diversité des scénarios en e-santé. Il met en relief l'importance et des modèles

ontologiques pour raisonner en fonction du contexte et relier chaque tâche en e-santé avec les ressources disponibles (lits, bloc opératoire, etc.) et les plus adéquates aux situations du patient. Ces aspects-là sont très importants et peuvent être exploités dans notre problématique de gestion d'acheminement des messages. Toutefois, selon [86], l'échange des données médicales et des messages d'urgence en e-santé doivent tenir compte non seulement de la disponibilité des ressources matérielles logistiques mais aussi des compétences des professionnels de santé et de leurs connaissances relatives au patient.

[87] a proposé une nouvelle architecture pour l'échange interopérable et sensible au contexte des messages en e-santé. Cette architecture est basée sur les systèmes intelligents et est conforme au standard OpenEHR [88], un standard pour l'échange des dossiers de santé électroniques. L'architecture proposée comporte un package nommé *CAManager* qui gère l'échange des messages sensibles au contexte. Le *CAManager* inclut le package « *ContextManager* » pour traiter la dynamique du contexte (par exemple la localisation de l'utilisateur) et autres types de contexte (par exemple l'identité et les rôles des utilisateurs). Le *ContextManager* interagit avec le package *Adapter* pour effectuer des adaptations dans le contenu médical du message.

L'avantage de cette architecture est qu'elle s'appuie sur la séparation logique entre les données, la présentation et les traitements, ce qui favorise sa réutilisation. Sa bonne structuration s'explique par le fait qu'elle repose sur le modèle MVC (Modèle, vue, contrôleur), un patron de conception d'architectures logicielles [89]. En outre, l'architecture d'échange de données médicales proposée présente l'avantage de garantir l'interopérabilité vu sa compatibilité avec le standard openEHR. Ce standard représente une originalité dans le sens qu'il propose une approche qui dissocie le modèle d'information en deux : le modèle de référence et les archétypes. Le modèle de référence est l'équivalent de HL7 RIM. Les archétypes sont une modélisation du concept du domaine qui utilise des contraintes sur les classes du modèle de référence (Exemple : restreindre la classe « Observation » du modèle de référence à l'archétype « pression artérielle »). Néanmoins, selon [90], openEHR présente moins de cohérence sur le plan formalisme que HL7 : d'une part parce que son modèle de référence est beaucoup moins riche que le RIM et d'autre part, parce qu'il ne permet pas la modélisation complète d'un domaine. En plus, HL7 demeure le standard international le plus connu et le plus utilisé en matière d'échanges informatisés de données cliniques.

2.4.2 Projets de développement connexes

2.4.2.1 Projet MSSanté

Récemment, l'ASIP santé a mis au point un système de messagerie sécurisée en santé, appelé MSSanté. Cependant, ce système est réservé aux échanges entre seulement les professionnels de santé. Il leur permet d'échanger entre eux et par mail des données médicales à caractère personnel en toute sécurité. Les messageries MSSanté possèdent des annuaires communs et certifiés de l'ensemble des professionnels de santé. L'objectif de MSSanté est triple : Simplifier la pratique médicale et gagner du temps, assurer la sécurité des échanges pour garantir le respect des droits du patient, fournir un espace de confiance pour échanger

entre professionnels de santé. Un premier service de MSSsanté a été mis en place en 2013 et est disponible sur [91]. Mais, il ne s'agit que d'un système de mail classique, dépourvu de stratégies autonomes et intelligentes de communication.

2.4.2.2 Projet ACM

Dans son document « *IHE Patient Care Device Technical Framework* » délivré récemment en octobre 2015 [92], l'IHE a présenté le profil d'intégration de la gestion de la communication des alertes, connu sous l'acronyme ACM « *Alert Communication Management* ». Un profil d'intégration IHE décrit les échanges en fonction des besoins cliniques ainsi que les acteurs et les transactions IHE. Un acteur IHE peut être une application, un système d'information qui gère ou traite des informations. Les transactions IHE sont les échanges effectués entre les acteurs IHE, comme par exemple l'échange des données médicales du patient, utilisant les standards tels que DICOM et HL7. L'ACM définit la communication des alertes (alarmes physiologiques, techniques, conseils consultatifs) du dispositif médical du patient « *Alarm Reporter* » (AR) vers le système de gestion des alarmes (AM) et de l'AM vers le *Alarm Communicator* (AC) qui va délivrer le message vers le dispositif du destinataire (Figure 15).

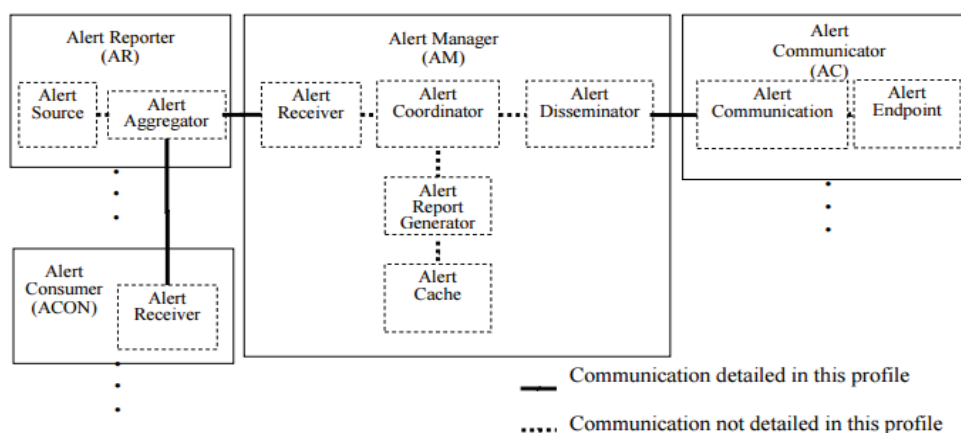


Figure 15. Le diagramme des acteurs du profil ACM [93]

L'objectif de l'ACM est d'émettre les « bonnes » alarmes avec la « bonne » priorité aux « bons » individus via des équipements avec le « bon » contenu [93]. Toutefois, le profil ACM demeure un système spécifique à un cas particulier qui est l'échange des messages de type alerte. En outre, il s'intéresse plus spécifiquement aux messages de l'ancienne version HL7 version 2. Cette version, comme nous l'avons vu dans le chapitre 1, ne repose pas sur un modèle d'information implicite, comme le RIM dans HL7 version 3. Pour faciliter l'interopérabilité des systèmes, le profil ACM a intégré d'autres segments spécifiques aux messages d'alertes dans la structure de messages HL7 version 2 qui peuvent influencer l'acheminement des messages d'alerte tels que : la localisation en temps réel du patient, l'état de l'alarme (activée, désactivée, verrouillée, etc.). Cependant, d'autres éléments du contexte importants pour définir une stratégie de routage des messages n'ont pas été considérés dans la conception de l'ACM, comme par exemple : le niveau de confiance du dispositif émetteur du message d'alerte (calculé en fonction du nombre de fausses alarmes par exemple, la validité

mentale de la personne auteur de l'alerte si l'alerte n'est pas automatique, le contexte climatique du patient ou du médecin surtout s'il y a une possibilité de transfert du patient, etc. Les scénarios de cas d'utilisation type qui sont décrits sont par exemple : 1) l'appel du personnel infirmier pour obtenir un coussin, 2) une alarme générée par un équipement au personnel infirmier, 3) les scénarios 1) et 2) dans le cas où l'alerte est supprimée à la source ou côté cible, ou 4) des alarmes techniques (niveau de batterie insuffisant, etc.).

2.4.2.3 Projet régional de télé-expertise : Expert-Assist

Le projet Expert-Assist a été piloté par notre équipe de recherche [2]. Le principe d'Expert-Assist consiste à automatiser la régulation des messages de télé-expertise en remplaçant les centres d'appel dont le coût est relativement important par un serveur capable de traiter l'information, de contrôler le transfert des messages et d'alerter les acteurs appropriés du système. Le serveur proposé garantit la redirection des demandes d'expertise vers l'expert disponible et compétent pour les prendre en charge dans des délais raisonnables. A cet effet, une architecture de type client/serveur a été conçue (Figure 16) : Les clients envoient et reçoivent les messages de télé-expertise via une couche HTTPS et le serveur permet la gestion d'acheminement de ces messages. Le serveur est composé d'un serveur web et d'un serveur d'application. Le serveur web exécute des scripts PHP pour : (1) créer, modifier et supprimer les paramètres des utilisateurs du système, (2) établir les règles d'acheminement capables de rediriger les messages vers les ressources humaines et matérielles disponibles, (3) créer et gérer les dictionnaires de données propre à chaque utilisateur. Le serveur d'application traite l'information et gère l'acheminement des messages de télé-expertise.

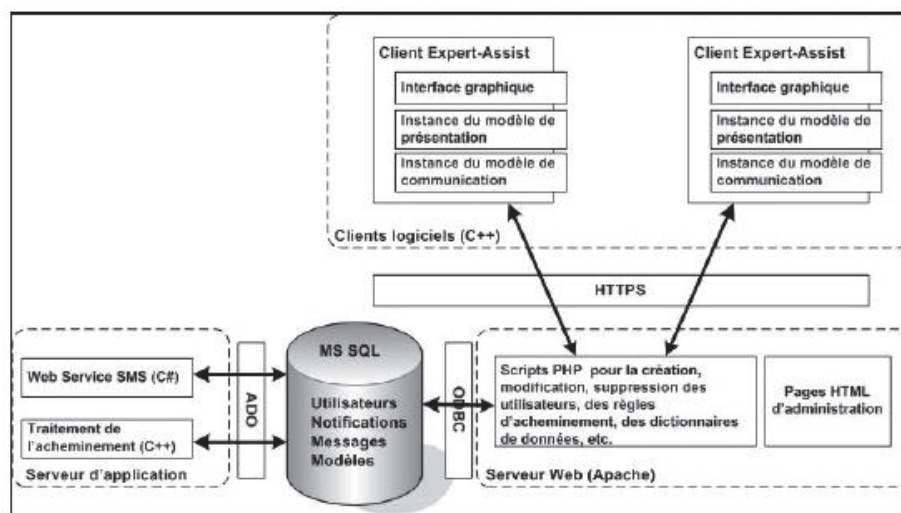


Figure 16. Architecture logicielle du système Expert-Assist [2]

Cette architecture s'appuie sur deux modèles : un modèle de représentation du message et un modèle de communication pour le pilotage et l'acheminement des messages. Le modèle de représentation du message permet à chaque utilisateur, acteur source ou cible, d'avoir ses propres modèles d'interface d'écriture du message et de composition de son message. Chaque message est assimilé à une macro-donnée véhiculant au format XML non seulement le

contenu du message mais également le modèle de représentation de l'expéditeur du message. Ainsi, chaque message sera présenté sur l'ordinateur du destinataire selon le même modèle d'interface que celui de l'expéditeur, afin d'éviter des erreurs d'interprétation, et ceci quelques soit les expéditeurs. Le modèle de réponse spécifique au destinataire sera fusionné à celui de chaque expéditeur tout en préservant l'intégrité du modèle de ce dernier. Le modèle de communication décrit les échanges entre les différents acteurs en fonction des priorités établies par ces mêmes acteurs. A la réception d'un message de télé-expertise, le serveur de traitement analyse les paramètres de communication (expéditeur, destinataire et niveau de priorité (demande courante, demande non urgente, réponse dans la journée, réponse rapide, réponse urgente). Selon des règles d'acheminement, il établit sa stratégie de routage. Ces règles intègrent trois paramètres importants : délai, priorité et moyen d'acheminement (SMS, Beeper, etc.). Quatre règles ont été conçus : (1) Emettre un avertissement via un beeper, un téléphone ou par SMS au destinataire ; (2) Transférer le message vers un autre destinataire issu du même profil ; (3) Transférer le message à un département hospitalier approprié ; (4) Envoyer le message à un service d'urgence. Chaque notification de télé-expertise a une durée de vie limitée. Si le destinataire est disponible, il peut accepter ou non la demande. S'il n'est pas disponible ou s'il refuse la demande, le serveur de traitement applique la règle suivante. Le paramétrage des règles d'acheminement est fourni par l'administrateur et se réalise via l'interface graphique suivante (Figure 17).

	routine	non urgent	in the day	quick answer	urgent
SMS	après : 1 semaine ordre : 2	1 semaine 2	1 semaine 2	1 semaine 2	1 semaine 2
Beeper	après : 1 minute ordre : 1	1 minute 1	1 minute 1	1 minute 1	1 minute 1
Secondary Doctor	after : 1 semaine order : 3 contact : Chevalier Philip	1 semaine 3 Chevalier Philip	1 semaine 3 Chevalier Philip	1 semaine 3 Chevalier Philip	1 semaine 3 Chevalier Philip
Hospital Service	after : 1 semaine order : 4 contact : Chevalier Philip	1 semaine 4 Chevalier Philip	1 semaine 4 Chevalier Philip	1 semaine 4 Chevalier Philip	1 semaine 4 Chevalier Philip
Emergency Centre	after : 1 semaine ordre : 5 contact : Chevalier Philip	1 semaine 5 Chevalier Philip	1 semaine 5 Chevalier Philip	1 semaine 5 Chevalier Philip	1 semaine 5 Chevalier Philip

Figure 17. Interface pour le paramétrage des règles d'acheminement [2]

L'avantage de cette architecture est qu'elle repose sur l'aspect modélisation (modèle de communication et modèle de représentation du message) qui fournit une personnalisation de l'acheminement des messages. Chaque utilisateur possède un modèle de représentation des données qui lui est propre. En outre, le facteur temps (délai de prise en charge de la demande) a été bien pris en compte dans la conception de l'architecture Expert-Assist. Nous avons vu qu'à chaque utilisateur est associé un délai requis de réponse. Néanmoins, l'aspect

intelligence et autonomie sont absents pour la gestion de l'acheminement. En effet, l'intervention humaine est toujours obligatoire pour définir les règles. De notre point de vue, les délais du traitement des messages doivent être adaptés d'une façon automatique et dynamique et non seulement en fonction des préférences du médecin responsable mais aussi en fonction du contexte à l'instant t de l'acteur cible et du message lui-même. Le délai de traitement du message peut être réduit automatiquement sans une intervention humaine si par exemple le contexte médical du patient s'aggrave ou si le message a été déjà redirigé plusieurs fois vers d'autres destinations ou si l'expert lit le message mais refuse de le traiter. Dans Expert-Assist, le choix du destinataire se fait par l'acteur expéditeur du message. Toutefois, outre les préférences d'acheminement de l'acteur source, un système de routage automatisé des messages devrait également fournir des suggestions de destinations qui correspondent bien à la fois au contexte médical du patient, à son contexte social, géographique, etc.

2.4.3 Synthèse des travaux connexes

Bien que les différents travaux étudiés aient abordé la problématique d'acheminement des messages en e-santé sous différents axes de recherche, aucune des solutions proposées ne couvre l'intégralité des besoins identifiés dans la section 2.3 : personnalisation, sensibilité au contexte, auto-adaptation, interopérabilité, supervision, autonomie, sécurité et intelligence.

Le tableau suivant rassemble les différents travaux détaillés précédemment et met en évidence leurs avantages et leurs limitations compte tenu des besoins qui sont importants pour le pilotage de l'acheminement des messages en e-santé.

Travaux de recherches	Besoins d'un système de gestion d'acheminement en e-santé							
	Personnalisation	Sensibilité au contexte	Auto-adaptation	Interopérabilité	Supervision	Autonomie	Sécurité	IA
[91]							×	
[2]	×	×		×	×			
[92]	×			×	×		×	
[78]-[79]		×			×			
[77]		×	×		×			
[85]-[86]	×	×						×
[94]		×	×	×			×	×
[81]		×	×					

Tableau 2. Tableau comparatif des recherches étudiées

2.5 Conclusion

La conception d'un système de gestion d'acheminement des messages en e-santé est une problématique qui a suscité beaucoup d'intérêt au cours des dernières décennies et beaucoup de méthodes ont vu le jour. Avec l'émergence de l'informatique ambiante et des environnements ubiquitaires, les défis de concevoir un tel système deviennent de plus en plus importants. Dans ce chapitre, nous avons mis en lumière notre problématique de recherche.

Nous avons déduit les différentes contraintes à surmonter pour concevoir un système de gestion d'acheminement des messages en e-santé. Pour illustrer ces contraintes, nous avons choisi de présenter deux scénarios issus de deux projets réalisés dans notre laboratoire. Ceci nous a permis par la suite d'identifier les besoins que doivent satisfaire le système. A l'instar de ces besoins, notre problématique se situe à l'intersection de plusieurs disciplines : Les systèmes autonomes, la sensibilité au contexte en environnement ubiquitaire, l'intelligence artificielle. Ainsi nous orientons la partie suivante de la thèse vers l'étude de quelques aspects dans ces disciplines en rapport avec notre problématique de recherche.

Partie II
**Modélisation des systèmes dynamiques,
intelligents, autonomes et sensibles au
contexte**

La seconde partie de ce mémoire présente un survol de l'état de l'art sur les systèmes intelligents, autonomes et sensibles au contexte. Dans le chapitre 3, nous cherchons à apporter des réponses aux interrogations théoriques suivantes : Qu'entendons-nous par systèmes autonomes, systèmes multi-agents et sensibles au contexte ? Quel est leur apport ? Le chapitre 4 s'intéresse à l'aspect modélisation des systèmes dynamiques et à l'étude de quelques méthodes d'intelligence artificielle relatives à notre problématique de recherche. Cette vue synthétique et non exhaustive des méthodes va nous permettre par la suite de faire un choix méthodologique pour l'implémentation de notre système. Nous allons explorer l'ère probabiliste à travers les réseaux bayésiens classiques et dynamiques, l'ère méta-heuristique à travers l'algorithme des colonies de fourmis et l'ère du raisonnement déterministe via les systèmes experts et les arbres de décision. La modélisation des systèmes dynamiques est un concept, comme nous le verrons, très important, dans notre problématique d'acheminement des messages. Nous présentons les réseaux de Petri en tant qu'outil graphique et mathématique de modélisation du comportement dynamique des systèmes.

Chapitre 3

Les systèmes autonomes et sensibles au contexte

3.1 Introduction

L'objectif principal de ce chapitre est de présenter une synthèse non exhaustive de la littérature et des articles relatifs à trois concepts afférents à notre travail : l'informatique autonome (*autonomic computing*), les systèmes multi-agents et la sensibilité au contexte. Nous présentons dans la première section de ce chapitre les systèmes autonomes, leurs caractéristiques et leur architecture telle qu'elle est définie par leur fondateur IBM. Nous explorons par la suite les systèmes multi-agents et ses différents domaines d'application. Nous étudions enfin les systèmes sensibles au contexte selon trois axes : un premier axe concerne la gestion du contexte et le développement des intergiciels sensibles au contexte, un second axe étudie la modélisation du contexte. Dans les travaux de recherche sur les intergiciels, nous distinguons deux approches : l'approche ingénierie dirigée par les modèles (IDM) et l'approche ontologie. Nous discutons ensuite les points forts et les limites des travaux présentés. La troisième partie de ce chapitre sera consacrée à exposer l'apport des systèmes autonomes aux systèmes sensibles au contexte et des systèmes multi-agents. Nous terminons ce chapitre par une conclusion.

3.2 Les systèmes autonomes

3.2.1 Définition des systèmes autonomes

Initié par IBM *Research* en 2001, les systèmes autonomes sont inspirés du système nerveux autonome du corps humain, pour permettre aux systèmes de s'autogérer, de s'auto-adapter aux changements imprévus et de gérer la complexité des applications informatiques [95], [96]. Après IBM, plusieurs autres industries telles que HP, Microsoft, Sun ont également opté pour la recherche dans ce nouveau domaine. Un système autonome doit être capable de se connaître lui-même, de prendre ses propres décisions, de contrôler le fonctionnement de ses applications et d'adapter automatiquement son comportement, sans intervention humaine, de la même façon que le système nerveux humain qui gère de nombreuses fonctions corporelles sans contrôle volontaire. Les systèmes autonomes constituent un des fondements essentiels des systèmes pervasifs. En effet, ils fournissent des mécanismes d'adaptation aux changements et à la dynamique en continu nécessaires en informatique pervasive [97].

3.2.2 Caractéristiques des systèmes autonomes

IBM définit quatre propriétés fondamentales des systèmes autonomes :

- Auto-configuration : C'est la capacité du système de se configurer dynamiquement en fonction du changement de l'environnement. Par exemple, à l'issue de l'ajout ou de la suppression d'un nouveau composant, le système autonome doit adapter son comportement en conséquence et modifier les paramètres de configuration sans intervention humaine.
- Auto-guérison : Le système observe l'environnement d'exécution et les logiciels qu'il administre, détecte automatiquement des problèmes de pannes, s'ils existent et se corrige lui-même.
- Auto-optimisation : Le système optimise l'utilisation des ressources.
- Auto-protection : Le système est capable de détecter automatiquement des attaques malveillantes et de s'en auto-protéger proactivement via des technologies comme SSL (Secure socket layer) ou de cryptage des données, etc.

D'autres propriétés des systèmes autonomes ont été évoquées dans la littérature tels que: auto-adaptation, auto-diagnostic, auto-organisation, auto-planification, auto-apprentissage, auto-évolution, auto-correction [98], [99], [100]. Les systèmes autonomes doivent faire face à plusieurs défis qui ont débouché sur de nouvelles perspectives de recherche. Parmi ces défis, on trouve la détermination et la spécification du problème sur lequel les systèmes autonomes doivent raisonner et planifier leurs décisions, la conception, l'implémentation, la gestion des politiques d'action, le test, la post-implémentation et la maintenance [101]. D'autres problèmes à résoudre par les systèmes autonomes tels que leur robustesse et la théorie d'apprentissage ont été soulignés par [98], [102].

3.2.3 Architecture des systèmes autonomes

Les systèmes autonomes sont fondés sur des agents logiciels autonomes appelés « éléments autonomes » [103]. Comme indiqué dans la Figure 18, un « élément autonome » est composé de deux parties : l'« élément géré » et le « gestionnaire autonome ». L'« élément géré » constitue la ressource contrôlée par le « gestionnaire autonome ». Il peut être une base de données, une unité CPU, un logiciel, une imprimante. Le « gestionnaire autonome » observe son environnement externe, contrôle l'élément géré, reçoit des informations au moyen de capteurs, les analyse et prend des plans d'actions. Un concept très important sur lequel se base le « gestionnaire autonome » et qui permet d'autogérer les composants logiciels et matériels, est la « boucle de contrôle ». Le « gestionnaire autonome » implémente la « boucle d'autonomie » (ou *control loop* en anglais) sur quatre niveaux ou parties [101] décrites ci-dessous:

- *Monitor* : Il fournit des mécanismes pour collecter, agréger, gérer et stocker des informations sur les « éléments gérés » dans une base de connaissance.

- *Analyse* : Il permet au « gestionnaire autonome » d’analyser les données déjà collectées par le « monitor », d’apprendre à partir de ses expériences et de prédire des futures situations.
- *Plan* : Selon les données analysées et en se basant sur des politiques, il planifie des actions pour atteindre certains objectifs. Un « gestionnaire autonome » d’un téléphone mobile peut par exemple baisser sa luminosité pour optimiser la consommation de la batterie selon certaines politiques. Ces politiques peuvent être définies par un administrateur et stockées comme connaissances pour guider le « gestionnaire autonome » dans sa prise de décision.
- *Execute* : Il exécute les actions recommandées sur l’élément géré.

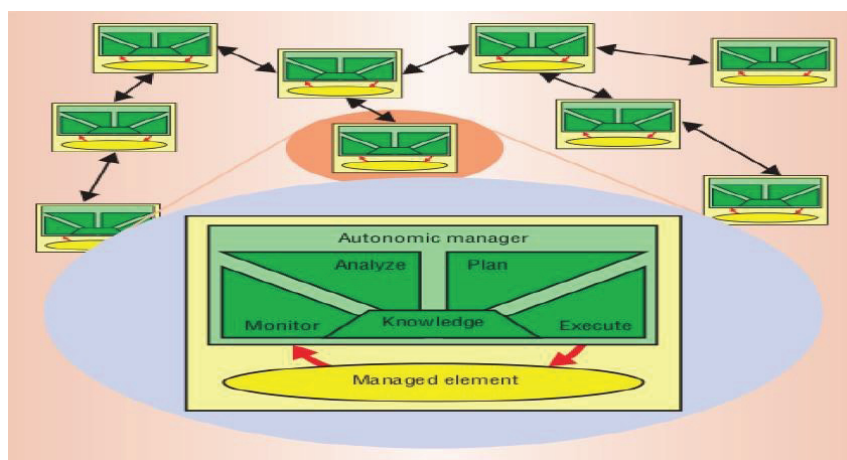


Figure 18. Architecture de systèmes autonomes [101]

3.3 Les systèmes multi-agents

L’approche multi-agents constitue aujourd’hui une technologie intéressante en pleine effervescence pour la conception et la mise en œuvre des systèmes coopératifs, autonomes et distribués [104]. Un système multi-agents (SMA) est constitué d’un ensemble d’agents dotés d’autonomie, de capacités d’adaptation et d’apprentissage qui partagent des ressources communes et qui communiquent entre eux, via des langages de communication entre agents (*Agent Communication Language*, ACL), comme KQML [105].

3.3.1 Quelques définitions

D’abord qu’est-ce qu’un agent ?

Définition 1. « Un agent est une "entité computationnelle", comme un programme informatique ou un robot, qui peut être vue comme percevant et agissant de façon autonome sur son environnement » [106].

Définition 2. « Un agent est une entité physique ou virtuelle qui est : (1) capable d’agir dans un environnement, (2) peut communiquer directement avec d’autres agents, (3) mue par

un ensemble de tendances (sous la forme d'objectifs individuels ou d'une fonction de satisfaction, voire de survie, qu'elle cherche à optimiser), (4) possède des ressources propres, (5) capable de percevoir (mais de manière limitée) son environnement, (6) ne dispose que d'une représentation partielle de cet environnement (et éventuellement aucune), (7) possède des compétences et offre des services, (8) peut éventuellement se reproduire, (9) dont le comportement tend à satisfaire ses objectifs, en tenant compte des ressources et des compétences dont elle dispose, et en fonction de sa perception, de ses représentations et des communications qu'elle reçoit » [104].

A partir de ces deux définitions, nous pouvons déduire qu'un agent est une entité de base d'un système qui interagit avec son environnement d'une façon autonome et adaptative. On parle d'autonomie d'un agent : 1) parce qu'il est doté d'un aspect réflexif, c'est-à-dire que chaque agent dispose de son propre mécanisme de prise de décision, de son propre but, et 2) parce que son comportement dépend au moins partiellement de son expérience. L'adaptabilité signifie qu'un agent est capable de changer son comportement et le cours de son exécution, et de mettre à jour ses connaissances par apprentissage en cas d'évolution de son environnement.

Les agents peuvent être *cognitifs* disposant d'un aspect de réflexion et de raisonnement, *réactifs*, c'est-à-dire dépourvu d'intelligence et réagissant par stimulus-réponse à l'état courant de l'environnement, et/ou *intentionnels* [107]. Il existe aussi les agents qui coopèrent d'une façon collective pour résoudre un problème complexe, ce sont les agents coopératifs. Dans d'autres cas, les agents sont en compétition, on parle alors d'agents compétitifs.

Qu'est-ce qu'un système multi-agents ?

Définition. « Un système multi-agent (ou SMA) est un système composé des éléments suivants : (1) Un environnement E , c'est-à-dire un espace disposant généralement d'une métrique. (2) Un ensemble d'objets O . Ces objets sont situés, c'est-à-dire que, pour tout objet, il est possible, à un moment donné, d'associer une position dans E . Ces objets sont passifs, c'est-à-dire qu'ils peuvent être perçus, créés, détruits et modifiés par les agents. (3) Un ensemble A d'agents, qui sont des objets particuliers ($A \subseteq O$), lesquels représentent les entités actives du système. (4) Un ensemble de relations R qui unissent des objets (et donc des agents) entre eux. (5) Un ensemble d'opérations O_p permettant aux agents de A de percevoir, produire, consommer, transformer et manipuler des objets de O . (6) Des opérateurs chargés de représenter l'application de ces opérations et la réaction du monde à cette tentative de modification, que l'on appellera les lois de l'univers » [104].

Selon cette définition, dans le domaine de la robotique par exemple, les agents A sont les robots, E est l'espace dans lequel se déplace les robots, une salle par exemple, O est l'ensemble des objets qui existent dans la salle et qui peuvent présenter des obstacles pour les agents A , R est l'ensemble des relations qui relient les agents entre eux (des messages peuvent être communiqués entre les robots) et O_p est l'ensemble des opérations exécutées par les robots tels que les opérations de déplacement, de communication, d'éloignement des autres objets, etc.

Le concept « système multi-agent » a été motivé par la réalisation des entités computationnelles intelligentes qui simulent le comportement humain et qui communiquent entre elles. Par ailleurs, Les systèmes multi-agents permettent de réduire la complexité de la résolution des problèmes en divisant un problème en sous problèmes, en les distribuant sur les agents et en coordonnant l'activité de ces agents. On parle ainsi de l'intelligence artificielle distribuée.

3.3.2 Intelligence artificielle distribuée

L'intelligence artificielle distribuée consiste à distribuer de l'expertise sur un ensemble d'entités qui communiquent pour atteindre un objectif global. Elle vise une meilleure structuration et organisation des connaissances factuelles et opératoires et la traduction d'un ensemble d'expertises sous une forme unifiée [108].

3.3.3 Applications des systèmes multi-agents

Les domaines d'application des SMA sont très riches : télécommunications, gestion et extraction de données, contrôle du transport, e-commerce, interfaces homme machine, simulations d'environnements sociaux et écosystèmes, divertissement, applications médicales, intelligence ambiante. Dans son mémoire, [109] distingue deux types d'utilisation des SMA : la simulation ou la modélisation de phénomènes complexes, la résolution de problèmes et la prise de décision.

La simulation multi-agent est souvent utilisée dans les domaines de la physique, la chimie, la biologie, les sciences sociales, etc., pour modéliser un phénomène naturel, économique, social, etc. Selon [110] les techniques de simulation existantes posent des problèmes dont les principaux sont : étanchéité du niveau d'analyse, complexité et t réalisme des paramètres, difficulté à modéliser l'action, etc. Les systèmes multi-agents apportent une solution radicale pour ces problèmes en permettant de simuler les interactions existantes entre des agents autonomes. [111] est un exemple de simulateur qui vise à simuler la dynamique de population des poissons, en tenant compte des nombreux facteurs biologiques et topologiques qui peuvent affecter leur évolution, et la prise de décision des pêcheurs.

La résolution de problèmes et la prise de décision supposent qu'il est possible de résoudre un problème complexe en le divisant en sous-problèmes exécutés par des agents autonomes et disposants de compétences complémentaires. Par exemple, le système CONDOR [112] est un SMA conçu pour l'aide à la conception de produits électromécaniques. Il est composé d'un ensemble d'agents représentés par les différents spécialistes en conception, en assemblage, en production, etc., qui coopèrent ensemble pour un objectif global commun, la construction d'un produit industriel. L'approche multi-agents se révèle être une piste très prometteuse dans le domaine de l'e-santé. Elle fournit une plateforme puissante pour l'aide à la décision médicale sous différents aspects (traitement, diagnostic, pronostic, etc.) [113]. Dans ce même contexte, [114] propose un système nommé RPM (*Remote patient system*) utilisant une architecture basée sur les systèmes multi-agents pour la téléassistance médicale des personnes âgées souffrant d'une maladie chronique. L'objectif principal de ces travaux est d'améliorer la

prise de décision médicale et de promouvoir la qualité de vie du patient en minimisant les coûts de transport et de prise en charge médicale.

[115] combine l'approche multi-agents avec le raisonnement à base de cas, ou (*case-based reasoning*) (CBR) en anglais pour l'aide à la décision médicale. Un agent ontologique est conçu pour proposer des suggestions de traitement que ce soit pour le patient ou pour le professionnel de santé en se basant sur une base de connaissances du domaine et sur l'analyse des cas cliniques similaires. Le raisonnement à base de cas sert principalement dans ce travail à trouver, mémoriser et à restaurer les données d'expérience pour résoudre des problèmes analogues.

3.4 Les systèmes sensibles au contexte

3.4.1 Le concept "contexte"

Selon Schilit [56], le contexte inclut l'identité et la localisation des personnes et des objets. Cette étude restreint le contexte à la réponse aux deux simples questions : « Où es-tu ? », « Avec qui es-tu ? ». [116] quant à lui a défini le contexte comme les éléments qui concernent l'environnement de l'utilisateur. Cet environnement, qui selon lui inclut, en plus de la localisation et l'identité, la saison, la température, l'heure, etc.

Le concept du contexte a été revu dans [54] en intégrant une nouvelle notion: la pertinence. Le contexte est ainsi défini comme n sous-ensembles d'états physiques et conceptuels ayant un intérêt pour une entité particulière. [53] a abouti à la définition suivante : « *Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and application themselves* ». Dans l'optique des travaux sur les interfaces homme machine (IHM) [117], le contexte se représente par le triplet <plateforme, environnement, utilisateur> où l'environnement est un ensemble d'entités qui peuvent changer le comportement de l'utilisateur ou du système.

A la lumière de cette définition, nous concluons que le contexte est un ensemble d'information liées à une entité située dans un environnement donné et auquel une application peut adapter son comportement.

3.4.2 Définition des systèmes sensibles au contexte

Selon [118], un système est sensible au contexte s'il utilise le contexte pour offrir des informations ou des services pertinents pour l'utilisateur. Brawn et al. [57] définit les applications sensibles au contexte comme étant des applications dont le comportement peut varier en fonction du contexte. Ces définitions partagent l'idée qu'un système sensible au contexte est capable de changer son plan d'adaptation en fonction de la dynamique qui caractérise les données contextuelles.

Un système ambiant se situe dans un environnement en perpétuelle évolution. Il doit ainsi être aussi sensible au contexte pour qu'il puisse s'adapter en conséquence. Le terme

informatique sensible au contexte ou *Context aware computing* a été introduit par [56]. Parmi les premières applications sensibles au contexte, citons l'application de [119] qui propose un assistant pour guider les utilisateurs à faire leurs achats en fonction de leurs localisations dans le magasin, et l'application de [120] qui assiste les utilisateurs de terminaux mobiles en adaptant la taille de l'écriture en fonction de l'activité de l'utilisateur et en changeant le mode de téléphone (silencieux, vibreur, sonnerie) en fonction de l'environnement externe de l'utilisateur.

3.4.3 Adaptation des systèmes sensibles au contexte

Le concept « adaptation » en environnement ubiquitaire sous-entend qu'il existe un ensemble de stratégies et de mécanismes à appliquer pour répondre aux variations de l'environnement. [121] a distingué deux types d'adaptation : l'adaptation réactive et l'adaptation proactive. L'adaptation réactive signifie que lorsque le contexte arrive à une certaine situation, une opération d'adaptation se déclenche. Par exemple, éteindre la lumière si personne n'est dans la chambre ou bien être prévenu lorsqu'une personne est à proximité, etc. L'adaptation proactive consiste à intercepter les invocations de chaque composant de l'application et à ré-aiguiller ces appels vers les opérations appropriées, tel que la redirection de requête d'un composant indisponible vers un autre disponible.

Dans la littérature sur les systèmes pervasifs, l'adaptation peut se répercuter sur quatre éléments : le service, l'interface, le contenu et le logiciel. [81] propose un service dynamique et adaptatif pour traiter à la fois les changements en environnements ubiquitaire et la contrainte « ressources limitées ». Trois étapes sont importantes pour concevoir les systèmes auto-adaptatifs : la modélisation de l'adaptation, l'analyse et la validation. [122] combine l'aspect orienté modèles et les modèles à l'exécution (*runtime*) pour concevoir des modèles d'adaptation dans le but d'améliorer la configuration des systèmes pendant l'exécution. Le modèle d'adaptation inclut un ensemble de règles d'adaptation qui ont été introduites pour générer des changements dans le comportement du système à l'exécution.

3.4.4 Exemples d'intergiciels sensibles au contexte

Les premières applications intégraient dans leur code des tâches de sensibilité au contexte, telles que la collecte des données contextuelles, l'analyse, l'adaptation au contexte, ce qui rendait ces applications extrêmement dépendantes des types de capteurs, du processus d'adaptation, du type de données à traiter. Pour pallier à ce problème, des travaux se sont focalisés sur la conception et le développement des intergiciels sensibles au contexte tels que Context Toolkit [118], CAMidO [123], CA3M [124]. Ces intergiciels permettent de découpler la partie sensibilité au contexte de la partie métier de l'application. Nous détaillons ci-après quelques exemples d'intergiciels.

3.4.4.1 Exemple de CAMidO

CAMidO est un intergiciel pour la sensibilité au contexte basé sur un méta-modèle d'ontologie (Figure 19) [123]. L'originalité de ce travail se décèle dans le fait qu'il associe

l'aspect modélisation du contexte à la conception d'un intergiciel sensible au contexte. CAMidO a été fortement basé sur les ontologies pour modéliser non seulement le contexte (couche contexte) mais aussi le mode d'acquisition des informations de contexte (couche intergiciel) et les actions à exécuter en cas de changement, c'est-à-dire les règles d'adaptation.

L'architecture CAMidO est composé de : *Collection Manager* qui est chargé de la collecte des informations contextuelles issues des capteurs. Ces données seront transférées vers le *ContextInterpreter* et le *ContextAnalyser*. Le *Context analyser* permet de détecter les changements contextuels pertinents en comparant les données collectées avec les anciennes données stockées dans le *ContextRepository*. Le *ContextInterpreter* déduit les données contextuelles via la relation «*Howdeduce*» fournie dans le meta-modèle de CAMidO. Le *ComponentAdapter* adapte le composant de l'application en fonction des changements du contexte et selon des règles définies par le concepteur de l'application.

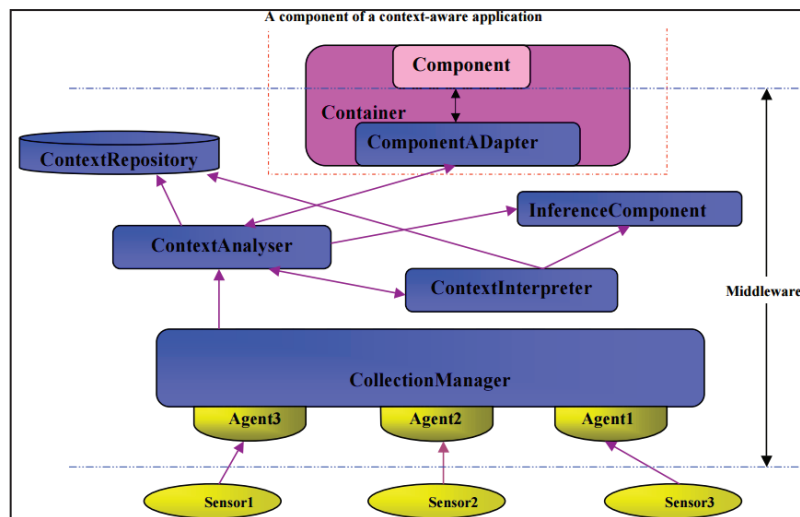


Figure 19. Architecture CAMidO [123]

3.4.4.2 Exemple de CA3M

CA3M est un modèle pour la sensibilité au contexte pendant l'exécution, et un intergiciel pour la gestion du contexte dynamique [124]. L'architecture CA3M (Figure 20) se compose de l'application sensible au contexte, de l'intergiciel CA3M, des collecteurs distribués (*Distributed Collectors*) et des sources du contexte distribué (*Distributed Context Sources*). La gestion du contexte est pilotée par le modèle CA3M. Les *Distributed Collectors* permettent la gestion du contexte et son interprétation. Ils collectent les données contextuelles à partir du *Distributed Context Sources* et les interprètent. L'architecture CA3M adopte le concept de boucle de contrôle «*control loop*» (Figure 19) fournie par les systèmes autonomes [125]. En effet, le modèle CA3M correspond à la partie «*Knowledge*» de la boucle, le collecteur est dans les parties «*Monitor*» et «*Analyse*», le *CaController* est dans la partie «*Plan*» et l'application est dans la partie «*Execute*» de la boucle.

L'intergiciel CA3M comporte deux composants : le *CaController* et le *Model Manager*. La relation entre ces deux composants est bidirectionnelle. Le *CaController* peut interroger le *Model Manager* pour exécuter le modèle CA3M. L'exécution de ce dernier est suivi par la

création d'un pont ou d'un lien pour chaque observable du contexte. Ce pont permet de découpler le code de l'application de la gestion du contexte et de la collecte des données contextuelles. En cas du changement du modèle, le *Model Manager* demande au *CAController* de créer ou bien de supprimer les liens.

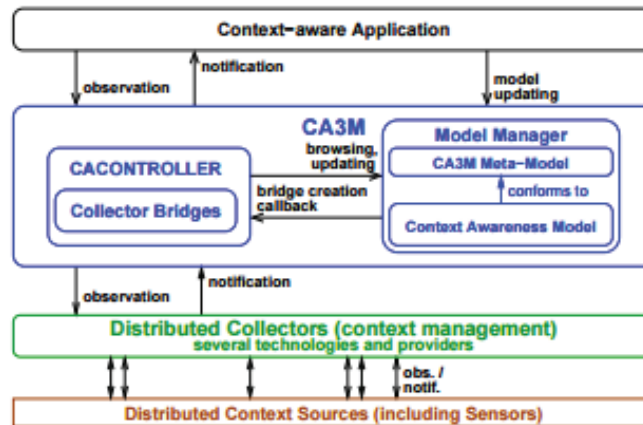


Figure 20. Architecture de CA3M [124]

L'originalité de cette architecture est qu'elle s'appuie sur un modèle sensible au contexte qui est changeable à l'exécution, une piste très prometteuse. Ce qui favorise l'adaptation de l'application au moment de l'exécution en cas de changement du contexte. Toutefois, contrairement à l'architecture CAMidO, l'architecture de CA3M ne fournit pas des détails sur le processus d'adaptation et les règles qui peuvent s'appliquer. D'autres concepts n'ont pas été abordés dans ce travail à savoir le respect de la vie privée lors de la collecte des données contextuelles, et la qualité du contexte collecté.

3.4.4.3 Exemple de Context Toolkit

Context Toolkit est un cadre qui vise à faciliter le développement et le déploiement des applications sensibles au contexte [118]. L'architecture Toolkit (Figure 21) met en évidence le découplage entre applications et gestion de contexte. Elle comporte plusieurs boîtes à outils pour la gestion du contexte, des *widgets* pour la collecte, des *interpreters* pour le calcul de contextes interprétés, des *aggregators* qui symbolisent les entités observables, un *discoverer* pour enregistrer et découvrir les différentes boîtes à outils.

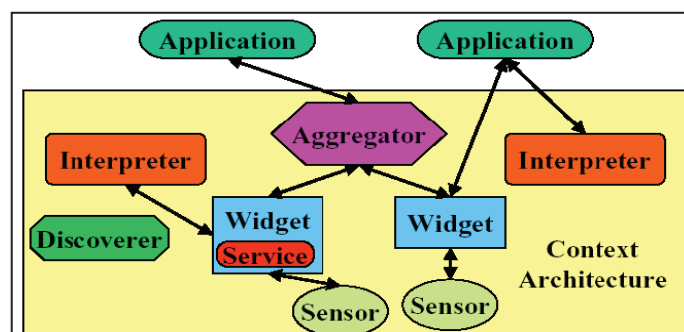


Figure 21. Architecture Toolkit [118]

3.4.5 Modélisation du contexte

La modélisation du contexte permet de formaliser les données du contexte, améliorer l'accès aux données, et constitue un pas très important vers le découplage entre applications et gestion de contextes. Dans la littérature, plusieurs approches ont été adoptées pour modéliser le contexte, telles que l'approche orientée base de données et ontologie et l'approche Ingénierie Dirigée par les Modèles (IDM).

3.4.5.1 Approche orientée ontologie

3.4.5.1.1 Principe de l'ontologie

Située au carrefour de plusieurs domaines en informatique tels que l'ingénierie des connaissances, l'ontologie fournit un formalisme permettant à la fois de représenter le contexte et de raisonner sur des données contextuelles. Une ontologie représente un ensemble de concepts organisés en graphe dont les relations sont sémantiques. Les ontologies sont généralement associées à des moteurs d'inférence pour raisonner sur le contexte en se basant sur un ensemble de règles d'inférence stockées dans une base de règles. Dans le cadre du web sémantique, plusieurs langages de description sont apparus pour modéliser les ontologies et permettre le partage des connaissances comme le langage OWL, langage basé sur RDF (Resource Description Framework) [126].

3.4.5.1.2 Exemple du méta-modèle CAMidO

Afin de faciliter la création des applications sensibles au contexte et garantir la réutilisation du contexte, [123] a proposé un méta-modèle basé sur des ontologies décrites en langage OWL. Ces ontologies permettent de décrire le contexte, les capteurs à partir desquels les données contextuelles sont collectées, les règles d'interprétation et les politiques d'adaptation (Figure 22).

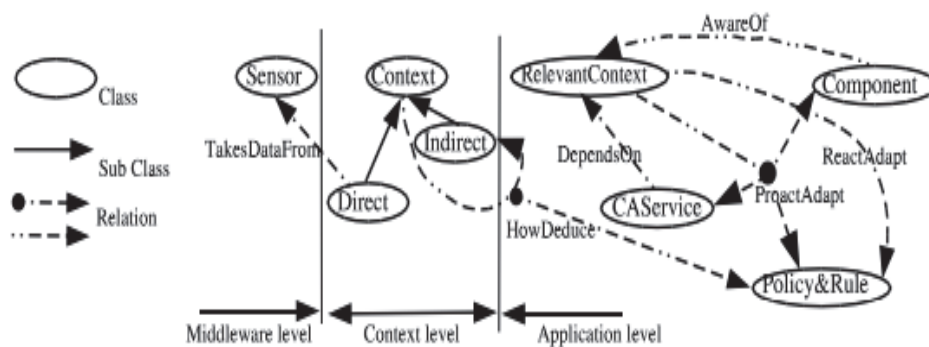


Figure 22. Le méta-modèle CAMidO [123]

Nous présentons ci-après un exemple de règle d'adaptation décrite en langage OWL. La règle indique la redirection de la requête « *ProvideImageAndText* » vers le service « LV » s'il n'y a pas de connexion. La connexion représente ici le contexte.

```

:ProactAdapt rdf:ID="adaptation2">
  <IfRelevantContext rdf:resource="#noConnexion"/>
  <WhenRequest>ProvideImageAndText</WhenRequest>
  <RedirectTo>LV</RedirectTo>
:/ProactAdapt>

```

3.4.5.2 Approche orientée Ingénierie Dirigée par les Modèles (IDM)

3.4.5.2.1 Principe de l'IDM

L'ingénierie dirigée par les modèles connue sous l'acronyme IDM, ou MDE en anglais pour *Model Driven Engineering*, est une pratique d'ingénierie des systèmes en pleine émergence. Elle fournit des outils, concepts et langages pour créer et transformer des modèles afin d'améliorer le développement des systèmes d'informations complexes. L'IDM repose sur l'aspect abstraction, sur les concepts clé suivants : modèles, méta-modèle, méta-méta-modèle. Selon [127], un modèle est une simplification d'un système construit dans une intention particulière. Un méta-modèle est un modèle qui définit le langage d'expression d'un modèle [128]. Un méta-méta-modèle est un modèle qui décrit un langage de méta-modélisation. Tous ces concepts sont organisés par l'OMG (*Object Management Group*) [126] sous forme d'une architecture à 4 niveaux structurée en pyramide (Figure 23). Le niveau M_0 correspond à la base de la pyramide et représente le monde réel, en d'autres termes l'implémentation du programme. Le niveau M_1 de la pyramide correspond aux modèles du programme. A un niveau plus abstrait M_2 , on trouve les méta-modèles. Enfin, le sommet de la pyramide, le niveau M_3 , représente le MOF (*Meta Object Facility*). Le MOF représente le méta-méta-modèle unique servant de base à la définition de tous les méta-modèles.

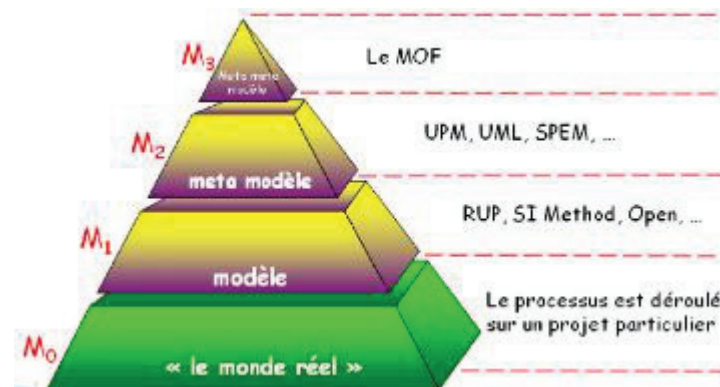


Figure 23. Pyramide de modélisation de l'OMG

Architecture dirigée par des modèles MDA– L'OMG a intégré l'architecture dirigée par des modèles, connue sous l'acronyme MDA (*Model Driven Architecture*), pour promouvoir l'utilisation de l'ingénierie des modèles. Le principe de MDA consiste à modéliser l'application d'une façon indépendante de son code d'implémentation. En d'autres termes, le MDA permet de distinguer la modélisation technique de la modélisation fonctionnelle, ce qui rend la réutilisation des modèles possible. MDA est composé de plusieurs modèles UML qui vont servir à modéliser l'application, puis à générer automatiquement le code source par transformations successives. Tout le code n'est pas généré, une partie du code qui représente

le squelette de l'application est fourni à l'aide des outils générateurs de code comme Acceleo de la fondation Eclipse.

Transformation des modèles – C'est le passage d'un niveau de modèle à un autre. Il s'agit par exemple de passer d'un méta-modèle qui décrit les contraintes que doit respecter l'application au modèle fonctionnel. La transformation peut se faire aussi en partant d'un modèle métier indépendant de l'informatisation (*Computation Independent Model, CIM*) vers un modèle indépendant de la plate-forme (*Platform Independent Model, PIM*), et enfin vers un modèle spécifique à la plate-forme cible (*Platform Specific Model, PSM*) pour l'implémentation concrète du système. Le PSM prend en compte les contraintes de déploiement et d'implémentation et peuvent utiliser des langages de programmation comme C++, Java. La transformation d'un modèle vers un autre modèle se fait par des outils de transformation comme ATL (*ATLAS Transformation Language*) qui est disponible en tant que plugin dans le projet Eclipse.

3.4.5.2.2 Exemples de modélisation du contexte en IDM

3.4.5.2.2.1 Exemple du méta-modèle CA3M

La Figure 24 décrit le méta-modèle de CA3M. Un système sensible au contexte est composé d'un ensemble d'entités. Une entité est un objet physique ou logique pouvant être observée, par exemple un dispositif. Chaque entité est associée à un ou plusieurs observables, par exemple le niveau de batterie, l'état de la connexion, etc. L'observable peut être interprété à partir d'autres observables atomiques, par exemple : l'état de la connexion est dérivé à partir du débit et de la latence. Une situation d'adaptation est un observable interprété qui prend un nombre déterminé de valeurs. Selon ces valeurs, une adaptation peut être nécessaire (par exemple : niveau de batterie = 'très faible'). Un observable est lié à la classe « *ContextAwarenessContract* ». Ce dernier définit les événements et les notifications qui peuvent répercuter des modifications au niveau de l'application du système sensible au contexte « *ContextAwareSystem* ».

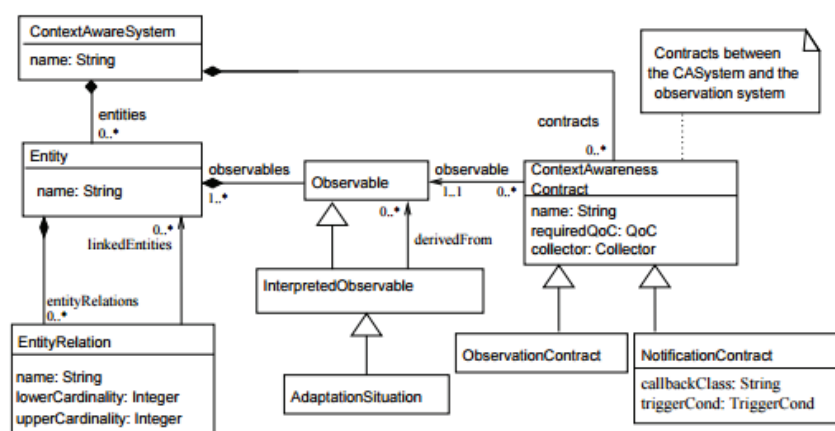


Figure 24. Méta-modèle CA3M de la sensibilité au contexte [124]

3.4.5.2.2 Exemple du méta-modèle ContextUML

[129] s’est appuyé sur l’IDM pour concevoir un méta-modèle de contexte en UML. Un des intérêts majeurs de cette approche est de pouvoir établir un lien entre les éléments du contexte et les éléments de l’application (service, opération, message, paramètres) et de pouvoir intégrer des contraintes en OCL (*Object Constraint Language*) [126] pour exprimer les règles qui sous-tendent les situations d’adaptation et les actions que doit exécuter l’application en cas de changement de contexte. Deux aspects sont pris en considération dans le méta-modèle du ContextUML: l’aspect modélisation du contexte et l’aspect modélisation de la sensibilité au contexte (*context awareness*) (Figure 25).

La modélisation du contexte intègre les classes « contexte » et « source du contexte ». Il existe deux catégories de contexte : le contexte atomique et le contexte composite. Le contexte atomique est le contexte de bas niveau qui ne dépend d’aucune autre donnée contextuelle. Il peut être directement fourni à partir des sources du contexte (par exemple des capteurs). Au contraire, le contexte composite se dérive à partir d’autres contextes et regroupe plusieurs autres données. Dans la modélisation de la sensibilité au contexte, on trouve la classe « *CA mechanism* » et la classe « *CAObject* » avec leurs classes filles. « *Context awareness mechanisms* » est assigné aux objets sensibles au contexte « *CAObject* » via la relation « *MechanismAssignment* ». Ceci signifie que chaque objet de l’application est lié à un mécanisme de sensibilité de contexte. Un objet peut être un service, une opération, un message ou une partie. Chaque service fournit une ou plusieurs opérations et chaque opération appartient à un seul service. Il existe deux types de « *Context awareness mechanisms* » : « *Context Binding* » et « *Context triggering* ». « *Context Binding* » permet d’établir un lien entre l’objet et son contexte, par exemple entre l’objet utilisateur et son contexte localisation. Le « *Context triggering* » modélise le mécanisme d’adaptation du service en fonction du contexte. Il comporte un ensemble de contraintes et d’actions représentées respectivement par les classes « *Constraint* » et « *Action* » dans le méta-modèle ContextUML. Les actions sont exécutées si et seulement si toutes les contraintes sont vérifiées. Un exemple de contraintes formalisées en logique booléenne peut être « NiveauBatterieFaible = vrai » qui déclenche l’action « ActiverModeEconomieEnergie ».

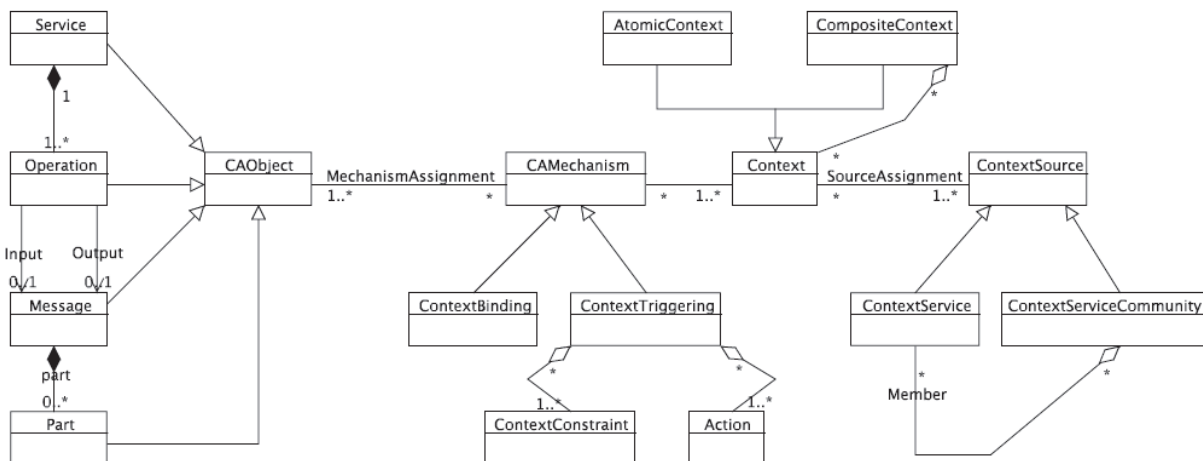


Figure 25. Le méta-modèle ContextUML [129]

Contrairement au méta-modèle CA3M, le méta-modèle ContextUML est beaucoup plus explicite. En effet, il a permis la distinction entre les types de contexte (atomique et composite). Ce méta-modèle a également explicité le mécanisme d'adaptation, les contraintes et les actions à déclencher en cas de changement du contexte. En outre, il a détaillé les différents types d'objets de l'application qui peuvent être sensibles au contexte (service, opération, message, partie). Néanmoins, ce modèle peut être aussi étendu en intégrant les aspects confidentialité et respect de la vie privée de l'entité source du contexte. En effet, de notre point de vue, la collecte des données contextuelles d'une personne (par exemple sa localisation) doit se faire sous l'accord de la personne en question et donc en fonction de ses préférences. Une relation de dépendance pourra ainsi être établie entre une nouvelle classe que nous pouvons nommer « préférences » et « *ContextBinding* ». Pour promouvoir la pertinence du contexte collecté, nous proposons également l'ajout d'une classe qui permet de définir le rôle de l'entité source du contexte que nous pouvons nommer « *Role* » et dont dépend le lien existant entre l'objet et le contexte « *ContextBinding* ». Nous pensons que le choix des données contextuelles recueillies à partir de l'entité dépend du rôle de l'entité elle-même dans l'application. Par exemple, dans une application de forum mobile nous avons besoin de savoir le contexte « localisation » de la personne qui joue le rôle de l'expéditeur des messages.

3.4.5.3 Discussion

Les travaux de recherche abordant le concept « sensibilité au contexte » sont très nombreux. Il y a des travaux qui se concentrent sur la modélisation du contexte, d'autres sur la conception et le développement des intergiciels pour la gestion du contexte et d'autres sur la réalisation des intergiciels d'adaptation. Dans le cadre de notre travail, nous allons prendre avantage du concept sensibilité au contexte et de ses mécanismes d'adaptation pour concevoir notre système de gestion d'acheminement des messages en e-santé. Nous retenons la décomposition fonctionnelle de la prise ne compte du contexte qui est conforme à la boucle de contrôle des systèmes autonomes. Rappelons que cette décomposition respecte globalement les étapes suivantes : Collecte des informations contextuelles, détection des changements de situations de contextes, réalisation des adaptations (sans intervention de l'utilisateur). Pour concevoir notre système, nous nous intéressons à l'adaptation dynamique, c'est-à-dire au changement du comportement de l'application à l'exécution. Nous admettons que les décisions de routage par exemple soient automatiquement adaptées au moment de l'exécution si un changement du contexte est détecté. A cet effet, nous pouvons bénéficier du méta-modèle CA3M [124] puisqu'il traite la problématique de l'adaptation à l'exécution (*run-time*). De ce modèle, nous retenons la classe « *ContextBinding* » qui reflète l'aspect dynamique lors de la création des liens automatiques entre le l'objet et son contexte pertinent. L'approche ontologie adoptée dans la conception de l'architecture CAMidO [123] est aussi très importante pour à la fois la modélisation du contexte et l'interprétation des données contextuelles. Elle intègre l'aspect intelligence artificielle pour interpréter le contexte et le transformer à un niveau plus haut d'abstraction.

Les travaux présentés dans ce chapitre ont mis en lumière l'importance du paradigme « sensibilité au contexte » et le concept de découplage entre application et système de gestion de contexte. Toutefois, ils n'ont pas tenu compte de la qualité du contexte. Selon [130], la qualité du contexte (QoC) signifie « *any inherent information that describes context information and can be used to determine the worth of the information for a specific Application* ». [131] identifie quelques critères de la qualité du contexte comme : l'exactitude, la précision, la confiance, la granularité de l'information, l'actualisation. Les données provenant des sources sensibles comme les capteurs peuvent être incorrectes, incertaines et incomplètes. Il est ainsi très important que la gestion du contexte prenne en compte ces contraintes pour pouvoir raisonner sur un contexte certain, ce qui va améliorer la prise de décisions [132] [133].

3.5 Apport des systèmes autonomes à la sensibilité au contexte et aux systèmes multi-agents

L'auto-configuration, l'auto-guérison, l'auto-optimisation, etc., toutes ces caractéristiques des systèmes autonomes requièrent que ces systèmes soient à l'écoute de l'environnement, capables de détecter les changements du contexte pour pouvoir s'adapter et donc s'auto-configurer, s'autogérer, etc. Ainsi, les systèmes autonomes adoptent le concept sensibilité au contexte et les mécanismes d'adaptation. L'architecture classique des intergiciels sensibles au contexte offre de fortes similitudes avec le modèle MAPE-K de *l'autonomic computing* [101]. Cela consiste dans un premier temps à collecter les données brutes. C'est la phase de recueil des données contextuelles. Ensuite, ces données sont interprétées et transformées en des observables à un niveau plus haut d'abstraction. C'est l'étape de la transformation des observables. Ces observables servent par la suite à la phase de décision, l'identification des situations d'adaptation et à l'identification du plan de réaction. Ce dernier sera exécuté par un mécanisme d'adaptation sur l'application. Ces étapes sont illustrées dans le cycle de la prise en compte du contexte dans le travail de [134]. Les systèmes autonomes et les systèmes multi-agents constituent un changement de paradigme important dans le domaine de l'intelligence artificielle. Il s'agit du même changement de paradigme qui fait que l'informatique s'intéresse de plus en plus aux problèmes du calcul distribué (parallélisme massif, etc.), que la gestion des problèmes abandonne les structures monolithiques pour des structures réparties. Les systèmes multi-agents font appel au concept de l'autonomie qui caractérise les agents qui les composent.

3.6 Conclusion

Ce chapitre a permis d'explorer les systèmes autonomes et les systèmes multi-agents ainsi que leurs caractéristiques. Nous avons vu que les systèmes autonomes, suscités par la complexité des systèmes informatiques, sont capables de s'autogérer, de s'auto-adapter aux changements imprévus. L'autonomie fait appel à un concept très important, la sensibilité au contexte, que nous avons présenté dans ce chapitre. Nous avons présenté les travaux réalisés concernant les intergiciels CA3M, CAMidO. Ces intergiciels s'appuient sur des modèles de

sensibilité du contexte modifiables à l'exécution et permettant l'adaptation de l'application au contexte. Cependant, le verrou à lever aujourd'hui est de réaliser l'adaptation dans des délais acceptables. Des évaluations du temps nécessaire à une adaptation deviennent ainsi essentielles.

La modélisation du contexte est une étape incontournable pour le découplage entre les tâches de gestion de contextes et l'application. Nous avons investigué deux approches pour modéliser les données contextuelles : une approche orientée ontologie et une approche orientée Ingénierie Dirigée par les modèles (IDM). L'approche ontologique est la plus souvent utilisée vu la sémantique qu'elle intègre et sa capacité non seulement à modéliser le contexte, mais aussi à raisonner sur les données contextuelles.

Chapitre 4

Méthodes d'aide à la décision et de modélisation des systèmes dynamiques

4.1 Introduction

Les systèmes de gestion d'acheminement des messages requièrent une certaine intelligence pour promouvoir les prises de décisions en cours d'exécution de routage. En effet, le choix de la stratégie d'acheminement la plus adaptée aux contextes source et cible du message nécessite notamment des prises de décisions intelligentes au niveau du choix du destinataire, du temps requis pour le traitement du message, des changements d'acheminement à effectuer en temps réel si nécessaire, etc. En outre, ces systèmes doivent faire face au challenge « dynamique du contexte » et fournir des adaptations dynamiques en fonction des données contextuelles collectées et de leurs changements. Ces deux aspects « dynamique » et « intelligence » nous intéressent tout particulièrement dans notre problématique de recherche. L'objectif de ce chapitre est ainsi d'explorer quelques méthodes d'intelligence artificielle pour l'aide à la décision et des méthodes de modélisation des systèmes dynamiques. Ce chapitre aborde un aspect plutôt méthodologique qui s'articule autour de différents types de modélisation à savoir la modélisation stochastique (les réseaux bayésiens statiques et dynamiques), mathématique (les réseaux de Petri), méta heuristique (les colonies de fourmis), et logique (les systèmes experts et les arbres de décision).

Le présent chapitre est agencé comme suit. Dans un premier temps, nous présentons les méthodes d'aide à la décision, tels que les arbres de décision, les systèmes experts, les colonies de fourmis, les réseaux bayésiens statiques et dynamiques. Dans un deuxième temps, nous présentons les réseaux de Petri comme étant un outil pour la modélisation du comportement dynamique des systèmes. Notons que la liste des méthodes exposées dans ce chapitre n'est pas exhaustive. Nous nous limitons à quelques méthodes eu égard à leur pertinence pour notre problématique de recherche.

4.2 Méthodes d'intelligence artificielle pour l'aide à la décision

L'aide à la décision a donné naissance à de nombreuses méthodes d'intelligence artificielle tels que les arbres de décisions, les systèmes experts, les réseaux bayésiens que nous présentons dans cette section. Nous allons explorer l'ère probabiliste à travers les réseaux bayésiens statiques et dynamiques et les colonies de fourmis. Il existe notamment d'autres

méthodes qui s'appuient également sur des formalismes probabilistes, comme les réseaux de neurones. Néanmoins, elles ont besoin de plusieurs cas réels servant d'exemples pour leur apprentissage (base d'apprentissage). Les systèmes experts, quant à eux s'appuient sur une approche déterministe qui modélise plutôt le raisonnement logique de l'expert. Mais, dans notre problématique, il est très difficile de définir des règles d'acheminement propres à chaque scénario en e-santé et aux différentes situations contextuelles.

Notre objectif est d'utiliser la méthode qui pallie au mieux les problèmes spécifiques posés par notre problématique d'acheminement des messages, à savoir : prendre des décisions basées sur des données incomplètes et de natures différentes et modéliser l'incertitude inhérente au raisonnement sur le contexte.

4.2.1 Les arbres de décision

4.2.1.1 Définition

Un arbre de décision est un outil d'aide à la décision qui a pris son plein essor dans le cadre des approches de l'intelligence artificielle et plus particulièrement l'apprentissage automatique *Machine Learning*. Les arbres de décision sont appliqués dans différents domaines tels que la sécurité, la médecine. Ils permettent la classification supervisée d'une population par exemple d'individus (patients, clients d'une entreprise, consommateurs d'un produit, etc.) selon certains critères (maux de gorges, fidélité, temps passé sur un site marchand, etc.). La traduction de l'arbre de décision se fait sous la forme de règles logiques (**Si** telles conditions sur telles variables sont remplies **alors** le cas appartient à une telle classe). Il existe deux types d'arbre de décision : (1) les arbres de classification (*Classification Tree*) qui permettent de déterminer à quelle classe un individu de la population appartient, et le résultat est une étiquette de classe, par exemple, la personne est malade ou non, (2) les arbres de régression (*Regression Tree*) permettent de prédire une valeur numérique de la classe, par exemple le niveau de priorité du message (0, 1, 2).

4.2.1.2 Exemple introductif

Pour introduire la notion d'arbre de décision, nous allons nous appuyer sur un exemple très connu présenté dans l'ouvrage de Quinlan [135]. L'exemple d'arbre présenté dans la Figure 26 permet de prédire le comportement des joueurs (jouer ou non) en fonction des prévisions météorologiques (ensoleillement, température, humidité, vent).

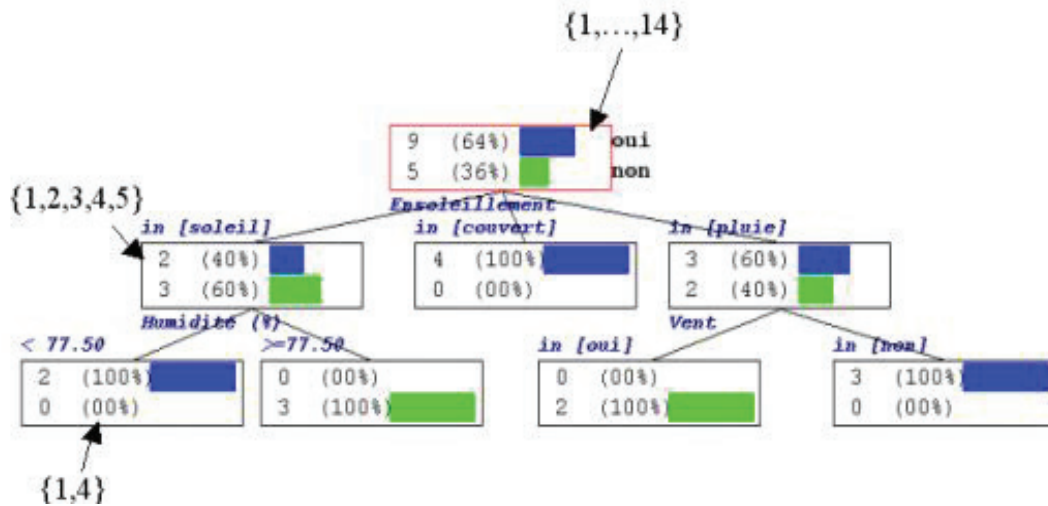


Figure 26. Arbre de décision sur le fichier de données « Weather » [135]

Cet arbre est construit à partir d'un fichier composé de 14 observations décrites dans le tableau ci-après.

Numéro	Ensoleillement	Température (°F)	Humidité (%)	Vent	Jouer
1	soleil	75	70	oui	oui
2	soleil	80	90	oui	non
3	soleil	85	85	non	non
4	soleil	72	95	non	non
5	soleil	69	70	non	oui
6	couvert	72	90	oui	oui
7	couvert	83	78	non	oui
8	couvert	64	65	oui	oui
9	couvert	81	75	non	oui
10	pluie	71	80	oui	non
11	pluie	65	70	oui	non
12	pluie	75	80	non	oui
13	pluie	68	80	non	oui
14	pluie	70	96	non	oui

Tableau 3. Données « Weather » [135]

Le premier nœud situé au premier niveau de l'arbre est appelé racine de l'arbre. Il décrit la distribution de la variable « jouer ». Dans les 14 observations du fichier des données « weather », il existe neuf cas où la variable « jouer = oui » et cinq où aucune partie n'a eu lieu « jouer = non ». La racine de l'arbre est étiquetée par un test (ensoleillement) appelé aussi variable de segmentation qui peut être appliquée aux différents nœuds de décision. Les réponses possibles aux tests (soleil, couvert, pluie) représentent les labels des arcs reliant le nœud père aux nœuds fils. Le nœud fils, à gauche, se situant sur le deuxième niveau et qui résulte de l'arc soleil, décrit la distribution de cinq observations où la variable « ensoleillement = soleil ». Ces cinq observations correspondent aux individus numéros (1, 2, 3, 4, 5) dans le fichier « weather ». Cette distribution de fréquence nous indique qu'il existe deux individus (1, 5) où « jouer = oui » et trois autres (2, 3, 4) où « jouer = non ». Le second fils du nœud racine, au milieu, correspond à la distribution de la variable « jouer » dans le cas où la variable de segmentation « ensoleillement » est égal à « couvert ». Nous constatons que

les quatre joueurs de numéros (6, 7, 8, 9) ont décidé de jouer : « jouer = oui ». Le nœud, tout à gauche sur le troisième niveau de l'arbre est segmenté via la variable de test « humidité ». De cette variable résulte deux résultats de test possibles (humidité <77,50 et humidité >77,50) selon un seuil dit seuil de discrétisation (égale à 77,50).

Le processus de décision continue jusqu'à l'arrivée aux feuilles de l'arbre où toutes les observations prennent une seule valeur, soit toutes « jouer = oui » ou toutes « jouer = non ». Une feuille dans un arbre de décision est un nœud terminal qui ne possède pas des nœuds-fils. Cet arbre de décision peut être traduit sous forme des règles logiques, par exemple, **Si** ensoleillement = pluie **et** vent = oui **alors** jouer = non.

Pour classer un nouvel individu, il suffit de suivre le chemin depuis la racine jusqu'à une feuille de l'arbre en effectuant les différents tests à chaque nœud, par exemple, ensoleillement est-il égal à soleil ou couvert ou pluie ? On associe à l'individu le résultat attaché à la feuille auquel il aboutit.

4.2.1.3 Algorithmes d'apprentissage d'arbres de décision

A partir de l'exemple présenté Figure 26, plusieurs questions peuvent se poser telles que : Comment choisir la première variable de segmentation, par exemple, « ensoleillement » ? Sur quels critères le choix a-t-il été fait ? Comment savoir que le nœud est terminal ? Pour répondre à ses questions, des algorithmes d'apprentissage ont été développés afin de construire des arbres de décision sur la base d'un ensemble d'apprentissage. Parmi ces algorithmes nous citons, l'algorithme CART (*Classification and Regression Tree*) [136], l'algorithme C4.5 [135] et l'algorithme CHAID (*Chi-squared Automatic Interaction Detector*) [137]. Ces algorithmes adoptent une approche *top-down* dans la construction de l'arbre de décision, c'est-à-dire de la racine vers les feuilles. Ils partagent le même schéma de fonctionnement qui est le suivant :

Algorithme 1. Algorithme d'apprentissage générique [138]

entrée : échantillon S

début

Initialiser à l'arbre vide ; la racine est le nœud courant

répéter

Décider si le nœud courant est terminal

Si le nœud est terminal **alors**

Affecter une classe

sinon

Sélectionner un test et créer le sous-arbre

FinSi

Passer au nœud suivant non exploré s'il en existe un

Jusqu'à obtenir un arbre de décision

fin

4.2.1.3.1 Décider si un nœud est terminal

Un nœud est terminal lorsque tous les exemples associés à ce nœud appartiennent à la même classe ou lorsqu'il n'y a pas d'attribut à tester au niveau de ce nœud. Dans l'algorithme CART, un nœud p est terminal si l'indice de Gini (p) $< i_0$ où i_0 est un paramètre à fixer. Le calcul de cet indice sera présenté dans la section suivante.

4.2.1.3.2 Choix des variables de segmentation

Le choix de la variable de segmentation correspond au choix de l'attribut qui fait le mieux progresser la discrimination des données, c'est-à-dire qui procure le meilleur gain. Le calcul de la performance de la variable de segmentation diffère d'un algorithme à un autre. Il est calculé en fonction de l'indice de Gini dans l'algorithme de CART, du critère d'entropie dans l'algorithme de C4.5 et du test statistique khi-2 dans l'algorithme de CHAID. Le choix de la variable de segmentation « ensoleillement », avec la méthode CHAID de l'exemple présenté Figure 26, a été bien justifié dans [139]. Rappelons qu'en statistique, l'indice de Gini permet de mesurer la dispersion d'une distribution dans une population donnée, par exemple, la dispersion des salaires dans un pays. Plus l'indice de Gini est proche de 1, plus l'inégalité est importante. Plus l'indice de Gini est proche de 0, plus la dispersion des salaires est faible.

Dans la suite de cette section, nous allons adopter la notation mathématique utilisée dans [140] et [138] pour démontrer le calcul des deux indices : indice de Gini et indice d'entropie.

L'indice de Gini est calculé selon la formule suivante :

Soient :

- S un échantillon, A l'attribut cible
- S_1, \dots, S_k partition de S selon les classes de A .

$$\text{Gini}(S) = \sum_{i=1}^k \frac{|S_i|}{|S|} \left(1 - \frac{|S_i|}{|S|}\right) = \sum_{i \neq j} \frac{|S_i| \times |S_j|}{|S|^2}$$

L'indice d'entropie est proposé par Theil dans les années 1967 pour la mesure des inégalités, par exemple les inégalités de revenu dans la société. Il est égal à la somme des parts pondérées par leur logarithme :

$$\text{Ent}(S) = - \sum_{i=1}^k \frac{|S_i|}{|S|} \log \left(\frac{|S_i|}{|S|}\right)$$

Si l'indice d'entropie est égal à 0, cela signifie qu'il y a une égalité absolue dans la population. Au contraire, indice =1 indique une inégalité. Dans le cas où les attributs sont binaires, soient :

- p : position courante de l'arbre en construction
- T : un test sur un attribut

- f = Entropie ou f = Gini, S_p échantillon associé à p .
- S_{pi} : ensemble des exemples de S_p qui satisfont la i -ème branche de T
- P_i : proportion des éléments de S_p qui satisfont la i -ème branche de T

Le gain d'un attribut à une position est égal :

$$Gain_f(p, T) = f(S_p) - \sum_{j=1}^2 P_j \times f(S_{pj})$$

4.2.2 Les systèmes experts

4.2.2.1 Définition

« Un système expert est un programme conçu pour simuler le comportement d'un humain qui est un spécialiste ou un expert dans un domaine très restreint » [141] [142]. Il s'agit de l'une des voies qui aboutit à l'intelligence artificielle. Un système expert permet de résoudre généralement des problèmes de raisonnement, de classification et d'aide à la décision (diagnostic médical, diagnostic de panne, analyse de situation économique, etc.). Une étude approfondie sur les systèmes experts est fournie dans [143] [144].

4.2.2.2 Composition d'un système expert

Un système expert est composé de deux parties indépendantes : une base de connaissance et un moteur d'inférence (Figure 27). La base de connaissances comporte une base de règles et une base de faits. La base de règles modélise les connaissances de l'expert sous forme de règles logiques. Une règle est composée d'une partie prémisse et d'une partie conséquence sous la forme : **Si** condition ou prémisse **alors** conséquence (Exemple : **Si** Température <37,5 **et** gorge irritée **alors** malade). La base des faits contient les informations concernant le cas à traiter. Le moteur d'inférence permet de répondre à des questions en raisonnant à partir des règles et des faits.

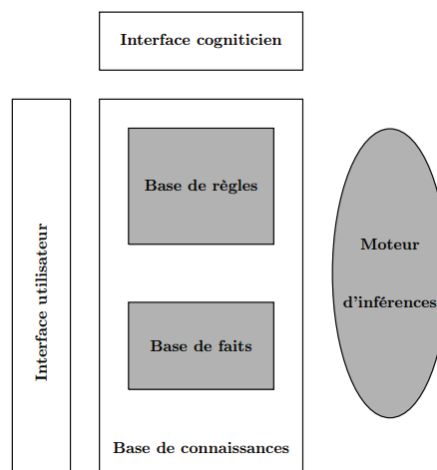


Figure 27. Organisation d'un système expert [145]

L'avantage de l'architecture d'un système expert est qu'il est possible de faire évoluer les connaissances sans avoir besoin de changer le mécanisme de raisonnement au niveau du moteur d'inférence. Les systèmes experts reposent sur des mécanismes de la logique formelle pour faire des déductions. Certains se basent sur la logique des propositions, c'est-à-dire la logique d'ordre 0, d'autres sur la logique des prédicats, c'est à dire la logique d'ordre 1, et d'autres sur la logique d'ordre 2.

La logique des propositions permet d'étudier les relations logiques entre des propositions. Une proposition peut fournir une information sur l'état d'un objet, par exemple « la lumière est éteinte », « la porte est fermée ». Les relations entre les propositions sont exprimées via des connecteurs logiques comme la conjonction, la disjonction, la négation, l'implication, l'équivalence, etc. La logique des propositions est insuffisante pour représenter des connaissances relatives à des systèmes plus complexes et expressifs alors que la logique des prédicats le permet. En effet, elle autorise l'intégration des variables à l'intérieur des règles. Ces variables permettent de déduire la valeur du prédicat en fonction de l'instanciation de la variable, par exemple le prédicat SeSitue (X, Europe) est vrai si X = 'France' et faux si X = 'Chine'. La logique d'ordre 2 ou d'ordre supérieur étendent la logique d'ordre 1 et permettent de décrire des règles plus complexes où les prédicats peuvent avoir pour variables des fonctions de variables.

4.2.2.3 Types de moteur d'inférence

Il existe trois catégories de moteurs d'inférence : Moteur à chaînage avant, moteur à chaînage arrière, moteur à chaînage mixte.

4.2.2.3.1 Moteur à chaînage avant

Le moteur à chaînage avant raisonne de façon déductive. Il déduit des faits-conséquences à partir des faits entrés par l'utilisateur dans la base des faits. Le processus de déduction est le suivant : le moteur parcourt toutes les règles existantes dans la base des règles et recherche les prémisses dans la base des faits. La conséquence de la règle ne peut être vérifiée que lorsque toutes les prémisses sont dans la base des faits. On dit alors que le moteur utilise des faits et des règles pour produire de nouveaux faits. Le processus de chaînage avant s'arrête lorsque le parcours des règles n'entraîne aucune conséquence. L'algorithme suivant extrait de [146] explique bien le fonctionnement du moteur d'inférence à chaînage avant :

Soient : BF, une base de faits ; BR, une base de règles ; F, le fait que l'on cherche à établir

Algorithme 2. Chaînage avant [146]

ChaînageAvant (BF, BR, F)

Debut

Tant que ($F \notin BF$) **et** ($\exists R \in BR$, applicable (R))

Faire

choisir une règle applicable R (par métarègles, heuristiques,)

$BR = BR - R$ (désactivation de R)

$BF = BF \cup \text{concl}(R)$ (déclenchement de la règle R, ajout de sa conclusion)

Fin tant que

Si $F \in BF$ **alors** F est établi

Sinon F n'est pas établi

Fin

Illustrons cet algorithme par un exemple très simple. Soient les règles suivantes :

R1 : **Si** maux de tête **alors** personne est fatiguée

R2 : **Si** toux sèche et fatigué **alors** personne grippée

Supposons que l'utilisateur a entré dans la base de faits, les faits suivants :

F1 : Maux de tête

F2 : Toux sèche

Dans un premier tour, le moteur d'inférence peut déduire à partir de la première règle un nouveau fait qui sera ajouté dans la base des faits, qui est le suivant :

La personne est fatiguée

Ensuite, dans un deuxième tour de déduction et au niveau de la règle 2, le moteur d'inférence déduit que la personne est grippée. Le moteur de déduction s'arrête enfin puisqu'il n'y a pas d'autres déductions supplémentaires.

4.2.2.3.2 Moteur à chaînage arrière

Le moteur à chaînage arrière raisonne d'une manière inductive. En effet, il permet d'induire à partir d'une conséquence les prémisses des règles qui aboutissent à la conséquence en question (Algorithme 3). En d'autres termes, il s'agit de remonter d'une façon récursive d'un fait final vers les hypothèses qui permettent de le vérifier.

Algorithme 3. Chaînage arrière [146]

ChainageArriere (BF, BR, F)
Debut
 Si (F ∈ BF) **alors** ChainageArriere ← OK
 Sinon
 construire ER ensemble de règles R, telle que F ∈ conclusion (R)
 Faire
 Valide ← VRAI
 R ← premier element de ER
 ER ← ER – {R}
 Pour tout Fr ∈ premisses (R)
 valide ← valide **et** ChainageArriere (BF, BR, Fr)
 Fin pour
 jusqu'a (valide **ou** ER ≠ ∅)
 Si valide **alors** BF = BF ∪ {F}
 ChainageArriere ← valide
Fin

Illustrons le fonctionnement de cet algorithme avec l'exemple suivant :

Soient les règles suivantes :

R1 : Si X alors Y

R2 : Si Y et Z alors W

R3 : Si W alors U

Supposons qu'on cherche à vérifier la conséquence « U ». Le moteur d'inférence parcourt les règles en remontant de la conséquence U jusqu'à X. Il indique à la fin que pour que U soit vérifié, il faut que les prémisses X et Z soient aussi vérifiées.

4.2.2.3.3 Moteur à chaînage mixte

Le moteur à chaînage mixte représente la combinaison entre le chaînage avant et le chaînage arrière (Algorithme 4). Il fonctionne comme le chaînage avant pour produire des conséquences-faits. Il remonte en arrière pour vérifier les prémisses des règles de la base de connaissances. Ce fonctionnement permet d'explorer de nouvelles combinaisons non envisagées par les règles.

Algorithme 4. Chaînage mixte [146]

ChainageMixte (BF, BR, F)

Debut

Tantque F n'est pas déduit mais peut encore l'être

Faire

Saturer la base de faits par chaînage AVANT

Chercher quels sont les faits encore éventuellement déductibles

Déterminer une question pertinente à poser à l'utilisateur et ajouter sa réponse à la base de faits

Fin tantque

Fin

4.2.3 Les colonies de fourmis

4.2.3.1 Définition

Initié par Marco Dorigo dans les années 1990, les algorithmes de colonies de fourmis s'inspirent des comportements des fourmis pour rechercher le chemin le plus court entre leur colonie et une source de nourriture. Ils constituent une famille de méta-heuristiques d'optimisation [147]. Selon [148], la méta-heuristique est un processus de génération itératif visant à résoudre des problèmes d'optimisation difficiles.

4.2.3.2 Expérience du double pont binaire

Initialement menée par Deneubourg et al. [149], l'expérience du double pont binaire a montré que les fourmis sont capables de résoudre des problèmes difficiles d'une façon collective notamment la recherche du chemin le plus court vers la source de la nourriture. Des fourmis circulent dans un graphe constitué de deux sommets, le sommet « Nid » et le sommet « source de nourriture » (Figure 28). Entre ces deux sommets, il y a deux chemins de longueurs différentes. Ces fourmis communiquent entre elles via une substance volatile qui s'évapore avec le temps, appelée phéromones. Les fourmis déposent ces phéromones dans leurs chemins grâce à une glande située dans leur abdomen. Ces phéromones jouent un rôle très important dans la direction des fourmis et dans leur choix du chemin le plus court.

Le modèle expliquant le déroulement de l'expérience est le suivant : Une fourmi parcourt au hasard l'environnement autour de la colonie. Si elle découvre une source de nourriture, elle rentre directement à son nid en laissant sur son chemin des phéromones. Ces phéromones attirent les autres fourmis qui vont avoir tendance à suivre cette piste. Les premières fourmis qui reviennent avec de la nourriture au nid sont celles qui ont emprunté le chemin le plus court. Le chemin le plus court sera donc de plus en plus renforcé. Le chemin le plus long finira par disparaître vu l'évaporation des phéromones. A terme, les fourmis choisissent le chemin le plus court reliant leurs points de départ (le nid) à leurs sites de destination (la nourriture).

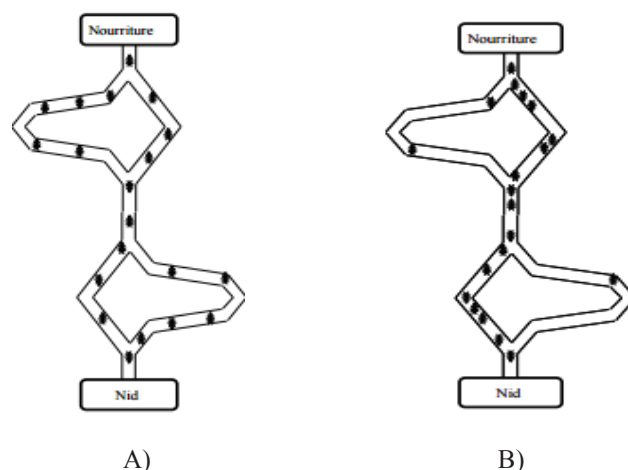


Figure 28. Expérience du double pont binaire pour la sélection des branches les plus courtes par une colonie de fourmis : A) Début de l'expérience, B) Fin de l'expérience [147]

4.2.3.3 Algorithme de colonies de fourmis appliqué au problème du voyageur de commerce

Dans cette section, nous nous sommes basés sur les travaux de Dorigo et al. [150] [151] [152]. Nous adoptons les notations mathématiques utilisées dans [151].

Le problème du voyageur de commerce (*Travelling Salesman Problem*) (TSP) a fait l'objet de la première implémentation de l'algorithme de base de colonies de fourmis le (*Ant System*) (*AS*) [152]. Le problème TSP consiste à un voyageur de commerce de trouver le trajet le plus court reliant n villes, chaque ville ne sera visité qu'une fois. En d'autres termes, il s'agit de trouver un cycle hamiltonien (circuit qui passe exactement une fois par chaque sommet du graphe) de longueur minimale dans le graphe $G = (X, U)$, où :

- X représente l'ensemble de nœuds associés à des villes ;
- $U = \{(i, j) \mid i, j \in X\}$ ensemble d'arcs reliant les nœuds de X ;
- Un ensemble de constantes d_{ij} représentant la longueur de chaque arc $(i, j) \in U$, c'est-à-dire la distance séparant la ville i de la ville j (avec $i, j \in X$).

Algorithme de base. Dans l'algorithme 5 « *Ant System* » ci-après, à chaque itération t ($1 < t < t_{\max}$), chaque fourmi choisit aléatoirement un sommet de départ dans le graphe G , par exemple, la ville i . Ensuite, chaque fourmi k choisit un nouveau sommet j . Le trajet entre i et j dépend de :

- La quantité de phéromones déposées par les fourmis entre les villes i et j , appelé l'intensité de la piste. Cette quantité évolue à chaque passage d'une fourmi.
- La liste des villes déjà visitées, quand la fourmi k est sur la ville i .
- La visibilité qui est l'inverse de la distance entre les villes. Elle sert à aider les fourmis à choisir les villes les plus proches et éviter les villes lointaines.

Règle de transition. La règle de déplacement de la ville i vers la ville j appelée aussi « règle aléatoire de transition proportionnelle » par [153], est décrite formellement par la probabilité suivante :

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot \eta_{ij}^\beta}{\sum_{l \in N_i^k(t)} [\tau_{il}(t)]^\alpha \cdot \eta_{il}^\beta} & \text{si } j \in N_i^k \\ 0 & \text{si non} \end{cases} \quad (1)$$

Où N_i^k est l'ensemble des villes que la fourmi n'a pas encore visité, τ_{ij} est la quantité de phéromones et η_{ij}^β est la visibilité, α et β sont deux paramètres qui contrôlent respectivement l'importance relative entre les phéromones et la visibilité. Si α est égal à 0, le choix du chemin sera seulement fonction de la visibilité. Au contraire, si β est égale à 0, seule la quantité de phéromones sera prise en compte dans le choix du chemin.

Une fois la tournée des villes effectuée, chaque fourmi dépose une certaine quantité de phéromones sur l'arc (i, j) , $\Delta\tau_{ij}^k(t)$, dans l'intervalle de temps $[t, t+n]$. Cette quantité dépend de la solution trouvée.

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{Q}{L^k(t)} & \text{si } (i, j) \in T^k(t) \\ 0 & \text{si } (i, j) \notin T^k(t) \end{cases} \quad (2)$$

Où T^k est le trajet effectué par la fourmi k à l'itération t , $L^k(t)$ la longueur parcourue et Q est une constante.

Mise à jour des phéromones. A la fin de l'itération, la mise à jour des phéromones se fait selon la règle suivante :

$$\tau_{ij}^k(t+n) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij}(t) \quad (3)$$

Où $\rho \in [0, 1[$ représente la vitesse d'évaporation des phéromones entre l'instant t et l'instant $t+n$, $\tau_{ij}(t)$ est la quantité de phéromones déposée entre les villes i et j à l'instant t . En augmentant la valeur de ρ ($\rho \approx 1$), les phéromones stagnent encore plus au niveau de l'arc (i, j) . La recherche sera intensifiée sur ce trajet, ce qui implique des inconvénients tels que le fait d'avoir des mauvaises solutions persiste. En diminuant la valeur de ρ ($\rho \approx 0$), la quantité de phéromones s'évapore très vite et le choix de la ville va dépendre seulement de la visibilité.

Algorithme 5. Algorithme de fourmis de base : le « Ant System »

Pour $t = 1, \dots, t_{\max}$
 Pour chaque fourmi $k = 1, \dots, m$
 Choisir une ville au hasard
 Pour chaque ville non visitée i
 Choisir une ville j , dans la liste N_i^k des villes restantes, selon la formule (1)
 Fin Pour
 Déposer une piste $\Delta\tau_{ij}^k(t)$ sur le trajet $T^k(t)$ conformément à l'équation (2)
 Fin Pour
 Evaporer les pistes selon la formule (3)
Fin Pour

4.2.3.4 Intensification versus diversification

Le défi levé par les algorithmes de colonies de fourmis comme dans d'autres méta-heuristiques est de trouver un compromis entre l'intensification de la recherche autour des composants offrant de bonnes solutions et la diversification de la recherche en explorant un nouvel espace de recherche [153].

Intensification dans l'algorithme ACO «

Ant Colony Optimisation » : L'intensification se fait par l'intermédiaire des phéromones. Plus la quantité de phéromones est importante entre les sommets i et j , plus le trajet est intensifié, et plus la probabilité du choix du composant j est importante. Cette intensification peut être modulée via le réglage des paramètres α et ρ . Rappelons que α définit le poids du facteur phéromonal dans le choix du chemin et ρ définit le taux d'évaporation des phéromones. En augmentant la valeur de α , l'intensification sera importante. Plus, on augmente la valeur de ρ , plus on accentue l'influence de dépôts de phéromones sur le choix du chemin, et donc on intensifie la recherche aux alentours des voies les plus prometteuses.

Diversification dans l'algorithme ACO : L'intensification dans l'algorithme ACO rend la stratégie de choix des combinaisons dans l'algorithme limitée à certains composants où la quantité de phéromones est importante. On risque ainsi d'observer un problème de stagnation des phéromones et de rendre les fourmis incapables d'explorer de nouvelles zones dans l'espace de recherche. D'où l'intérêt de la diversification.

4.2.4 Les réseaux bayésiens

4.2.4.1 Définition

Un réseau bayésien (RB) (*Belief Network* ou réseau causal ou diagramme d'influence), initié par Judea Pearl dans les années 1980, est un modèle graphique probabiliste représenté par un graphe orienté acyclique (DAG). Il constitue une des techniques les plus puissantes en IA et en apprentissage automatique. Basés sur un formalisme probabiliste, les réseaux

bayésiens forment une machine puissante pour le calcul des probabilités conditionnelles. Ils permettent d'inférer les probabilités en fonction d'autres valeurs observées. Les réseaux bayésiens permettent par exemple d'inférer la probabilité de la survenue d'une anomalie en fonction de la connaissance de certains symptômes. Les réseaux bayésiens sont utilisés dans plusieurs domaines d'application : santé, industrie, finance, marketing. Historiquement, les premières applications des réseaux bayésiens étaient développées dans le domaine de la santé et, plus spécifiquement pour le diagnostic médical. Le système Pathfinder a été conçu en 1990 pour le diagnostic histopathologique, fondé sur l'analyse des biopsies. La NASA a développé également une première application non médicale qui est purement basée sur un formalisme bayésien, nommé système VISTA [154]. Ce dernier permet l'aide à la décision en temps réel pour le suivi des moteurs de positionnement orbital d'une navette spatiale. Selon [155], quatre aspects fondamentaux rendent les réseaux bayésiens privilégiés vis-à-vis d'autres modèles tels que : le modèle logique, les systèmes experts, les réseaux de neurones, etc. Ces aspects sont : (1) Acquisition des connaissances, (2) Représentation des connaissances, (3) Utilisation des connaissances, (4) Qualité de l'offre en matière de logiciel.

4.2.4.2 Notation graphique

Un Réseau Bayésien (RB) est un graphe orienté acyclique (DAG) $G = (U, V)$, où U est l'ensemble des nœuds et V l'ensemble des arcs. Les arcs dans le RB représentent les relations de dépendances reliant les différents nœuds du DAG. Chaque nœud dans le graphe représente une variable aléatoire. Un exemple de réseau bayésien est donné Figure 29.

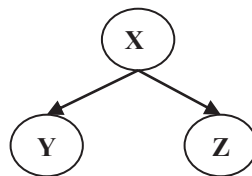


Figure 29. Exemple de réseau bayésien

Chaque nœud enfant, Y et Z dans cet exemple, contient une description de la variable associée sous forme d'une table de probabilité.

Exemple. Supposons que : X : Pluie, Y : Prendre un parapluie, Z : Chemin mouillé

Ces nœuds contiennent des tables de probabilités (Figure 30) qui reflètent la relation de dépendance entre les différentes variables du réseau. On trouve par exemple dans le nœud X : $P(\text{Pluie} = \text{oui}) = 30\%$, $P(\text{Pluie} = \text{non}) = 70\%$, et dans le nœud Y : $P(\text{Prendre un parapluie} = \text{oui} \mid \text{pluie} = \text{oui}) = 100\%$, $P(\text{Prendre un parapluie} = \text{oui} \mid \text{pluie} = \text{non}) = 50\%$, $P(\text{Prendre un parapluie} = \text{non} \mid \text{pluie} = \text{oui}) = 0\%$, $P(\text{Prendre un parapluie} = \text{non} \mid \text{pluie} = \text{non}) = 50\%$. Les arcs reliant la variable X aux deux variables Y et Z montrent que la même cause « pluie » produit les deux conséquences « prendre un parapluie » et « Chemin mouillé »

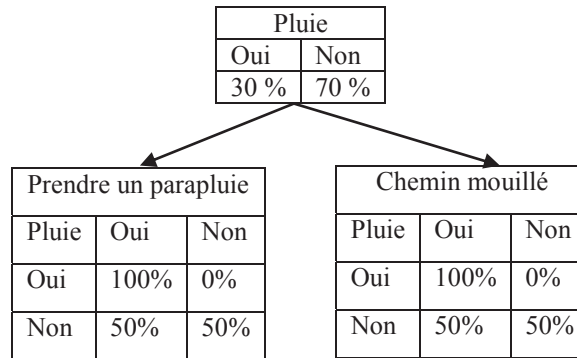


Figure 30. Exemple de tables de probabilités d'un Réseau Bayésien

4.2.4.3 Distribution de probabilité conjointe

Un réseau bayésien peut être défini par une probabilité conjointe sur les variables aléatoires modélisées dans le réseau. Soit G un réseau bayésien avec X_i l'ensemble des variables du réseau. Chaque nœud du réseau bayésien G est associé à la distribution de probabilité suivante :

$$P(X_i | pa(X_i)), \text{ où } pa(X_i) \text{ représentent les parents directs du nœud } X_i$$

La probabilité conjointe est le produit des probabilités conditionnelles sur les variables du réseau bayésien. Elle est décrite par la formule suivante [142] :

$$P(X_1, X_2, \dots, X_n) = \prod_i P(X_i | pa(X_i))$$

Appliquons cette formule sur l'exemple, Figure 29, la probabilité conjointe est égale :

$$P(X, Y, Z) = P(X) \cdot P(Y|X) \cdot P(Z|X)$$

4.2.4.4 Indépendance conditionnelle

Contrairement aux autres Systèmes d'Aide à la Décision tels que les systèmes experts, les arbres de décision, les RBs permettent d'explicitier des relations probabilistes, non déterministes de cause à effet entre les différentes variables aléatoires. Reprenons l'exemple décrit dans la Figure 29, la connaissance de la variable X n'entraîne pas systématiquement l'effet Y mais change la probabilité d'observation de Y .

Le critère d-séparation (le d signifie directionnel) : Considérons X , Y et Z trois ensembles de nœuds dans un graphe de réseau bayésien G . Z d -sépare X et Y signifie que Z bloque la circulation de l'information de X vers Y . On dit que X et Y sont indépendantes conditionnellement, et donc, que la connaissance de X n'a pas d'influence sur Y .

Il existe trois types de connexions sur lesquelles s'appuie la définition du critère d-séparation : La connexion en série, la connexion convergente et la connexion divergente (Figure 31).

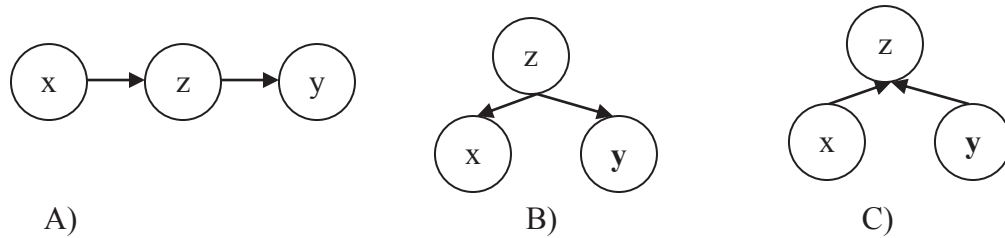


Figure 31. Trois types de connexion dans un réseau bayésien :
 A) Connexion en série ; B) Connexion divergente ; C) Connexion convergente

Dans les connexions en série et divergente, l'information circule entre x et y uniquement lorsque z n'est pas instancié, c'est-à-dire qu'on n'affecte pas une valeur à z. Au contraire, dans la connexion convergente l'information circule entre x et y lorsque z ou l'un de ses descendants est connu.

Définition. Dans un réseau causal, les 2 nœuds x et y sont dits d-séparés si sur toute chaîne entre x et y, il existe un nœud intermédiaire z telle que l'une des deux conditions suivantes soit réalisée :

- La connexion en z est en série ou divergente et z est instancié.
- La connexion en z est convergente et l'on ne dispose pas d'informations élémentaires ni sur z ni sur ses descendants.

Si on conditionne z dans le type de connexion convergente ou série, ceci bloque l'information circulant de x à y. Dans le cas d'une convergence, les deux causes (x et y) amènent au même effet (z), et donc x et y restent indépendantes tant que leur effet commun n'est pas connu.

Dans l'exemple Figure 31, si on connaît s'il pleut ou non (X), alors les deux variables 'prendre la parapluie' (Y) et 'Chemin mouillé' (Z) deviennent indépendantes. On dit que X d-sépare Y et Z.

4.2.4.5 Inférence dans un réseau bayésien : l'arbre de jonction

Le terme inférence désigne de manière générale la détermination et l'interprétation des observations étant donné des données a priori, des prémisses. L'inférence bayésienne permet de déduire la probabilité d'une hypothèse à partir d'autres observations connues, des connaissances a priori. Elle s'appuie sur la formule de Bayes :

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Etant donné deux événements A et B, le théorème de Bayes permet de calculer la probabilité de A sachant B si on connaît la probabilité de A, B et de B sachant A. Effectuer une inférence dans un réseau bayésien est NP-difficile [157].

L'algorithme de l'arbre de jonction ou JLO, du nom de ses auteurs F.V. Jensen, S.L. Lauritzen et K.G. Olesen [158], [159] est un algorithme d'apprentissage automatique qui permet l'inférence bayésienne dans un réseau bayésien à valeurs discrètes. Il se déroule en deux phases :

La phase de construction consiste à factoriser le graphe initial et à le transformer en un arbre de jonction, dont les nœuds représentent des cliques de nœuds du graphe d'origine. Cette transformation est nécessaire dans la mesure où elle permet de supprimer les boucles du graphe et d'optimiser ainsi le processus d'inférence en termes de temps de calcul. L'algorithme de construction de l'arbre est simple. Il se réalise en trois étapes :

- (1) **La moralisation** : Il s'agit de marier les parents du même nœud du graphe et de transformer les arcs dirigés en arcs non dirigés. Ainsi, on passe d'un graphe orienté à un graphe non orienté.
- (2) **La triangulation** : Un graphe non orienté est triangulé si tous les cycles de plus de 3 nœuds ont une corde, c'est-à-dire une arête reliant deux sommets non adjacents.
- (3) **La construction de l'arbre de jonction** : La création d'un arbre couvrant minimal, est appelé arbre de jonction.

La phase de propagation est la phase de calcul probabiliste, c'est-à-dire de l'inférence bayésienne. Elle consiste à propager les valeurs observées dans le réseau de manière à mettre à jour la distribution des probabilités de chaque nœud du réseau. Ceci se fait par un passage des messages entre les différents nœuds de l'arbre de jonction précédemment créé et le calcul probabiliste des nœuds du réseau est calculé en fonction des nouvelles observations. Ceci se traduit par le calcul de $P(Z|\gamma)$ sachant que Z représente l'ensemble des variables du réseau bayésien et γ est l'ensemble des nouvelles informations sur les dites variables.

Exemple de propagation Dans l'exemple précédent présenté Figure 30, si on introduit l'observation « Chemin mouillé » par exemple, cette information sera propagée dans le RB et les tables de probabilités seront donc mises à jour. Après propagation, la nouvelle distribution de probabilités des nœuds X (Pluie) et Z (Chemin mouillé) par exemple devient comme suit :

Pluie	
Oui	Non
30 %	70 %

→

Chemin mouillé	
Oui	Non
80%	20%

Après propagation

Pluie	
Oui	Non
42, 85 %	57, 15 %

→

Chemin mouillé	
Oui	Non
100%	0%

Le calcul de la probabilité conditionnelle se fait en appliquant la règle de Bayes :

$$\begin{aligned}
 P(\text{Pluie}|\text{Chemin mouillé}) &= \frac{P(\text{Chemin mouillé}|\text{Pluie}) \times P(\text{Pluie})}{P(\text{Chemin mouillé})} \\
 &= \frac{1 \times 0,3}{1 \times 0,3 + 0,5 + 0,8} = \mathbf{0,4285}
 \end{aligned}$$

4.2.5 Les réseaux bayésiens dynamiques

Bien que les réseaux bayésiens classiques se révèlent des outils graphiques très utiles permettant une représentation intuitive et claire des connaissances, ils ne modélisent pas les relations de dépendances temporelles entre les variables. La notion du temps n'intervient pas dans les réseaux bayésiens statiques. Mais pour interpréter ou bien prédire par exemple l'état d'un patient à un instant t , il est très important de suivre l'évolution de ses paramètres médicaux dans le temps. Les réseaux bayésiens dynamiques représentent sous forme graphique les relations d'indépendance conditionnelles entre des processus stochastiques. Un réseau bayésien dynamique étend la notion du réseau bayésien classique. Il modélise une chaîne de réseaux bayésiens classiques dont les nœuds sont reliés par des liens causaux temporels, en fonction de pas temporels. En d'autres termes, il permet de représenter l'évolution des variables aléatoires dans le temps.

4.2.5.1 Définition

Un réseau bayésien dynamique (RBD) est défini par le couple (B_1, B_{\rightarrow}) où B_1 est un réseau bayésien représentant la loi a priori des variables aléatoires à l'instant initial 0. B_{\rightarrow} est un réseau bayésien dynamique à deux pas de temps (2TBN) [160]. Il décrit la transition du pas de temps $t-1$ au pas de temps t , c'est-à-dire $P(Z_t|Z_{t-1})$ au moyen d'un graphe orienté acyclique (Z_t, E) où $\{Z_t^1, \dots, Z_t^N\}$ est l'ensemble des variables aléatoires représentant les nœuds du réseau bayésien et E est l'ensemble des arcs les reliant. La probabilité conjointe à deux pas de temps donne la représentation factorisée suivante :

$$P(Z_t | Z_{t-1}) = \prod_{i=1}^N P(Z_t^i | Pa(Z_t^i))$$

où Pa est l'ensemble des parents de Z_t qui peuvent être des parents du réseau au temps t ou bien des parents au temps $t-1$.

Les Réseaux Bayésien Dynamiques sont basés sur la propriété de Markov, qui exprime qu'étant donné les états passés et l'état présent, la distribution conditionnelle de probabilité des états futurs ne dépend en fait que de l'état présent et non pas des états passés. Les réseaux bayésiens dynamiques sont une généralisation d'autres modèles comme les filtres de Kalman ou les modèles de Markov cachés HMM (*Hidden Model Markov*).

4.2.5.2 Hypothèse Markovienne

Les réseaux bayésiens dynamiques reposent sur la propriété du Markov. Un processus est dit markovien si son état d'évolution ne dépend pas de son passé mais de son état présent. Il existe deux types de modèles de Markov, le modèle de Markov observable et le modèle de Markov caché (HMM). Pour présenter ces deux modèles, nous nous appuyons sur les travaux de [161], [162], [163]. Nous adoptons les notations mathématiques utilisés dans [161].

Modèle de Markov observable : Dans ce modèle, la chaîne de Markov est représentée par un graphe de transition d'états (Figure 32). Les états forment les sommets du graphe, appelés états de la chaîne de Markov, $S = \{S_1, S_2, \dots, S_N\}$, où N est le nombre des états, $N = \sum S$. Les transitions entre les états sont représentées par des flèches reliant les états. Ces transitions sont pondérées par des probabilités regroupées dans une matrice, appelée matrice de transition : $A = \{a_{ij} = P(S_j | S_i)\}$; $\sum_{j=1}^N a_{ij} = 1$. Un vecteur d'initialisation regroupe l'ensemble des probabilités de départ qui découlent de l'état initial de la chaîne, le point 0. Ce vecteur est noté par :

$\Pi = \{\pi_i = P(S_i)\}$; $\sum_{i=1}^N \pi_i = 1$. Un tel modèle de Markov λ est appelé modèle observable puisque tous ses états sont observables. Il est représenté par le couple : vecteur d'initiation Π et matrice de transition A : $\lambda = \{\Pi, A\}$.

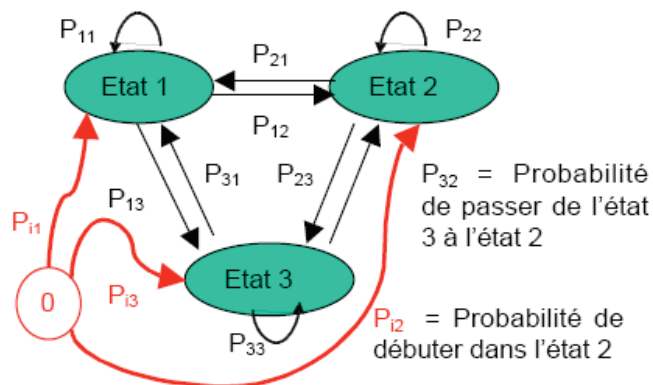


Figure 32. Graphe d'un modèle de Markov observable [164]

Modèle de Markov caché : Dans ce type modèle, les états $S = \{S_1, S_2, \dots, S_N\}$ sont non observables mais émettent des signaux observables $O = \{O_1, O_2, \dots, O_K\}$ qui sont pondérés par leurs probabilités. Outre la matrice de transition A et le vecteur d'initialisation Π , un modèle de Markov caché λ est caractérisé par une matrice qui regroupe les probabilités que l'état S_i émette le signal d'observation O_K . Cette matrice est appelée matrice d'émission et est notée par :

$$B = \{b_i(O_K) = P(O_K | S_i)\}; \sum_{j=1}^T b_i(o_j) = 1$$

Le modèle de Markov caché λ est ainsi représenté par le triplet Π (vecteur d'initialisation), A (matrice de transition), et B (matrice d'émission). On note : $\lambda = \{\Pi, A, B\}$.

Pour mieux appréhender les modèles de Markov cachés, nous allons présenter un exemple (Figure 33) extrait de [164].

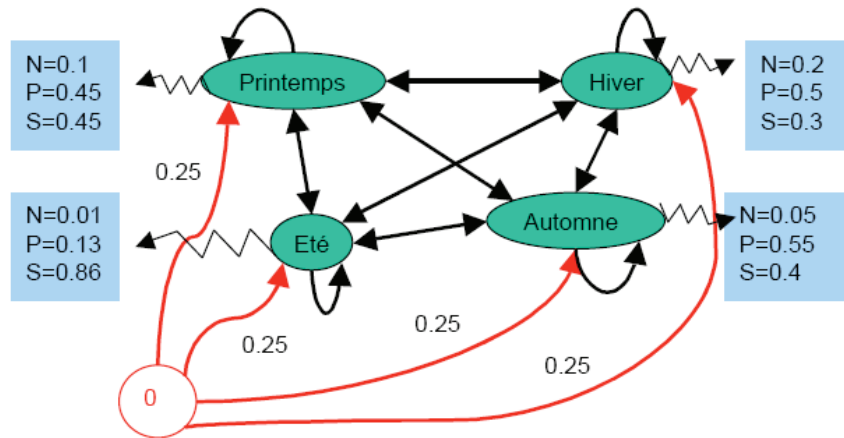


Figure 33. Exemple de modèle de Markov caché (HMM) [164]

Ce modèle représente 4 états de saisons {Printemps (P), Hiver (H), Automne (A), Été (E)}. Chaque état est non observable mais il émet des signaux observables qui représentent ici les temps de la journée {Nuage (N), Pluie (P), Soleil (S)}. Selon [161], le Modèle de Markov caché permet de résoudre 3 problèmes fondamentaux : la reconnaissance, l'analyse et l'apprentissage.

Etant donné la séquence d'observation $O = \{O_1, O_2, \dots, O_n\}$ et le modèle de Markov caché $\lambda = \{\Pi, A, B\}$.

- La reconnaissance : Quelle est la probabilité d'observer la séquence O , par exemple $O = \{\text{Nuage (N), Nuage (N), Pluie (P)}\}$, sachant le modèle $\lambda : P(O | \lambda)$.
- L'analyse : Quelle est la séquence des états qui maximise la probabilité d'avoir généré O , par exemple $O = \{\text{Soleil (S), Pluie (P), Nuage (N)}\}$
- Apprentissage : Comment ajuster les paramètres du modèle λ pour maximiser la vraisemblance de l'ensemble d'apprentissage $P(O | \lambda)$.

Le calcul de ces probabilités est effectué par des algorithmes d'inférence, décrits ci-après.

4.2.5.3 Inférence dans un réseau bayésien dynamique

L'inférence dans un réseau bayésien dynamique consiste à rechercher la distribution de probabilités sur les variables cachées du modèle à un instant donné, sachant la séquence des observations passées et éventuellement futures.

La loi de probabilité conjointe dans un RBD se calcule en « déroulant » le réseau sur une séquence temporelle de longueur T . On peut écrire :

$$P(Z_{1:T}) = \prod_{t=1}^T \prod_{i=1}^N P(Z_t^i | Pa(Z_t^i))$$

Il est possible d'appliquer l'algorithme de jonction pour l'inférence dans un RBD en le considérant comme un RB classique. Cependant, comme la topologie du RBD est une structure répétitive, le nombre de variables sera multiplié par la longueur de la séquence T. Ainsi, la construction de l'arbre de jonction sera donc très lourde et le processus d'inférence devient lent. D'autres algorithmes d'inférence sont ainsi utilisés comme l'algorithme Forward-Backward [165], [166], l'algorithme de Pearl, l'algorithme de l'interface. Nous explicitons dans ce qui suit l'algorithme Forward-Backward.

Forward-Backward : Etant donné un modèle de Markov caché, l'algorithme Forward-Backward permet de calculer $P(O | \lambda)$, la vraisemblance que le modèle génère la séquence observée O. Le calcul se fait en deux étapes principales : étape de calcul progressif qui se fait par passage du message en avant (Forward) et étape de calcul progressif (Backward) qui se fait par passage du message en arrière. $P(O | \lambda)$ peut être défini à chaque instant $t \in [1, T]$ par :

$$P(O | \lambda) = \sum_{i=1}^N \alpha_t(i) \times \beta_t(i)$$

Où :

- $\alpha_t(i)$ est la probabilité d'émettre la suite d'observations $\{O_1, O_2, \dots, O_t\}$ et de se trouver à l'état q_t à l'instant t (phase Forward) : $\alpha_t(i) = P(O, q_t = S_i | \lambda)$
- $\beta_t(i)$ est la probabilité d'émettre la suite $\{O_{t+1}, O_{t+2}, \dots, O_T\}$ en partant de l'état q_t au temps t (phase Backward) : $\beta_t(i) = P(O | q_t = S_i, \lambda)$

Phase Forward : Le calcul de la probabilité $\alpha_t(i)$ se fait via l'algorithme Forward suivant :

Algorithme 6. Algorithme Forward

Initialisation : $\alpha_1(i) = \pi_i \times b_i(O_1)$

Itération : $\alpha_{t+1}(j) = [\sum_{i=1}^N \alpha_t(i) a_{ij}] \times b_j(O_{t+1})$, pour $t \in [1, T-1], j \in [1, N]$

Terminaison : $P(O | \lambda) = \sum_{i=1}^N \alpha_T(i)$

Reprenons l'exemple de HMM de [164] présenté dans la Figure 33. Le calcul de $\alpha_1(i)$ à l'étape d'initialisation s'effectue comme suit :

Etant donnée la séquence observée $O = \{\text{Nuage}, \text{Nuage}, \text{pluie}\}$ et la matrice d'émission suivante :

$$B = \begin{matrix} & \begin{matrix} N & P & S \end{matrix} \\ \begin{matrix} P \\ E \\ A \\ H \end{matrix} & \begin{bmatrix} 0.1 & 0.45 & 0.45 \\ 0.01 & 0.13 & 0.86 \\ 0.05 & 0.55 & 0.4 \\ 0.2 & 0.5 & 0.3 \end{bmatrix} \end{matrix}$$

Cette matrice indique que par exemple la probabilité d'être au printemps et d'émettre l'observation N (Nuage) est égale à 10%. La probabilité d'être en Été (E) et d'observer la pluie est de 13%.

$$\alpha_1(i) = \pi_i \times b_i(O_1) = \begin{matrix} & \begin{matrix} t=1 & & t=1 \end{matrix} \\ \begin{matrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{matrix} \times \begin{matrix} 0,1 \\ 0.01 \\ 0.05 \\ 0.2 \end{matrix} & = & \begin{matrix} \begin{bmatrix} 0.025 \\ 0.0025 \\ 0.0125 \\ 0.05 \end{bmatrix} \end{matrix} \begin{matrix} P \\ E \\ A \\ H \end{matrix}$$

La deuxième ligne de la matrice, par exemple, indique que la probabilité d'être à l'été et d'avoir une journée nuageuse est de 0,0025.

Phase Backward : Le calcul de la probabilité $\beta_T(i)$ se fait via l'algorithme Backward décrit ci-après.

Algorithme 7. Algorithme Backward

Initialisation : $\beta_T(i) = 1, 1 \leq i \leq N$

Itération : $\beta_T(i) = \sum_{j=1}^N a_{ij} b_j(O_{t+1}) \beta_{t+1}(j),$

Pour tout $t \in [T - 1, 1], i \in [1, N]$

4.2.5.4 Exemples d'application des réseaux bayésiens dynamiques

La Figure 34 illustre un exemple de réseau bayésien dynamique utilisé pour détecter des changements dans l'état d'hydratation d'un patient sous dialyse péritonéale à domicile. Le projet est appelé Diatic™v3 [167]. Le modèle Diatic™v3 reflète un processus de monitoring répétitif. Le taux d'hydratation d'un jour a , de façon triviale, une influence directe sur le taux d'hydratation du lendemain. Les relations de dépendances entre les variables du réseau permettent de propager les observations passées sur l'état courant du patient.

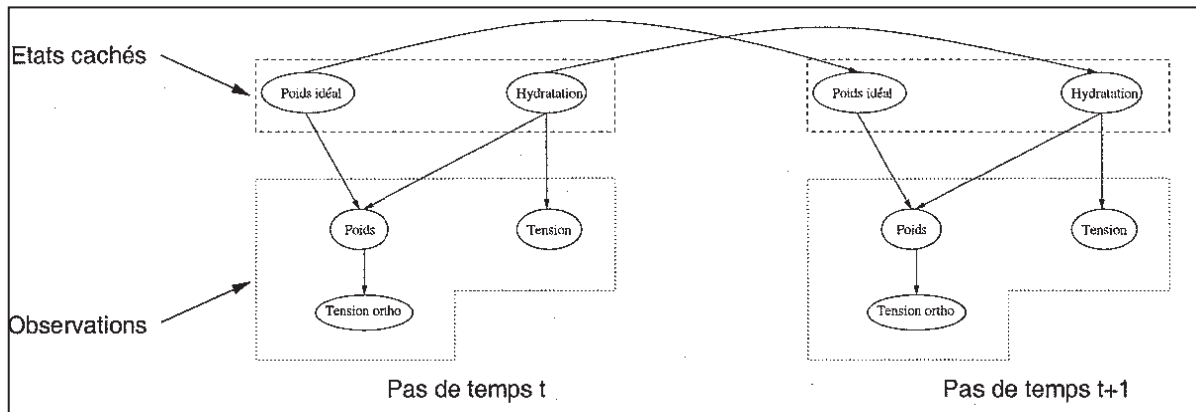


Figure 34. Modèle d'évolution du système DiatelicTMv3.

Les nœuds représentant *les états cachés* sont reliés d'un pas de temps à l'autre pour modéliser la relation markovienne d'ordre un qui existe entre eux [167].

La modélisation stochastique et plus particulièrement les RBDs sont également présents dans d'autres types d'applications de raisonnement médical en télémédecine, à savoir la segmentation d'électrocardiogrammes, la classification pour la prévention des complications vasculaires en hémodialyse [168].

4.3 Modélisation des systèmes dynamiques

4.3.1 Les réseaux de Petri

Le réseau de Petri est un modèle mathématique permettant la modélisation du comportement des systèmes dynamiques à événements discrets comme les protocoles de communication, les chaînes de production, les réseaux de transport, etc. [169]. Apparue en 1962 dans la thèse de Carl Adam Petri, les réseaux de Petri constituent un outil très puissant pour la modélisation des propriétés telles que la synchronisation, le parallélisme, l'allocation de ressources, la gestion de conflits, la mutuelle exclusion, etc., [170], [171].

Un réseau de Petri (RdP) R est représenté par le quadruplet suivant (Nous adoptons la notation [172]) :

$$R = \langle P, T, Pre, Post \rangle$$

Où :

- P est un ensemble fini de places : $P = \{p_1, \dots, p_n\}$.
- T est un ensemble finis de transitions : $T = \{T_1, \dots, T_n\}$.
- $Pre : T \times P \rightarrow N$ est l'application places précédentes
- $Post : T \times P \rightarrow N$ est l'application places suivantes

4.3.1.1 Sémantique des places, des transitions et des jetons

Une transition dans un RdP représente un événement (transport d'une marchandise, mise à jour d'une base de données, envoi d'un fichier, etc.). Une marque (appelée aussi jeton) peut jouer le rôle d'un objet physique (produit, personne, etc.), d'un objet informatif (signal, message, etc.), d'une collection d'objets, d'un indicateur de l'état d'un objet (imprimante en veille, etc.).

Un réseau de Petri est un moyen pour décrire des relations existantes entre des conditions et des événements.

4.3.1.2 Représentation graphique et notation matricielle

Un RdP est composé d'un ensemble de places représentées par des cercles, des transitions représentées par des barres et des jetons (ou marques) contenus dans des places. Les jetons représentent généralement les ressources disponibles (Figure 35). Les places et les transitions sont reliées par des arcs orientés.

4.3.1.3 Le marquage

Le marquage $M(p)$ correspond à la distribution des jetons dans la place p . Le marquage représente les conditions et les ressources nécessaires pour le déclenchement d'un événement. Un marquage M peut être représenté par une matrice dont le nombre de lignes correspond aux places et les colonnes correspondent aux transitions. La Figure 35 suivante illustre un exemple de réseau de Petri avec le marquage correspondant.

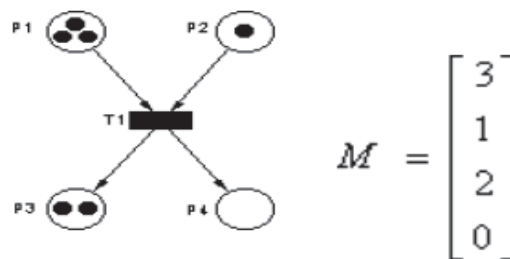


Figure 35. Exemple de marquage dans un RdP

4.3.1.4 Franchissement d'une transition

Le franchissement des transitions et l'évolution du marquage illustrent bien la modélisation dynamique fournie par le RdP. Une transition est dite franchissable ou tirable si et seulement si toute place prédécesseur contient au moins un jeton :

$$\forall p \in P \quad M(p) \geq Pre(p, t)$$

Le tir d'une transition correspond à enlever un jeton de chaque place 'amont' et à mettre un jeton à chaque place 'aval'. A l'issue du tir, le marquage M passe à un nouvel état, un marquage M' , tel que :

$$\forall p \in P \quad M'(p) = M(p) - Pre(p, t) + Post(p, t)$$

Si nous considérons l'exemple, Figure 36, nous constatons qu'après le tir de la transition t le jeton s'est déplacé de la place p_1 aux places p_2 et p_3 , d'où l'évolution du marquage initial M_0 au marquage M' .

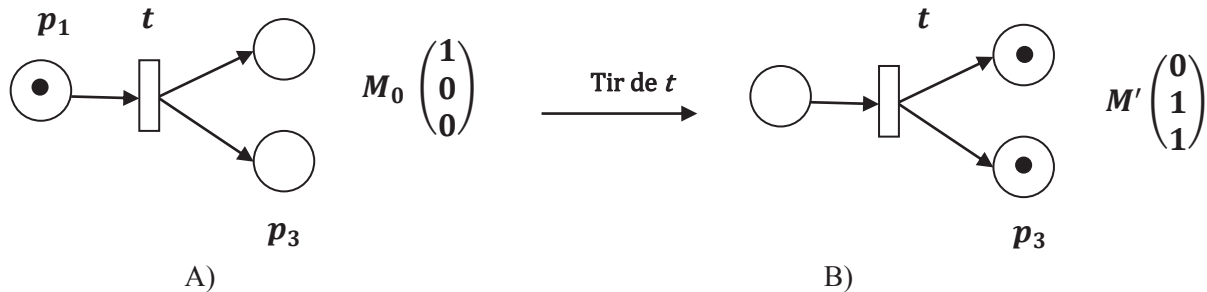


Figure 36. Principe de franchissement d'une transition dans un RdP : A) Avant le tir de t , B) Après le tir de t

4.3.2 Dimension temporelle dans les réseaux de Petri

Les réseaux de Petri, de par leur capacité à représenter la dynamique des systèmes, intègrent d'une manière explicite la dimension temporelle. Il existe deux familles d'extension temporelle des réseaux de Petri : Les réseaux de Petri temporisés introduits par Ramchandani [173] et les réseaux de Petri temporels introduits par Merlin [174].

4.3.2.1 Réseaux de Petri temporisés

Les réseaux de Petri temporisés constituent une technique puissante pour l'évaluation de la performance et pour la modélisation dynamique des systèmes à contraintes temporelles. La modélisation des réseaux de Petri temporisés permet de répondre à la question : Combien de temps durent les actions ? Il y a deux modes de temporisation : la temporisation des places (P-times nets) [120] et la temporisation des transitions (T-times nets) [173].

4.3.2.1.1 Les réseaux P-temporisés (P-timed nets)

Les réseaux P-temporisés consistent à associer une durée à chaque place du réseau. Une marque doit rester au moins le temps d associé à cette place. Elle reste invalide pendant d unités de temps. Ensuite seulement, la marque devient disponible et pourra ainsi franchir ses transitions de sorties.

Un réseau de Petri temporisé est défini par (R, D) où

R est un réseau de Petri $R = \langle P, T, Pre, Post \rangle$

$D : P \rightarrow Q^+$ est la fonction durée de franchissement qui correspond à la durée minimale de séjour du jeton dans une place.

4.3.2.1.2 Réseaux de Petri T-temporisés

Les réseaux de Petri T-temporisés consistent à associer une durée d à chaque transition t . La sémantique de cette durée correspond à la durée de tir de la transition t . A la fin du tir, les marques produites par t apparaissent. Cette règle est valide si la transition représente une activité et non un événement instantané. Sifakis dans [175] [176] a montré que les RdP T-temporisés et P-temporisés sont équivalents.

4.3.2.2 Réseaux de Petri temporels

Si les réseaux temporisés traitent la question : combien durent les actions ?, les réseaux temporels traitent la question : Quand se produisent-elles ? Concernant les réseaux de Petri temporels, le temps est exprimé comme paramètre quantifiable et continu associé et non ponctuel comme dans le RdP temporisé. Le principe général consiste à associer un intervalle de temps, une plage horaire, aux transitions [174] ou aux places. A la base, les réseaux de Petri temporels sont introduits par P. Merlin et D. Farber pour l'étude des protocoles de communication.

4.3.2.2.1 Réseaux de Petri T-temporels

Un réseau de Petri t-temporel [174] est un 6-uplet $\langle P, T, Pré, Post, M_0, IS \rangle$ tels que :

- $\langle P, T, Pré, Post, M_0, IS \rangle$ est un réseau de Petri
- $IS : T \rightarrow Q^+ \times Q_+ \cup \{\infty\}$ est la fonction d'intervalle statique, qui à chaque transition associe son intervalle statique, à bornes rationnelles de tir, Q_+ étant l'ensemble des nombres rationnels positifs.

La règle de franchissement est la suivante :

Pour $t \in T$ tel que $IS = [a_i, b_i]$ avec $0 \leq a_i \leq b_i$, t ne peut être franchi qu'entre a_i et b_i . La transition doit nécessairement être franchie si elle a été sensibilisée pendant b_i .

4.3.2.2.2 Réseaux de Petri P-temporels

- Un réseau de Petri P-temporel [177] est un tuple $\langle P, T, Pré, Post, M_0, IS \rangle$ dans lequel : $\langle P, T, Pré, Post, M_0 \rangle$ est un réseau de Petri marqué muni d'un marquage initial M_0 , IS est une fonction qui à chaque place p fait correspondre un intervalle fermé de temps de séjour.
- $IS : P \rightarrow (Q^+ \cup 0) \times (Q^+ \cup \infty)$; Q^+ est l'ensemble des nombres rationnels positifs
- $p_i \rightarrow IS_i = [a, b]$ avec $0 \leq a_i \leq b_i$.

IS est la fonction de temps de séjour d'un jeton dans une place p_i . Une marque dans une place doit quitter sa place p_i pour valider ses transitions de sorties au plus tôt à la date a_i et au plus tard à la date b_i . La règle de franchissement est ainsi décrite comme suit :

Si t_i est sensibilisé à l'instant β :

- t_i ne peut pas être tiré avant $\beta + a_i$
- t_i doit avoir été tiré à l'instant $\beta + b$

4.4 Conclusion

Ce chapitre présente une étude exploratoire de quelques méthodes d'intelligence artificielle dédiées à l'aide à la décision et d'autres méthodes de modélisation des systèmes dynamiques.

Nous avons vu que les arbres de décision et les systèmes experts reposent sur un raisonnement logique et déductif à partir de règles strictes et déterminantes. Au contraire, les colonies de fourmis et les réseaux bayésiens s'appuient sur un formalisme probabiliste. Les arbres de décision permettent l'aide à la décision pour la classification, par exemple, d'une population en groupes homogènes selon des variables discriminantes. Des algorithmes d'apprentissage comme l'algorithme de CART et C4.5 ont été développés pour la construction des arbres de décisions. Un système expert représente un outil puissant pour l'aide à la décision en intelligence artificielle. Il permet de reproduire, grâce au moteur d'inférence, le raisonnement d'un expert à partir de faits et de règles connues pour résoudre des problèmes. Des algorithmes de chaînage avant, arrière ou mixtes sont utilisés pour faire ce raisonnement. Dans ce chapitre, nous avons aussi présenté les algorithmes de colonies de fourmis qui constituent une famille de méta-heuristiques utilisées pour résoudre des problèmes d'optimisation. Nous avons décrit le fonctionnement de cet algorithme et son application sur le problème du voyageur de commerce (TSP). Nous avons présenté les réseaux bayésiens. Ces derniers sont des modèles graphiques probabilistes permettant la représentation des dépendances conditionnelles entre les variables aléatoires en s'appuyant sur le critère d-séparation. L'inférence dans un réseau bayésien consiste à estimer les lois a posteriori des variables du réseau après l'insertion de nouvelles connaissances. Un exemple d'algorithme d'inférence est l'algorithme JLO. Nous avons mis en lumière la prise en compte de la notion « temps » dans les réseaux bayésiens dynamiques. Ces derniers sont considérés à la fois comme un outil d'aide à la décision et de modélisation des systèmes dynamiques. Ils permettent de représenter l'évolution des variables aléatoires dans le temps via des liens causaux temporels reliant les différents nœuds du réseau. Enfin, les réseaux de Petri sont aussi des modèles mathématiques puissants pour la représentation des systèmes dynamiques discrets. Ils permettent la modélisation et la visualisation des comportements parallèles, de la synchronisation et du partage de ressources.

Dans le chapitre 8, nous allons effectuer une étude comparative de ces différentes méthodes eu égard aux besoins d'acheminement des messages en e-santé. Un choix méthodologique sera par la suite fourni dans ce même chapitre.

Partie III
Gestion de l'acheminement des messages
autonome, intelligente et sensible au
contexte

La suite de notre mémoire est dédiée à la présentation de nos principales contributions relatives à la problématique d'acheminement des messages. Après avoir suivi une démarche *bottom-up* dans le chapitre 2 pour déduire les spécifications d'acheminement des messages à partir des scénarios en e-santé, nous adoptons une approche *top-down* qui va d'une vision générique indépendante du domaine que nous présentons dans cette partie III, vers une vision spécifique et applicative au domaine de l'e-santé qui fera l'objet de la partie suivante. Le principe de notre démarche méthodologique est représenté Figure 37.

Notre approche consiste à proposer dans cette partie un modèle de haut niveau et une architecture générique indépendante du domaine de l'e-santé, satisfaisant les besoins et les spécifications d'acheminement des messages que nous avons précédemment déduits dans le chapitre 2. A partir de l'architecture proposée, nous avons procédé à une modélisation BPMN [3] de processus métiers exécutés par les services composants de l'architecture. Dans la partie IV, nous étudierons la problématique d'acheminement appliquée au domaine de l'e-santé. Nous proposerons alors une architecture d'orchestration de services d'acheminement des messages en e-santé et de systèmes d'information. Afin de préserver l'interopérabilité de cette architecture, nous l'avons construite en veillant à intégrer les spécifications du standard international en santé HL7 version 3.

Les chapitres de cette partie exposent en détail notre démarche méthodologique. Le plan de cette partie est agencé comme suit :

Le chapitre 5 présente un modèle de représentation de politiques d'acheminement de messages avec prise en compte du contexte et une architecture pour la gestion autonome, intelligente et auto-adaptative d'acheminement des messages.

Le chapitre 6 présente une vue métier de l'architecture proposée dans le chapitre précédent.

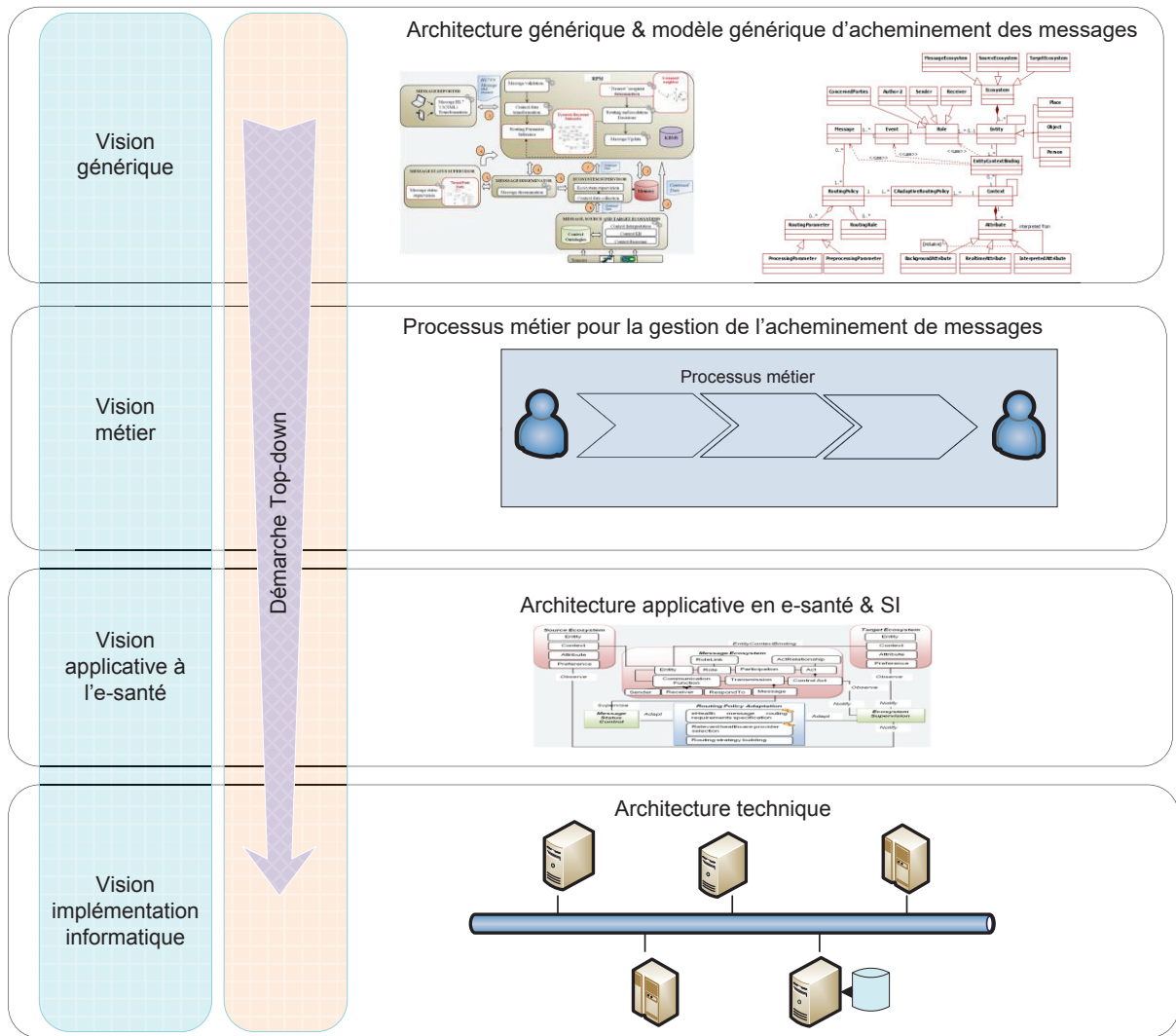


Figure 37. Schéma de principe de notre démarche méthodologique

Chapitre 5

Modèles et architecture pour la gestion autonome, intelligente et auto-adaptative de l'acheminement de messages

5.1 Introduction

Avec l'avènement des systèmes mobiles, les systèmes logiciels dédiés à la gestion d'acheminement des messages doivent être capables de s'adapter à un environnement fluctuant afin d'améliorer au mieux la qualité de transmission des messages. Ces systèmes auto-adaptatifs doivent observer le contexte de l'expéditeur et du destinataire du message et adapter leurs décisions de routage en conséquence par des expériences d'apprentissage. Le point de départ de notre réflexion est d'intégrer les critères d'auto-adaptation et de sensibilité au contexte dès la phase de conception pour éviter tout retour en arrière coûteux au moment du développement. Pour rendre notre système intelligent et capable d'autogérer ses décisions, nous faisons appel à des méthodes d'intelligence artificielle.

Nous présentons dans ce chapitre un modèle qui met en relation les différents éléments impliqués dans la mise en place d'une politique d'acheminement avec les concepts reliés au paradigme « *context aware* ». La deuxième partie de ce chapitre sera consacrée à la conception d'une architecture orientée services pour le pilotage intelligent de l'acheminement de messages. Nous proposons ensuite une modélisation ontologique des systèmes d'information source, cible et message, auxquels une politique d'acheminement est sensible. La dernière partie, quant à elle, sera dédiée à démontrer la prise en compte des besoins dans notre architecture : B1 : Personnalisation, B2 : Sensibilité au contexte, B3 : Auto-Adaptation, B4 : Interopérabilité, B5 : Supervision, B6 : Autonomie, B7 : Sécurité et protection de la vie privée B8 : Intelligence.

5.2 Modèle de représentation de politiques d'acheminement de messages avec prise en compte du contexte multi-dimensionnel

Dans cette section, nous présentons notre modèle de représentation de politiques d'acheminement en relation avec le paradigme « sensibilité au contexte » (Figure 38) [178]. Le diagramme de classe ci-après explicite plusieurs concepts impliqués dans le processus de routage ainsi que les relations entre eux. Il reflète l'aspect multidimensionnel du contexte auquel une stratégie d'acheminement est fortement sensible. Etablir une politique

d'acheminement, c'est définir un ensemble de règles et de paramètres nécessaires pour l'acheminement des messages entre deux environnements distants et hétérogènes : environnements source et cible. Nous distinguons deux types de paramètres :

- Les *Preprocessing Parameters* : Ils forment les besoins requis qui doivent être identifiés avant le traitement des messages comme le profil type du destinataire, le moyen d'acheminement requis (SMS, mail, etc.), etc.
- Les *Processing Parameters* : Ils représentent les paramètres nécessaires pour le contrôle et la supervision du traitement des messages comme les délais requis pour la réception, la lecture et la réponse aux messages.

Nous introduisons le terme « écosystème » pour désigner un ensemble de systèmes d'informations complexes et évolutifs qui sont reliés à une entité au sein d'un environnement donné. Nous considérons l'entité comme un élément logique ou physique représenté par une personne, un objet ou une place et impliqué dans le processus de routage. Nous distinguons trois types d'écosystèmes : source, cible et message. Chacun est composé d'un ensemble d'entités. Une entité peut être reliée à une autre entité. Elle peut être également attachée à différentes catégories de contextes (géographique, climatique, médical, social, etc.) via un lien automatique que nous appelons « *entity context binding* ». Ce lien est une sorte de pont établi en temps réel, durant la collecte des informations contextuelles. Il permet d'identifier, selon chaque scénario, les données pertinentes à collecter. Le choix de ces données dépend du rôle de l'entité (auteur de l'acte, expéditeur, partie concernée, destinataire), de l'événement déclencheur du message et du contenu du message lui-même, d'où le stéréotype « *use* ». Dans un scénario où il y a une forte probabilité de déplacements de personnes par exemple, suite à l'événement déclencheur du message, le contexte environnemental (état du climat, hostilité du lieu, facilité d'accès, etc.) devra être pris en considération. Le *CA adaptiveRoutingPolicy* est une classe qui formalise le mécanisme de sensibilité au contexte (*Context awareness*). Il modélise les adaptations basées sur les informations contextuelles qui peuvent se répercuter sur la politique d'acheminement. Le *CA adaptiveRoutingPolicy* indique quel objet ou service de la politique d'acheminement est sensible à un contexte et à quel contexte. Un contexte regroupe un ensemble d'attributs, appelés aussi des observations dans les systèmes sensibles au contexte.

Nous différencions quatre types d'attributs que nous définissons ci-après. Les attributs **non interprétés** sont des paramètres bruts, utilisables tels quels. Les **attributs interprétés** ou *interpreted attributes* sont calculés en fonction d'autres attributs atomiques, non interprétés. Un exemple d'attribut contextuel interprété est l'état climatique qui peut être déterminé en fonction de la valeur d'autres attributs comme la température, la neige, etc. Les **background attributes** constituent les observations relativement statiques, qui ne changent quasiment pas de valeurs lors du transfert des messages (par exemple la catégorie socio-professionnelle du destinataire/expéditeur). Au contraire, les **real-time attributes** sont dynamiques et peuvent changer de valeur en temps-réel (tel que la localisation géographique). Ils peuvent ainsi générer un processus d'adaptation sur le plan stratégique de l'acheminement. Les attributs *background* et *real time* peuvent être interprétés ou non.

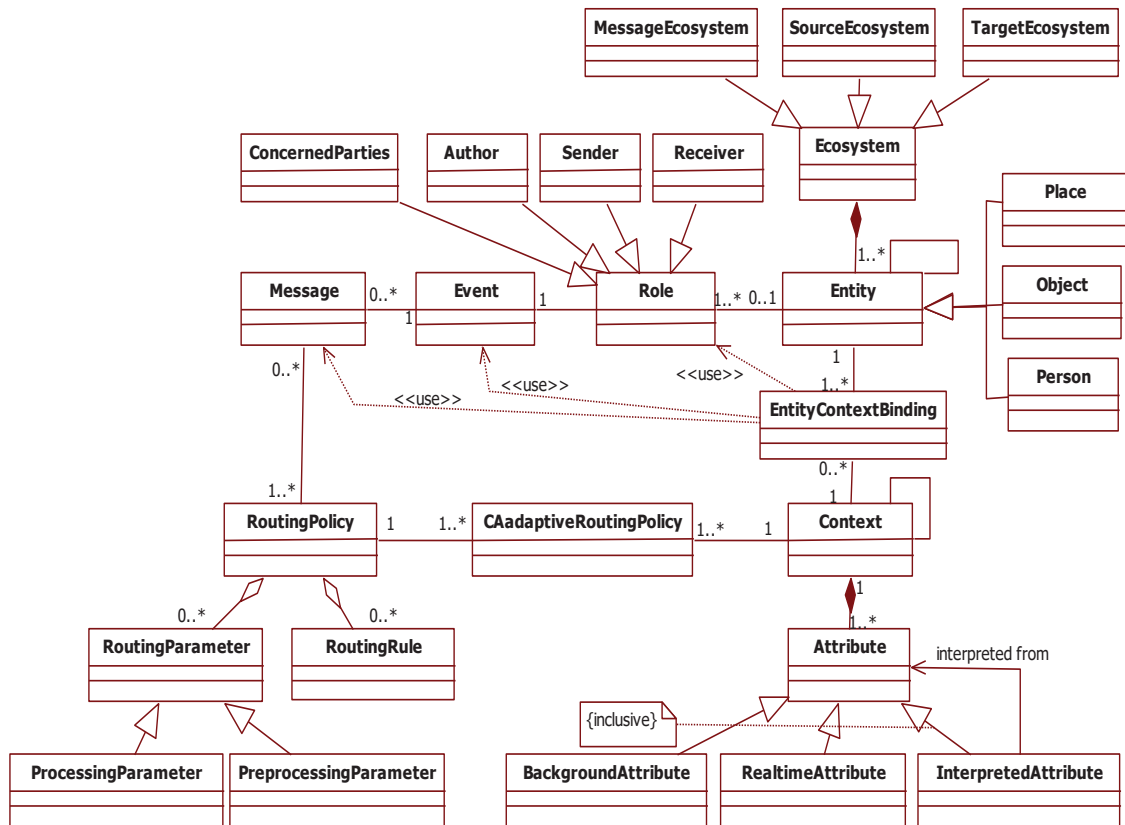


Figure 38. Modèle de représentation de politiques d'acheminement de messages avec prise en compte de contextes

5.3 Architecture orientée services pour la gestion autonome et intelligente de l'acheminement des messages

Cette section sera consacrée à la présentation de notre architecture orientée services pour le pilotage autonome et intelligent des messages (Figure 39) [179]. Lors de la modélisation de cette architecture, deux approches ont été envisagées. Dans la première approche, nous nous sommes basés sur l'ingénierie dirigée par les modèles et sur l'ingénierie des connaissances pour permettre une représentation formelle et sémantique du contexte et un raisonnement logique sur les différentes données contextuelles. La seconde approche que nous envisageons pour garantir un système décisionnel de routage autonome est de remplir deux propriétés particulièrement intéressantes dans les systèmes autonomes : « *self awareness* » and « *self adaptation* ».

5.3.1 La publication du message : « *Message Reporter* »

Le *Message Reporter* est responsable de la publication du message vers le *Routing Policy Manager* (RPM). Le *Message Reporter* constitue la source du message. Il peut être un PC, un Smartphone ou toute autre machine capable de déclencher le transfert d'un message automatiquement ou suite à une intervention humaine.

5.3.2 La dissémination du message : « *Message Disseminator* »

Le *Message Disseminator* reçoit le message du RPM et le transmet aux points terminaux, c'est-à-dire aux destinataires dans leurs dispositifs de communication (smartphone, etc.). Il publie le statut du message (reçu/annulé/accepté par le destinataire) au RPM et notifie le *Message Status Supervisor* lorsque le message est envoyé.

5.3.3 La représentation des connaissances : Ecosystèmes source, cible et message

Nous représentons les données contextuelles des écosystèmes source, cible et message par une ontologie. L'objectif est de formaliser les connaissances et de permettre des déductions logiques sur les informations du contexte à partir d'une base de connaissances et d'une base de règles.

Un moteur d'inférence peut ainsi raisonner sur les informations représentées dans l'ontologie pour déterminer les contextes dérivés, et pour calculer notamment ce que nous appelons les attributs interprétés.

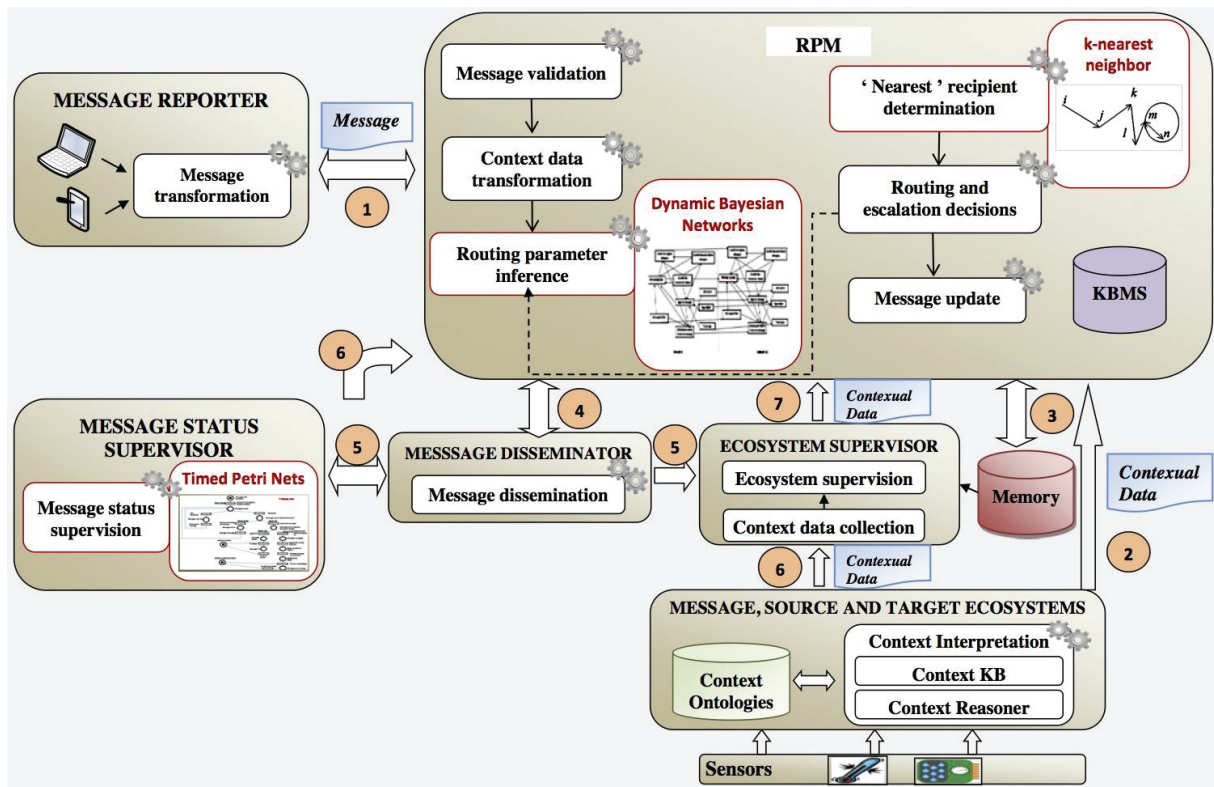


Figure 39. Architecture pour la gestion d'acheminement des messages

5.3.4 La gestion de la politique d'acheminement : *Routing Policy Manager*

Le *Routing Policy Manager* (RPM) forme le noyau de l'architecture. Il est doté d'un processus décisionnel intelligent et auto-adaptatif pour la prise de décision de routage, l'inférence des besoins requis pour l'acheminement des messages (moyen d'acheminement, ressources matérielles, spécialité/expérience du destinataire) et pour le choix de la « bonne »

destination à laquelle le message doit être envoyé. Le RPM est constitué de deux catégories de services décrites ci-après :

5.3.4.1 Services d'analyse

5.3.4.1.1 Validation du message

Après réception du message à partir du *Message Reporter*, le service de validation du message, comme son nom l'indique, analyse la syntaxe et la sémantique du contenu du message et notifie le *Message Reporter* en cas de non validité. L'analyse sémantique implique la vérification de l'existence d'une destination préférée. Si c'est le cas, le service de validation des messages invoque le *Message Disseminator* pour transférer le message au destinataire final. Si non, il invoque le service de transformation des données contextuelles.

5.3.4.1.2 Transformation des données contextuelles

Ce service invoque le composant « *Ecosystem Supervisor* » pour collecter des données contextuelles pertinentes, sur la base des modèles de sensibilité des paramètres d'acheminement au contexte, pour la définition d'une politique d'acheminement et transforme ces données pour qu'elles soient acceptables et exploitables par le service « *routing parameters inference* ».

5.3.4.2 Services auto-adaptatifs

Un système de routage autonome doit être capable de comprendre les informations contextuelles et d'adapter dynamiquement et pro-activement son comportement décisionnel en fonction. Cette propriété est connue sous le terme « *self adaptive* » ou « *auto-adaptation* » dans les systèmes autonomes. Dans notre système, l'adaptation de la politique d'acheminement se situe à trois niveaux, chacun constituant un service dans la structure du RPM que nous décrivons ci-après.

5.3.4.2.1 Inférence des paramètres d'acheminement

Le service « *routing parameters inference* » a pour vocation de raisonner en fonction des données contextuelles sources et des messages pour inférer des paramètres d'acheminement nécessaires pour le routage des messages (expérience requise, capacité requise pour traiter une certaine tâche, délai requis pour la lecture du message, etc.). Il possède également la capacité d'adapter automatiquement la valeur des paramètres en fonction de nouvelles observations et des changements contextuels. Par exemple, un changement de localisation géographique d'un des acteurs impliqués d'un environnement non hostile vers un autre hostile et difficile d'accès peut influencer le type de ressources matérielles requises. Un hélicoptère peut être par exemple demandé en cas de situations urgentes pour secourir une personne bloquée en haute montagne. Nous évoquerons dans le Chapitre 8, la méthode que nous avons choisie pour implémenter ce service.

5.3.4.2 Stratégie de routage et changement de niveaux de priorité des messages « *Routing and escalation decision* »

Ce service reçoit des notifications à partir du *Message status Supervisor*, de l'*Ecosystem supervisor* et du *Message disseminator*, et adapte automatiquement ses décisions de routage et de niveau de priorité du message selon un processus de progression par paliers (*escalation*). Une action décisionnelle de routage peut prendre trois valeurs : (ré) envoyer, annuler, rediriger. L'action d'*escalation* consiste à augmenter (*escalate*) ou bien à réduire (*de-escalate*) le niveau de priorité du message. Une base de connaissances est fournie pour permettre au *Routing and escalation decision* d'apprendre à partir des connaissances et des expériences passées pour prendre ses décisions de manière autonome. En temps réel, le service « *Routing and escalation decision* » peut prendre la décision de changer l'itinéraire du message à cause d'un changement de contexte (localisation géographique, etc.) ou d'un dépassement des délais requis pour le traitement du message ou d'un problème de connexion identifié par le « *Message disseminator* ». En cas de redirection du message, le *Routing and escalation decision* invoque le service « *Routing parameters inference* » pour déterminer de nouveaux besoins de routage et un nouveau profil type de destinataire.

5.3.4.2.3 La détermination de la destination : « *Destination determination* »

Ce service adresse la question suivante : Dans un scénario donné, quel est le destinataire le plus approprié vers lequel le message doit être envoyé ? A partir d'une liste de destinataires préconfigurée, le service « *Destination determination* » cherche le destinataire le plus proche du profil du destinataire type préalablement déterminé par le service « *Routing parameters inference* ». Ce processus demande une analyse approfondie des données contextuelles de la cible. Pour chaque scénario, le classement de la liste des destinataires se fait en deux étapes : (1) La première étape consiste à classer la liste des destinataires selon les *background-attributes*, par exemple selon la compétence, la spécialité, la catégorie professionnelle, etc. On aura ainsi une première liste de destinataires. (2) La deuxième étape consiste à classer les destinataires potentiels de la liste issue de la première étape, mais selon des *real-time attributes*, par exemple, selon l'emplacement géographique du destinataire, l'accessibilité de son lieu d'emplacement, etc. Le message est envoyé ainsi au destinataire classé premier, en tête de liste.

Le classement des différents destinataires se fait selon des critères auxquels sont assignés des poids dont leur importance diffère d'un scénario à un autre. Par exemple, dans le cas d'un scénario très urgent, le critère de localisation géographique du destinataire aura un poids très important vu l'importance d'identifier la distance la plus proche par rapport à l'expéditeur de la requête. Pour sélectionner la destination la plus appropriée, plusieurs méthodes peuvent être adoptées à savoir la fonction d'utilité multicritères ou l'algorithme du K-plus proches voisins (K-NN). La méthode K-NN permet par exemple de déterminer la destination la plus proche du destinataire type en se basant sur une fonction de distance euclidienne ou de dissimilarité [180].

5.3.5 La supervision des écosystèmes : « *Ecosystem supervisor* »

L'*Ecosystem Supervisor* supervise les écosystèmes source, cible et message. Il collecte les attributs contextuels actuels et passés, les compare et détecte des changements de situations de contexte et notifie le RPM. Une sorte de compromis est préétablie entre l'*Ecosystem supervisor* et le RPM pour définir des seuils pour les observations, à partir desquels l'*Ecosystem supervisor* doit notifier le RPM, pour des situations d'adaptation auxquels le RPM est fortement sensible. A titre d'exemple, l'aggravation d'un paramètre clinique en santé ou d'un indicateur météorologique en gestion territoriale peuvent entraîner un besoin d'adaptation.

5.3.6 La supervision du statut des messages : « *Message status supervisor* »

Ce composant logiciel a pour mission principale de contrôler l'état des messages tout au long de leur cycle de vie. Pour chaque destinataire du message, il supervise les temps d'attente des messages, détecte les dépassements de délais requis pour la réception, la lecture et le traitement des messages et notifie le RPM pour qu'il adapte ses décisions, si nécessaire. Le *Message Status Supervisor* active son système de chronométrage une fois qu'il reçoit une notification à partir du *Message Disseminator* indiquant l'heure d'envoi du message et les délais requis pour la réception, la lecture et la réponse au message. Nous intégrons ce composant pour éviter des impasses (*deadlock*) et promouvoir la possibilité d'évaluer la « performance » des destinataires en terme de temps de réponse, de la fréquence des impasses survenues dans un laps de temps donné, eu égard à l'expérience acquise par le système.

5.4 Modélisation ontologique des écosystèmes sources, cibles et messages

Comme le montre la Figure 40, nous avons opté pour le modèle de contexte basé sur les ontologies. Notre choix s'est porté sur une modélisation ontologique pour trois raisons essentielles : (1) les ontologies nous fournissent une forte expressivité dans la représentation des données contextuelles ainsi que les relations entre elles ; (2) elles fournissent une formalisation sémantique des objets du monde réel dans un langage compréhensible par la machine ; (3) en outre, les ontologies nous permettent l'inférence des informations du contexte grâce au mécanisme de raisonnement qu'elles fournissent. La Figure 40 présente le modèle ontologique générique ou méta-modèle que nous proposons pour modéliser les écosystèmes sources, cibles et messages. Nous avons ainsi trois ontologies couplées et dont le contenu contextuel peut être dépendant. Comme le montre la Figure 40, le modèle ontologique proposé est organisé en couches.

La couche « *context provision* » : c'est la couche qui représente la source des données contextuelles. Elle est composée d'un ensemble de capteurs physiques fournissant des données brutes comme le GPS pour la localisation géographique et/ou des capteurs virtuels qui sont disponibles via des services Web. Par exemple, on peut connaître la température ambiante en naviguant sur un site météorologique.

La couche « context » : c'est la couche qui modélise les données contextuelles brutes issues des capteurs et appartenant à une entité qui peut être un objet, une place ou une personne.

La couche « context interpretation » : cette couche est responsable d'inférer des attributs contextuels à haut niveau d'abstraction à partir d'autres attributs atomiques provenant de la couche précédente. Les données atomiques de bas niveau provenant directement des capteurs sont souvent brutes et donc insuffisantes pour prendre une action de décision par exemple. Il suffit de les agréger avec d'autres données, de monter à un niveau plus haut d'abstraction pour en déduire une donnée contextuelle plus pertinente. Par exemple, la donnée issue d'un capteur détecteur de mouvements ne permet pas toute seule de déterminer si la personne est tombée par terre ou non. Alors que combinée avec d'autres données brutes comme le niveau de son et la localisation, la déduction devient plus précise.

Cette interprétation est réalisée via des règles de raisonnement logique fournies par les ontologies. Nous distinguons deux types d'interprétations : l'interprétation locale et l'interprétation multi-sources distribuées. Nous appelons interprétation locale, la déduction des attributs à partir d'autres attributs appartenant au même écosystème et donc à la même ontologie. Par exemple, l'état climatique du site de l'expéditeur du message est déduit à partir des données appartenant au même écosystème, l'écosystème source, (par exemple en fonction de la température, pluie, neige, etc.). L'interprétation multi-sources distribuées se fait à partir d'attributs pouvant appartenir à différents écosystèmes. Par exemple, l'attribut « distance géographique » est interprété en fonction de la localisation géographique source (attribut qui appartient à l'écosystème source) et de la localisation de la cible (attribut qui appartient à l'écosystème cible).

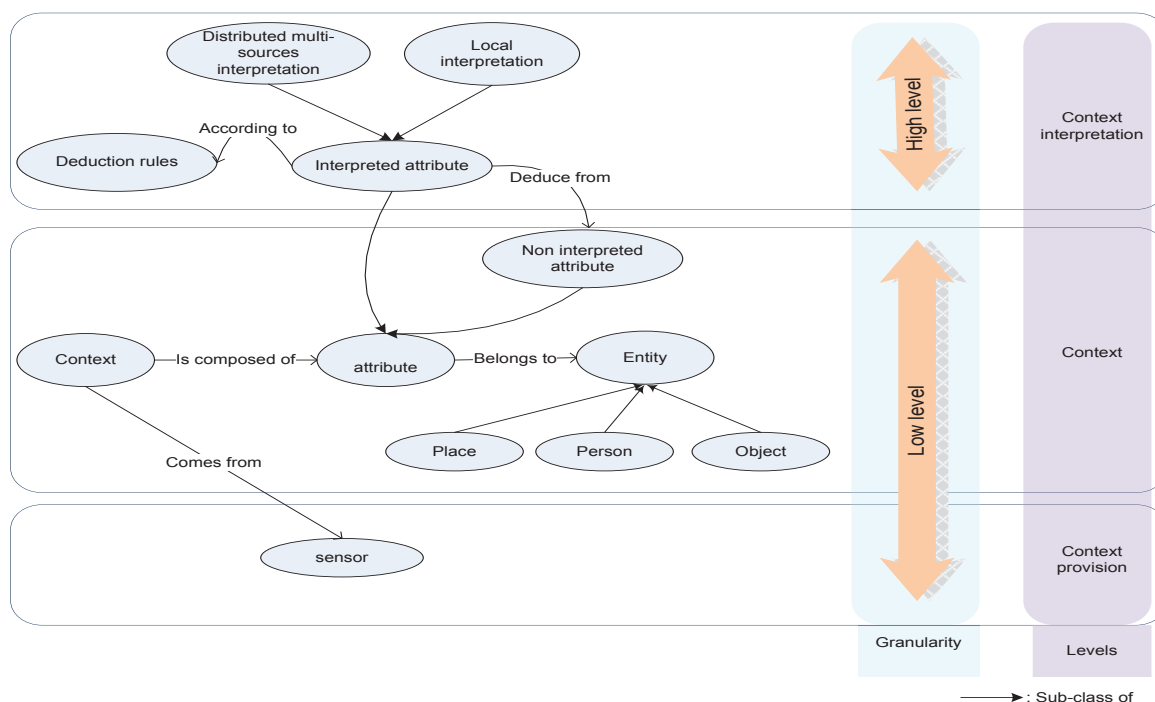


Figure 40. Modèle ontologique du contexte

5.5 Prise en compte des besoins requis pour l'acheminement des messages

Notre architecture répond aux neuf besoins requis pour l'acheminement des messages que nous avons déduits dans le chapitre 2 : *personnalisation, sensibilisation, auto-adaptation, interopérabilité, supervision, autonomie, sécurité et protection de la vie privée, intelligence, qualité de services*. Ces besoins représentent un chevauchement modélisé par un puzzle (Figure 41).

Sur un plan fonctionnel, notre architecture permet la *personnalisation*, la *supervision*, l'*autonomie* qui elles-mêmes sont liées à la *sensibilité au contexte*. Les mécanismes d'*auto-adaptation* aux propriétés de *personnalisation* confèrent la qualité d'*autonomie*. Sur un plan non fonctionnel, notre architecture doit répondre au besoin d'*interopérabilité* eu égard à sa compatibilité aux standards internationaux d'échanges de données. Cette compatibilité sera illustrée dans le framework proposé dans le chapitre 7. Le besoin d'*interopérabilité* fait lui-même appel à la sécurité des données échangées. Toutes ces propriétés seront pilotées par des méthodes d'*intelligence artificielle*.

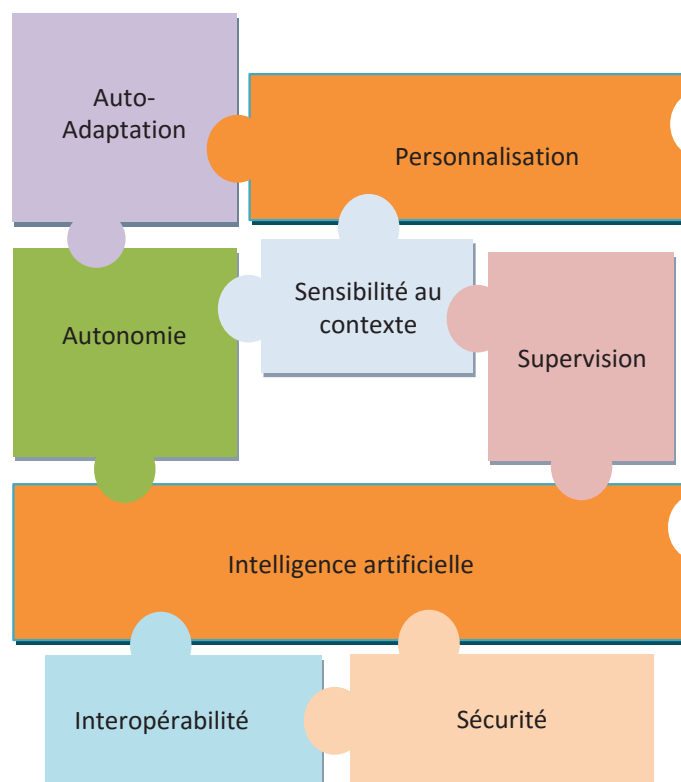


Figure 41. Représentation des besoins d'acheminement des messages

Chaque service de l'architecture prend en compte un ou plusieurs besoins. Comme le montre le tableau ci-après, les deux services « inférence des paramètres » et « détermination de la destination » répondent tous les deux au besoin « **personnalisation** » de la stratégie d'acheminement. En effet, le choix du destinataire dans le service « détermination de la destination » se fait d'une manière personnalisable, en fonction de la situation contextuelle à la fois de la source, de la cible et du message lui-même. En e-santé par exemple, le choix du professionnel de santé (médical/paramédical) vers lequel le message sera envoyé, est défini en

fonction de la situation médicale, de la situation géographique du patient (à domicile, en maison de retraite, etc.) et du professionnel de santé, du type de message (alarme, technique, etc.), etc. En gestion d'incendie par exemple, le choix de la destination « organisme de pompiers » dépend des missions à accomplir (prévention, protection, lutte contre les incendies, etc.), du lieu de l'incendie (à domicile ou sur la voie publique), du nombre des personnes à secourir, de la nature du secours (secours en montagne/mer, etc.).

Le service « inférence des paramètres » permet aussi la **personnalisation** puisqu'il infère les paramètres d'acheminement qui correspondent parfaitement aux situations contextuelles des acteurs. Par exemple, l'inférence du délai requis pour le traitement du message est fonction du niveau de priorité du message, de la situation sociale de l'expéditeur du message, etc.

Le besoin « **sensibilité au contexte** » est pris en compte dans les services « inférence des paramètres d'acheminement » et « Stratégie de routage et changement de niveaux de priorité des messages ». En effet, ces services s'appuient sur des données contextuelles et s'auto-adaptent en conséquence ; Ils satisfont ainsi au besoin B3 : « **auto-adaptation** ». Le service « transformation du message » permet l'interopérabilité vu qu'il transforme les données contenues dans un message conforme au standard international d'échange des données.

La « **supervision** » (B5) est présente dans notre architecture grâce aux deux services « supervision de l'écosystème » et « supervision de l'état de message ». Les services « inférence des paramètres d'acheminement », « la détermination de la destination » et « la stratégie de routage et changement de niveaux de priorité des messages » embarquent de l'intelligence artificielle, d'où ils remplissent le besoin B8 : « **intelligence** ».

Services de l'architecture	Besoins							
	B ₁ Person.	B ₂ Sensib.	B ₃ AutoAdapt.	B ₄ Interop.	B ₅ Superv.	B ₆ Auton.	B ₇ Sécu.	B ₈ Intel.
Analyse du message				×			×	
Inférence des paramètres d'acheminement	×	×	×			×		×
Stratégie de routage et changement de niveaux de priorité des messages		×	×			×		×
Détermination de la destination	×		×					×
Supervision des écosystèmes					×			
Supervision du statut des messages					×			

Tableau 4. Nature des besoins satisfaits par les services de notre architecture d'acheminement des messages ; B1 : Personnalisation, B2 : Sensibilité au contexte, B3 : Auto-Adaptation, B4 : Interopérabilité, B5 : Supervision, B6 : Autonomie, B7 : Sécurité et protection de la vie privée, B8 : Intelligence.

5.6 Conclusion

En résumé, dans ce chapitre, nous avons présenté une approche originale d'orchestration de services intelligents pour la gestion d'acheminement de messages. Nous avons proposé un modèle illustrant, d'une façon générale, la sensibilité du processus décisionnel d'acheminement aux contextes source, cible et message. Une architecture orientée services a été également proposée pour le pilotage autonome et intelligent des transmissions de messages. Cette architecture répond parfaitement aux besoins requis pour une meilleure gestion du pilotage des messages. Le chapitre suivant aborde l'aspect modélisation métier des services de notre architecture.

Chapitre 6

Modélisation des processus métiers pour la gestion de l'acheminement des messages

6.1 Introduction

L'objectif de ce chapitre est de présenter une démarche d'analyse dynamique du système de gestion de l'acheminement des messages. La modélisation des processus métiers nous permet de formaliser le fonctionnement précis des différents composants de l'architecture proposée dans le chapitre précédent. Un processus métier est un ensemble d'activités corrélées ou interactives qui transforment des éléments d'entrées en éléments de sortie (ISO 9000:2000). Chaque composant de l'architecture exécute de son côté un processus. Nous proposons dans ce chapitre une modélisation métier en utilisant la notation BPMN [3].

La première partie de ce chapitre sera consacrée à la modélisation métier du processus « supervision des écosystèmes ». La deuxième partie s'intéresse au processus « supervision du statut de message ». La dernière partie, quand à elle, sera dédiée à la modélisation du processus métier « gestion d'acheminement des messages ». Ce processus représente une certaine complexité dans son fonctionnement. Afin de mieux appréhender son exécution, nous avons ainsi choisi de modéliser en fin de chapitre les deux sous-processus qui le composent : « inférence des paramètres d'acheminement » et « détermination de la destination ».

6.2 Modélisation du processus métier « Supervision des écosystèmes »

A la réception d'une demande de supervision des écosystèmes depuis le *Message Disseminator*, le processus de supervision des écosystèmes collecte en parallèle des données contextuelles source, cible et message passées et courantes (Figure 42). Il compare ces données, vérifie s'il y a des changements de contexte. Si c'est le cas, il notifie le *Routing Policy Manager* (RPM) et le *Message status supervisor* pour qu'ils adaptent leurs comportements en conséquence. Par exemple, la distance séparant la source du message de sa cible est égale à « proche » au moment du transfert du message. Après transfert du message, cette valeur peut changer et devenir « loin », ce qui peut influencer la décision de routage. Lors de l'exécution de la tâche « vérifier changement du contexte » : trois événements interruptifs peuvent survenir générant la sortie de tout le processus : le destinataire du message rejette la requête, l'émetteur annule la requête (en cas de fausse alarme par exemple), le destinataire dépasse le délai requis pour la réception, la lecture ou la réponse au message.

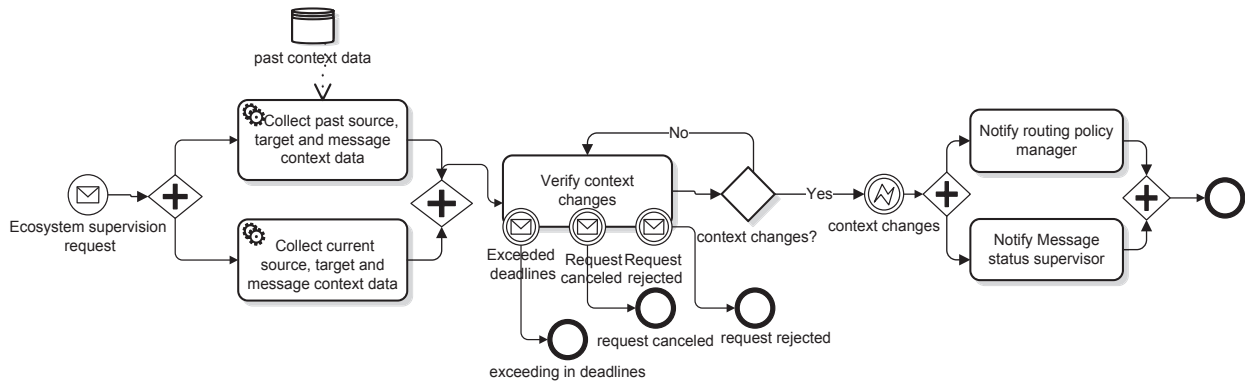


Figure 42. Le processus de supervision des écosystèmes

6.3 Modélisation du processus métier « Supervision du statut des messages »

Ce processus se déclenche lors de la réception d'une demande de supervision du *Message disseminator* (Figure 43). Le processus de supervision du statut de message reçoit à partir du *Message disseminator* les délais requis pour la réception, la lecture et la réponse au message, ainsi que la date et l'heure de son envoi. Il exécute par la suite la tâche « supervision de la réception ». Si au moment de la supervision, le *Message reporter* annule la transmission du message, le processus met à jour le niveau de confiance de la source, enregistre les modifications et se termine. Si le processus détecte un dépassement de délai de réception du message, il notifie le RPM pour adapter ses décisions en conséquence, mémorise l'exception et s'arrête. Si le message est reçu dans le temps, le processus passe à la supervision de la lecture. Six événements interruptifs de la tâche « *supervise reading* » peuvent survenir : acceptation du message par le destinataire sans le lire, rejet du message, annulation du message, changement de contexte détecté par l'*Ecosystem supervisor* ou annulation du message à partir du *Message Reporter*. En cas d'acceptation, d'annulation ou de rejet du message par le destinataire, le processus de supervision du statut du message met à jour le niveau de confiance du destinataire, enregistre ces modifications et se termine. En cas du déclenchement d'une exception de changement de contexte, tout le processus de supervision du statut du message s'arrête. Si le *Message Reporter* annule l'émission du message, un calcul du niveau de confiance source suivi d'un enregistrement du temps de déclenchement d'exception seront effectués. Si le destinataire excède le délai requis pour la lecture du message, le processus notifie le RPM, calcule et enregistre la nouvelle valeur du niveau de confiance cible. Si le message est lu par le destinataire, le processus passe à la dernière tâche « supervision de la réponse ».

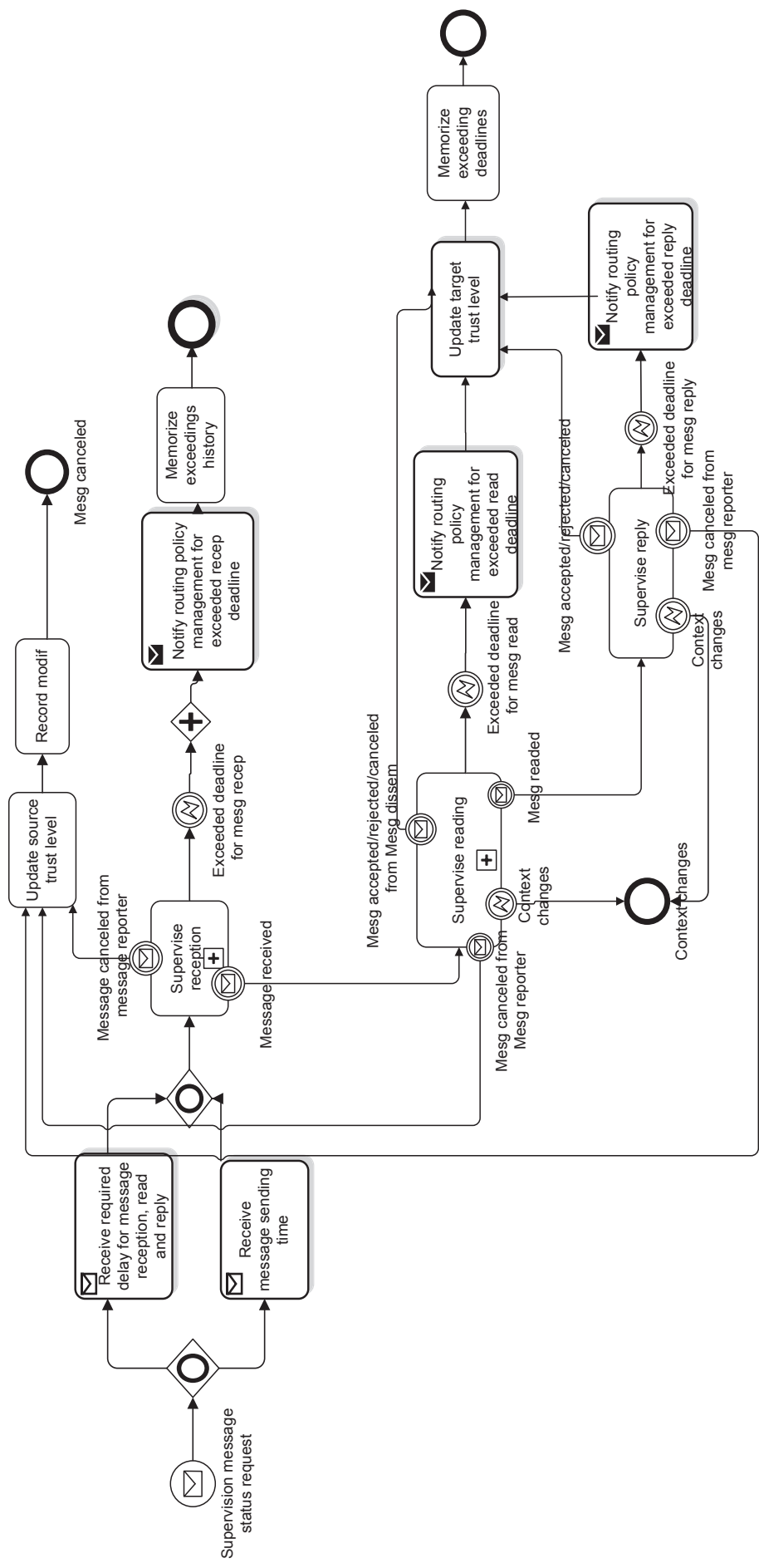


Figure 43. Le processus supervision du statut du message

Les mêmes événements interruptifs au niveau de la tâche « supervision de la lecture » peuvent survenir en temps réel lors de la supervision de la réponse.

6.4 Modélisation du processus métier « Gestion de l'acheminement des messages »

Ce processus est basé principalement sur des méthodes d'intelligence artificielle pour raisonner automatiquement sur les données contextuelles afin de délivrer le message au bon destinataire, prendre la bonne décision d'acheminement et adapter son comportement en fonction des notifications reçues des autres processus. Comme indiqué Figure 44, le processus se déclenche dès la réception d'un message. Ce dernier est issu du *Message Reporter* qui peut être un PDA, un système d'appel infirmiers, etc. Ensuite, le processus commence à valider le message syntaxiquement et sémantiquement. L'analyse sémantique consiste à vérifier l'existence d'un destinataire préféré par l'expéditeur, source du message. Dans le cas où la destination est indéterminée, le présent processus invoque les deux sous processus : *Routing requirements inference* et *Search for the relevant destination*.

Le sous-processus « *Routing requirements inference* » génère un profil type de destinataire, information en entrée du sous processus « *Search for the relevant destination* ». Ce dernier produit une liste de destinataires classés les plus adéquats pour la prise en charge de la requête. Le message est ensuite transféré vers le destinataire classé en tête de la liste. Le processus « *Message routing management process* » peut invoquer par la suite le service « *Decide routing and escalation* » pour prendre les décisions : rediriger ou renvoyer le message, augmenter ou réduire le niveau de priorité en cas de déclenchement des événements suivants : la réception d'un rejet du message par le destinataire, d'un changement de contexte détecté par l'*Ecosystem supervision process* ou d'un dépassement des délais identifié par le *message status supervision process*. En fonction de ces événements ainsi que de l'historique de disponibilité, de l'expérience et du « niveau de confiance » des destinataires, le service « *Decide routing and escalation* » prend la décision de renvoyer le message vers le même destinataire qu'auparavant, ou de totalement changer de profil type et donc appeler de nouveau le service « *Routing requirement inference* ». Par exemple une décision de renvoi du message vers le même destinataire peut être prise, si ce dernier est habitué à répondre positivement à la majorité des requêtes d'un expéditeur donné et s'il a une forte connaissance et longue expérience relative à la situation de l'expéditeur.

Le processus « *Message routing management* » se termine en cas de déclenchement des deux événements suivants : annulation de la requête par le destinataire, en cas de fausse alarme par exemple, ou par l'expéditeur lui-même. Dans le dernier cas, les deux autres processus « *Ecosystem Supervision* » et « *message status supervision* » se terminent également. L'implémentation du service « *Routing requirements inference* », comme nous le verrons dans le Chapitre 8-Section 8.2, sera basée sur les réseaux bayésiens dynamiques.

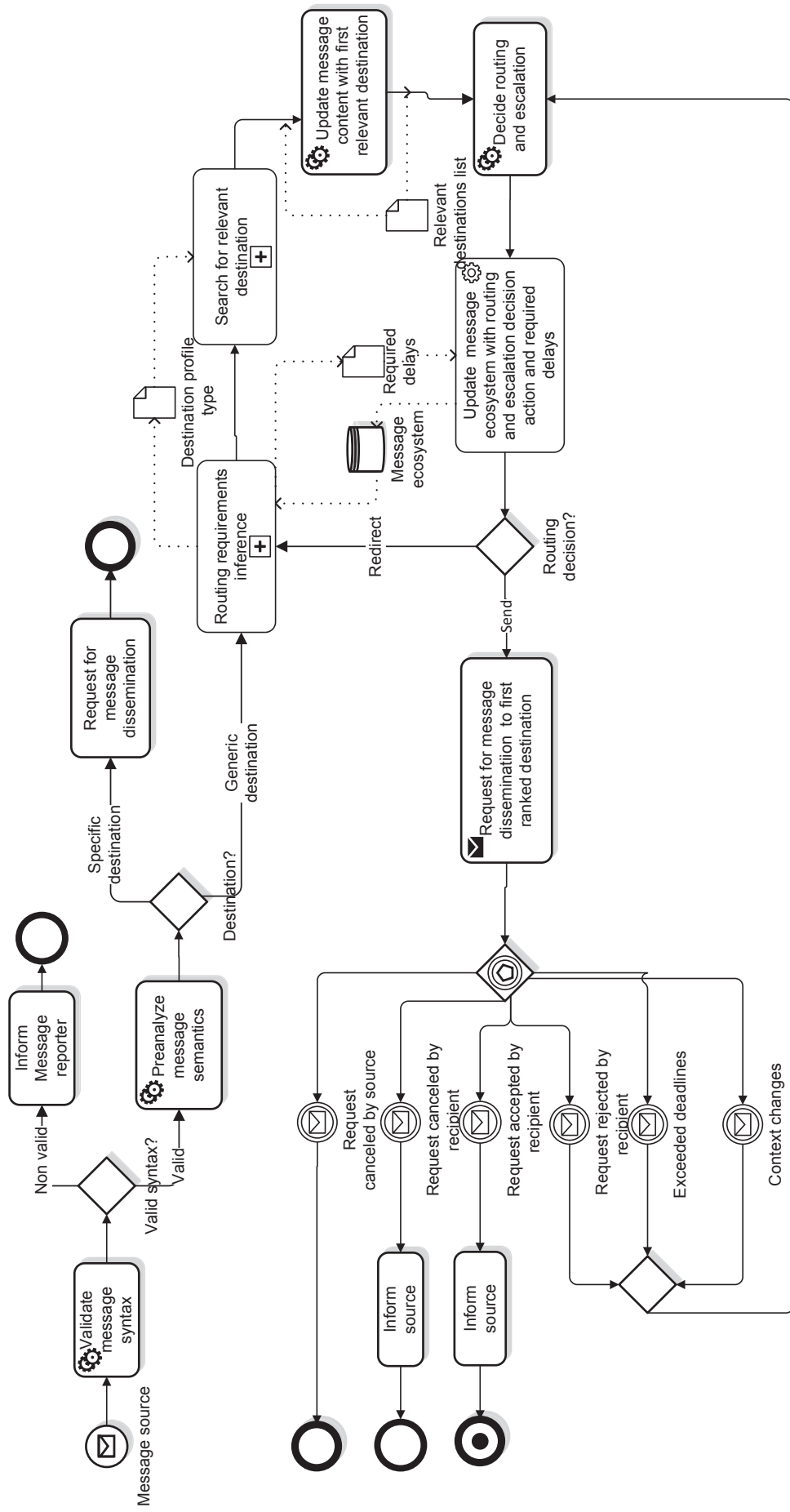


Figure 44. Le processus de gestion d'acheminement des messages

6.5 Modélisation du processus métier « Inférence des paramètres d'acheminement »

Explicitons maintenant le sous-processus « inférence des paramètres d'inférence » (Figure 45) du grand processus « Gestion de l'acheminement des messages ». Ce sous processus fait appel au superviseur des écosystèmes pour collecter des données contextuelles source et message. Il calcule la distribution de probabilités initiale à partir d'une base de connaissance et infère par la suite les paramètres d'acheminement. L'inférence peut être réalisée grâce à l'algorithme « *forward-backward* ».

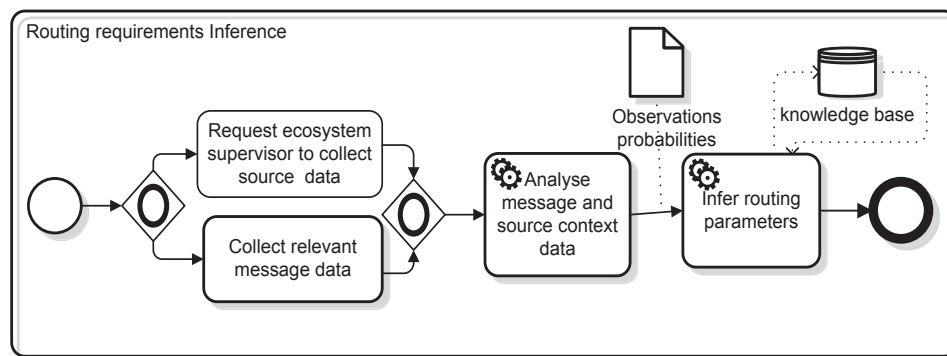


Figure 45. Le sous-processus d'inférence des paramètres d'acheminement

6.6 Modélisation du processus métier « Détermination de la destination »

Ce processus (Figure 46) se déclenche lors de la réception d'une requête de recherche de la destination la plus adéquate. Il reçoit le profil type de la destination et une liste préconfigurée des destinataires. A partir de cette liste et du profil type, il classe l'ensemble des destinataires selon les attributs *background* par exemple, selon leur niveau d'expérience relative à la situation de la personne expéditrice du message, leur compétence, leur catégorie professionnelle, leur niveau de confiance, leur niveau de professionnalisme, leur spécialité, etc. Une première liste de destinataires est donc fournie. A partir de cette liste, le processus exécute un deuxième classement multicritères des destinataires, mais cette fois-ci selon des attributs *real-time*, par exemple selon la distance qui le sépare de l'expéditeur, l'état climatique, les conditions du trafic, la disponibilité des ressources matérielles ou humaines, etc.

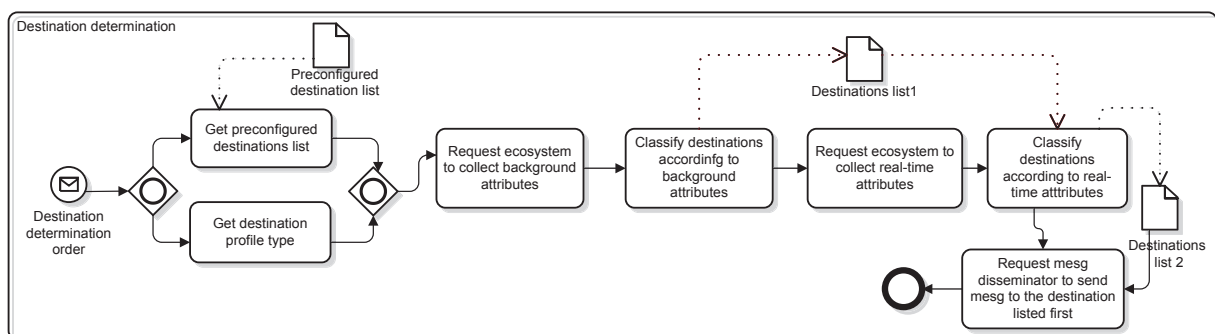


Figure 46. Le sous-processus de détermination de la destination en e-santé

6.7 Conclusion

Ce chapitre a permis de décrire le fonctionnement de notre système de gestion d'acheminement des messages au travers de la modélisation des processus métiers. Nous nous sommes concentrés sur les processus métiers exécutés par les services composants de notre architecture : la supervision des écosystèmes, la supervision de l'état de message et la gestion d'acheminement des messages.

La partie suivante constitue une étape vers l'implémentation d'un système de gestion d'acheminement des messages, dans le domaine particulier de l'e-santé. Le chapitre suivant sera dédié à expliciter les différents échanges existants entre les différents processus métiers. Aussi, pour préserver l'interopérabilité de notre solution, nous allons présenter notre architecture sous une vue synoptique appliquée à l'e-santé en général, et conforme au standard international d'échanges de données médicales en e-santé, HL7 version 3.

Partie IV
Vers l'implémentation d'un système de
gestion d'acheminement des messages.
Application à l'e-santé

L'objectif de cette partie est de proposer des solutions pour un pilotage intelligent, personnalisable et autonome de l'acheminement des messages qui répondent aux différents scénarios d'usage en e-santé. Nous présentons pour cela, une vue applicative de notre architecture générique présentée dans le chapitre 5, compatible HL7 version 3. Nous proposons trois types de modélisation : en couches, UML et ontologique de quelques exemples de contextes et d'écosystèmes. Nous démontrons avec des scénarios d'usage en e-santé que les réseaux bayésiens sont une piste très prometteuse pour garantir l'adaptation autonome du processus décisionnel d'acheminement des messages en fonction des changements contextuels continus dans le temps. Nous montrons, en outre, que la modélisation mathématique et plus particulièrement les réseaux de Petri temporels permettent de superviser le statut des messages, et donc éviter des impasses d'acheminement.

Nous présentons ci-après, le plan de cette partie.

Le chapitre 7 décrit les modèles et les flux d'informations dédiés au pilotage de l'acheminement des messages compatible HL7 version 3, que nous proposons.

Le chapitre 8 met l'accent sur le côté implémentation. Nous présentons notre choix méthodologique (les réseaux bayésiens dynamiques et les réseaux de Petri temporels) pour implémenter les processus métiers préalablement déduits dans le chapitre 6.

Le chapitre 9 est dédié à l'instanciation de notre solution et à l'illustration de son fonctionnement à travers un scénario concret dans le domaine de la cardiologie, et d'apporter des éléments de discussion sur la mise œuvre de notre approche.

Chapitre 7

Modèles et flux d'informations pour un pilotage de l'acheminement des messages en e-santé compatible HL7 version 3

7.1 Introduction

L'échange de données entre deux systèmes d'informations (SI) nécessite une interopérabilité sémantique pour garantir que les données échangées soient interprétées à l'identique par les SI destinataires et émetteurs. En santé, plusieurs organisations ont œuvré pour la standardisation des échanges informatisés tels que HL7 (*Health level 7*), IHE (*Integrating the Healthcare Enterprise*). Dans un souci de préserver l'interopérabilité de notre système, nous proposons dans ce chapitre une vue synoptique de représentation de notre architecture de gestion d'acheminement de messages, conforme au *Health Level Seven (HL7)* [50], standard international pour les échanges informatisés de données cliniques, financières et administratives entre différents systèmes d'informations hospitaliers.

Nous présentons dans une première partie de ce chapitre une vue applicative de notre architecture au domaine l'e-santé. Nous proposons trois modélisations des systèmes d'information de notre architecture auxquels une gestion d'acheminement peut être sensible : une modélisation UML, ontologique et en couches. Nous explicitons dans une troisième partie le contenu de différentes transactions existant entre les composants de l'architecture proposée dans le chapitre 5, appliquée ici au domaine de l'e-santé. Pour intégrer une vue dynamique à la solution proposée, nous illustrons les principales transactions par deux scénarios en e-santé que nous modélisons via des diagrammes de séquence UML.

7.2 Modèles d'informations appliqués au domaine de l'e-santé

Dans cette section, nous présentons un synoptique de représentation de notre architecture de gestion d'acheminement des messages (Figure 47) compatible HL7 RIM pour le pilotage intelligent, sensible au contexte et auto-adaptatif de messages HL7 [181]. En effet, à travers cette représentation, nous parvenons à étendre les fonctionnalités du RIM. Par ailleurs, nous combinons, comme le montre la Figure 47, les classes du RIM afférentes à notre problématique (*Entity, Role, Participation, Act, Transmission, Communication Function*) et préalablement définies dans le chapitre 1, aux aspects contextuels, l'ensemble étant piloté par des méthodes d'intelligence. L'aspect contextuel est illustré ici par les trois systèmes d'information : *Source Ecosystem, Target Ecosystem* et *Message Ecosystem*. Nous retrouvons

ici les composants logiciels (*Routing policy adaptation*, *Ecosystem supervision*, *Message Status Control*) qui interagissent avec nos trois écosystèmes via la relation représentée par le triplet (observer, notifier et adapter). Cette vue applicative de notre architecture au domaine de l'e-santé met en lumière l'importance du contenu du message dans le fondement d'une stratégie d'acheminement. Nous intégrons ainsi des éléments du RIM dans le contenu de l'écosystème message.

Nous n'allons pas revenir sur le fonctionnement de chaque composant logiciel. Néanmoins, nous allons mettre l'accent ici sur la structure de chaque écosystème et sur les relations qui existent entre ces différents composants.

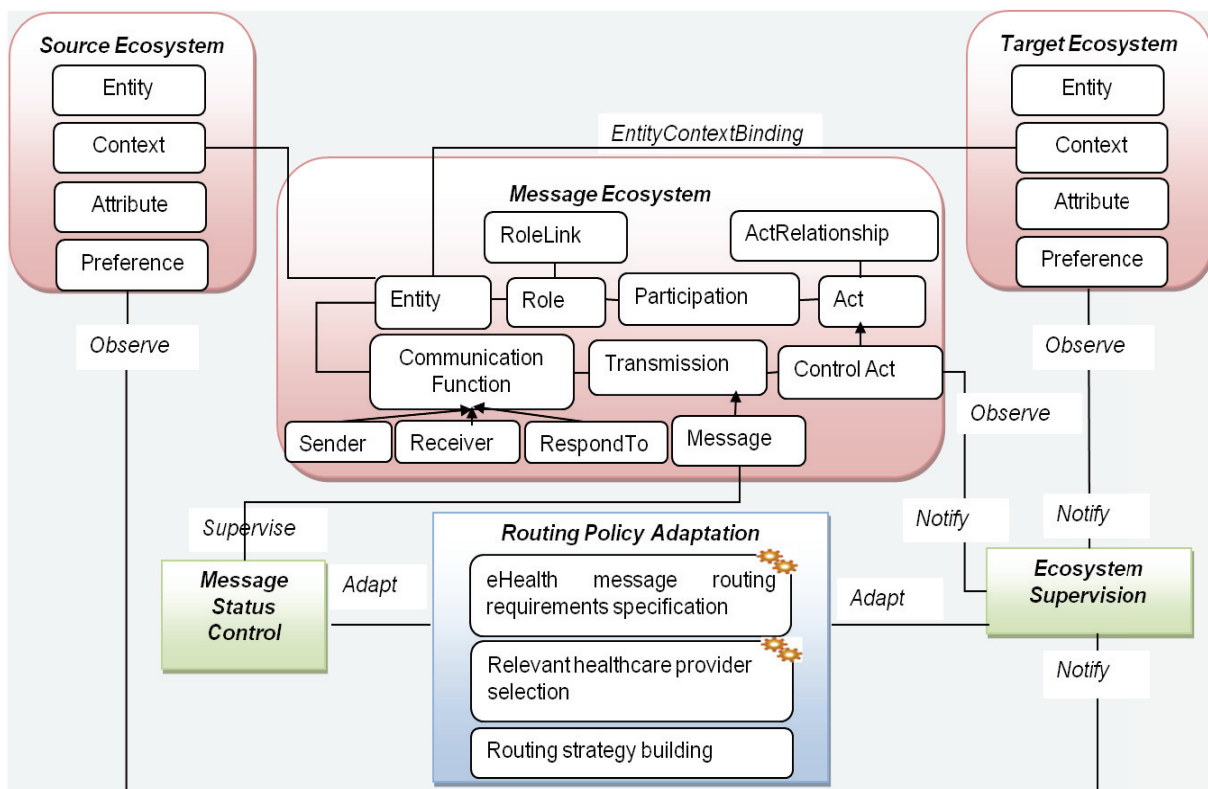


Figure 47. Synoptique de représentation de notre architecture de gestion d'acheminement des messages

7.2.1 Modélisation des écosystèmes source, cible et messages

Dans cette sous-section, nous allons proposer trois types de modélisations (en couches, UML et ontologique) des écosystèmes source, cible et message décrits dans la Figure 47. La modélisation en couches va nous permettre d'avoir une vue générique sur le contenu des systèmes d'information en fonction duquel les services du composant « *Routing Policy Adaptation* » doivent s'adapter. Nous procédons ensuite à une modélisation UML plus détaillée qui met en relation les différents objets du contexte. Motivés par la sémantique et les techniques d'interprétation que fournissent les ontologies, nous proposons une modélisation ontologique du contexte. Ces étapes de modélisation constituent pour nous un premier pas vers le découplage entre la partie données contextuelles et la partie application. L'objectif de cette modélisation est de donner une vue plus détaillée sur le contenu sémantique et contextuel des systèmes d'information. Le défi n'est pas simple car le contexte est appliqué à

une problématique spécifique dans le domaine particulier de l'e-santé, où les données sont volumineuses, diversifiées et donc difficiles à modéliser. Nous veillons donc à modéliser seulement les données contextuelles que nous jugeons pertinentes, c'est-à-dire auxquelles une politique d'acheminement peut être sensible.

7.2.1.1 Modélisation en couches des écosystèmes

Les écosystèmes source et cible : ont la même structure (Figure 48). Ils sont composés tous les deux de quatre couches : les couches entités, contextes, attributs et préférences.

- **La couche entités :** Elle est composée de tous les objets, personnes ou places impliqués dans le mécanisme d'acheminement de messages. Les entités de l'écosystème source sont le patient, l'expéditeur du message, le dispositif médical à l'origine du message, etc. L'expéditeur du message peut être un patient, un membre de sa famille, un professionnel de santé (infirmier(ère), médecin, etc.) ou un dispositif médical embarquant une gestion de communication des messages. Les entités de l'écosystème cible représentent généralement le dispositif qui reçoit la requête et son utilisateur qui peut être un professionnel de santé, ou un centre de surveillance, etc.
- **La couche contextes :** Les deux écosystèmes source et cible regroupent divers types de contextes (profil, environnemental, professionnel, spatio-temporel, etc.) associés à un ensemble d'entités. Le contexte profil regroupe toutes les coordonnées qui concernent l'entité de type « personne » tels que le nom, le prénom, l'âge, etc. Le contexte environnemental regroupe un ensemble d'attributs tels que le type de la localisation (hostile ou non hostile), l'état climatique (favorable ou non pour un déplacement), etc. Le contexte professionnel du professionnel de santé inclut son statut professionnel (médecin généraliste, infirmier (ère), assistant(e) médical(e), chirurgien, etc.), sa spécialité (cardiologie, neurologie, dentisterie, etc.), sa compétence, son niveau professionnel (faible, moyen ou haut) qui se détermine principalement en fonction de l'ancienneté du professionnel de santé dans son domaine, de sa connaissance relative à la situation médicale du patient, etc. Le contexte spatio-temporel regroupe les données qui concernent la localisation de la personne ou de la machine « expéditrice » ou « destinataire » du message. Le patient, par exemple, peut être à domicile, dans une maison de retraite, dans un centre de convalescence, en ambulatoire à l'extérieur, etc. Outre ces contextes, l'écosystème source comporte le contexte médical (les antécédents cliniques tels que : infarctus ancien, maladie d'Alzheimer, handicap physique, etc.), l'historique des prescriptions, les facteurs de risques (cholestérol, obésité, abus d'alcool, etc.), les paramètres biomédicaux (fréquence cardiaque, température, pression artérielle, etc.), les signes cliniques (douleurs thoracique, dyspnée, etc.) et la situation sociale du patient. Le contexte social comporte des données par exemple sur l'accompagnement du patient (est-ce qu'il est accompagné ou non ? si oui, est-ce par un professionnel de santé ou non ?).

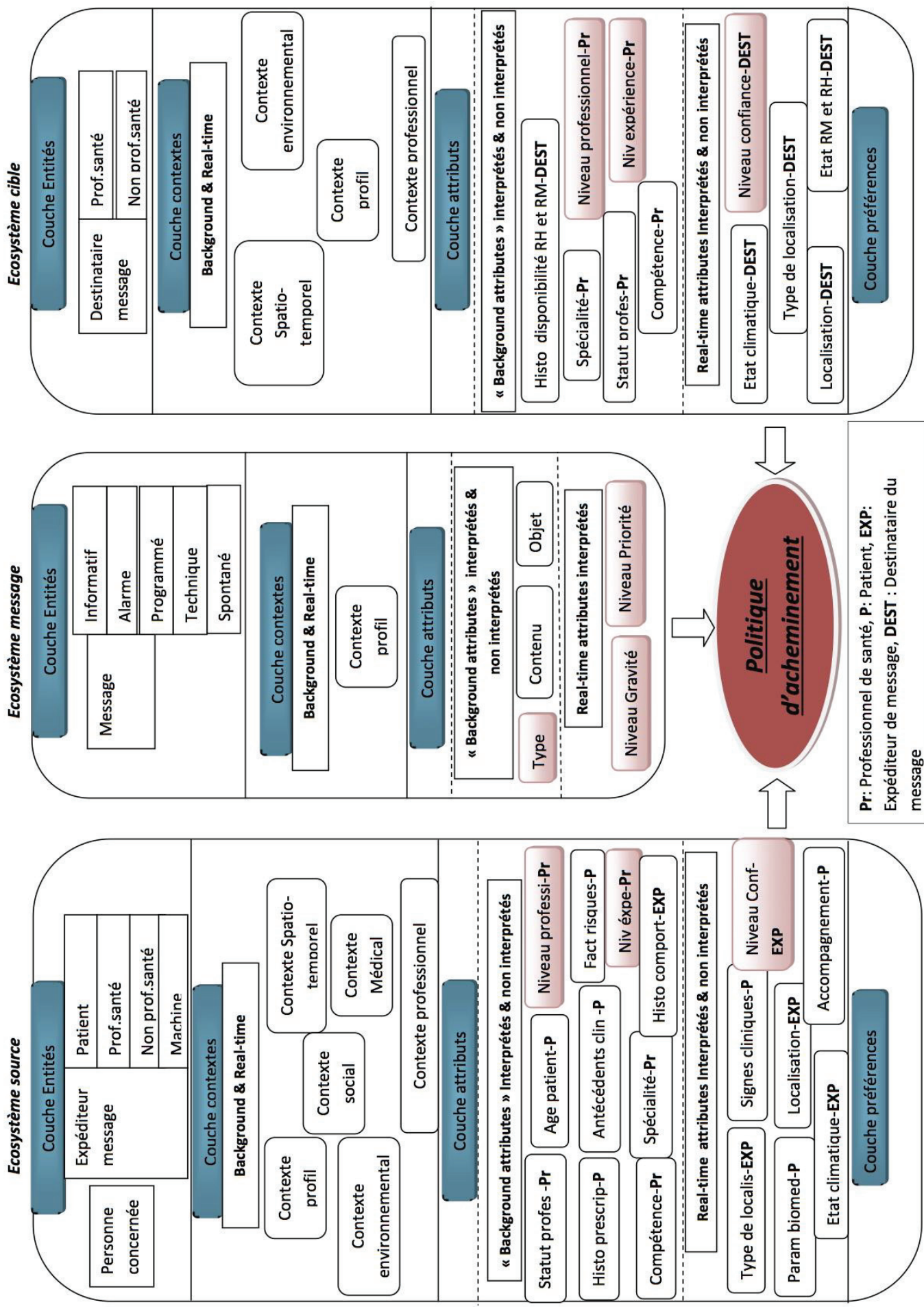


Figure 48. Modélisation en couches des écosystèmes source, cible et message

En effet, il est très important de connaître le contexte social, pour savoir par exemple si le patient qui est expéditeur et auteur du message est lucide, s'il est accompagné ou non et si l'accompagnant est un professionnel de santé ou non (un membre de famille). Ceci peut contribuer à définir un « niveau de confiance ² » sur le contenu du message, ce qui peut influencer la stratégie d'acheminement.

- **La couche attributs :** Elle intègre toutes les données contextuelles appartenant aux différentes catégories de contexte. Les antécédents cliniques du patient, son âge, ses facteurs de risques, présence de troubles comportementaux, etc. sont des exemples d'attributs de l'écosystème source. La compétence du professionnel de santé, et son expérience relative au patient, constituent des attributs appartenant à l'écosystème cible, tandis que le « niveau de confiance » est un attribut à la fois des écosystèmes source et cible.
- **La couche préférences :** L'expéditeur et le destinataire du message peuvent également avoir des préférences de routage. En effet, en tant qu'expéditeur, le patient peut choisir d'envoyer le message vers sa destination préférée, son médecin traitant par exemple. Il peut également posséder des préférences en termes de moyens d'acheminement (SMS, mail, etc.). De même, le professionnel de santé, en tant que destinataire, peut choisir par exemple le dispositif sur lequel le patient va recevoir la réponse de prise en charge (sur le PC du patient/ téléphone portable/ dispositif médical, etc.)

L'écosystème message : regroupe toutes les données contenues dans le message. Comme le montre la Figure 47, la structure de l'écosystème message est conforme au HL7 RIM. A partir du RIM, nous avons extrait les classes relatives à notre problématique de communication des messages. Nous retrouvons ici la classe entité qui peut jouer différents rôles eu égard à la transmission et qui participe au déclenchement d'un acte donnant naissance à un transfert du message. L'émetteur et le destinataire du message sont reliés à leur contexte via le lien « *EntityContextBinding* ». La modélisation en couche (Figure 48) montre que le message est composé d'un objet et d'un contenu. Dans le contenu du message, on trouve par exemple le type de message (alarme, informatif, programmé ou spontané), le niveau de gravité et le niveau de priorité. Un message technique peut faire l'objet d'une maintenance technique d'un dispositif médical par exemple. Un message peut être d'alarme majeure, mineure ou moyenne. Un message informatif peut être une demande d'avis sur une prise de médicament. On désigne par « programmé » un message qui est transmis automatiquement selon un protocole (par exemple, l'envoi de paramètres biomédicaux d'un patient chaque matin). Un message spontané est un message émis suite à une observation ou au déclenchement d'un événement.

² Il s'agit d'un niveau de confiance intrinsèque à notre problématique (compétences, disponibilité, etc.).

7.2.1.2 Modélisation UML des écosystèmes

Modélisation UML de l'écosystème source

Dans l'écosystème source (Figure 49), on trouve l'expéditeur du message qui peut être un être humain ou une machine. Un humain peut être un professionnel de santé ou non. Une machine peut représenter un dispositif médical intelligent qui détecte automatiquement la survenue d'un problème de santé et transmet un message à un destinataire quelconque. Un professionnel de santé est un acteur paramédical (infirmier, aide soignant, auxiliaire de soins etc.) ou médical (médecin traitant, centre d'urgence, spécialiste, etc.). Un non professionnel de santé représente par exemple le patient, un membre de sa famille, un technicien, etc. Le patient est lié à son contexte médical, spatio-temporel, environnemental et social. Chaque catégorie de contexte est composée d'un ensemble d'attributs que nous avons préalablement définis dans la modélisation en couches.

A un expéditeur de message correspond un niveau de confiance source que nous appelons dans le modèle « *Source_Trust_Level* ». Ce niveau de confiance est principalement calculé en fonction du contexte médical et de l'historique comportemental (le nombre de fausses alarmes ou de fausses requêtes de prise en charge si l'expéditeur est un patient), d'où la relation de dépendance avec le stéréotype « *use* » entre les classes « *Source_Trust_Level* », « *Medical_Context* » et « *Past_Behavior* ». Le même niveau de confiance dépend en plus du nombre de fausses alarmes, de l'identité professionnelle de l'expéditeur s'il est un professionnel de santé ou non. La déduction de ce niveau de confiance sera détaillée dans la modélisation ontologique.

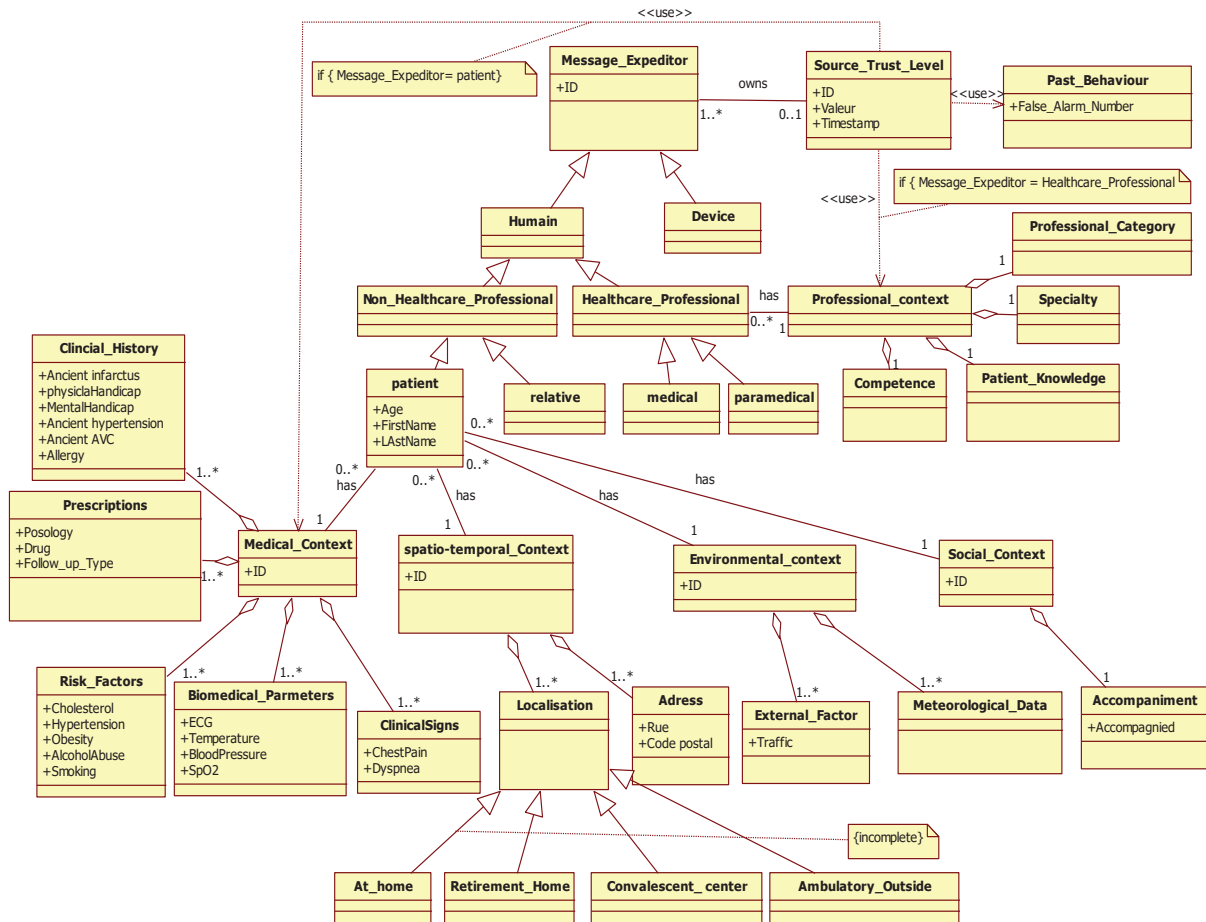


Figure 49. Diagramme de classes modélisant l'écosystème source

Modélisation UML de l'écosystème cible

L'écosystème cible regroupe les destinataires des messages qui peuvent être des professionnels ou non professionnel de santé (membre de famille du patient ou voisin, etc.). Nous modélisons, comme dans l'écosystème source, le contexte spatio-temporel, environnemental et professionnel, mais ici du destinataire du message. Nous ne modélisons pas le contexte médical et social puisque l'entité « patient » n'est pas présente dans l'écosystème cible. Le patient peut être aussi destinataire d'une réponse à un message de prise en charge par exemple. Mais, nous situons notre modélisation de l'écosystème cible dans un ordre chronologique avancé d'acheminement du message, c'est-à-dire depuis le déclenchement d'un problème de santé ou bien d'un problème technique, ou d'une demande d'expertise, etc., jusqu'à sa réception par un destinataire quelconque. Pour cela, le patient ne peut être qu'un expéditeur dans notre cas, et son contexte médical et social sont modélisés au niveau de l'écosystème source.

Comme indiqué dans la Figure 50, le destinataire est caractérisé par un niveau de confiance (*Target_Trust_Level*) qui peut être déduit en fonction de l'historique de disponibilité des ressources humaines (fréquence de non acquittement de message, temps écoulé entre la réception et la lecture du message, etc.) et matérielles (disponibilité de l'ambulance, d'un équipement/matériel médical particulier, etc.).

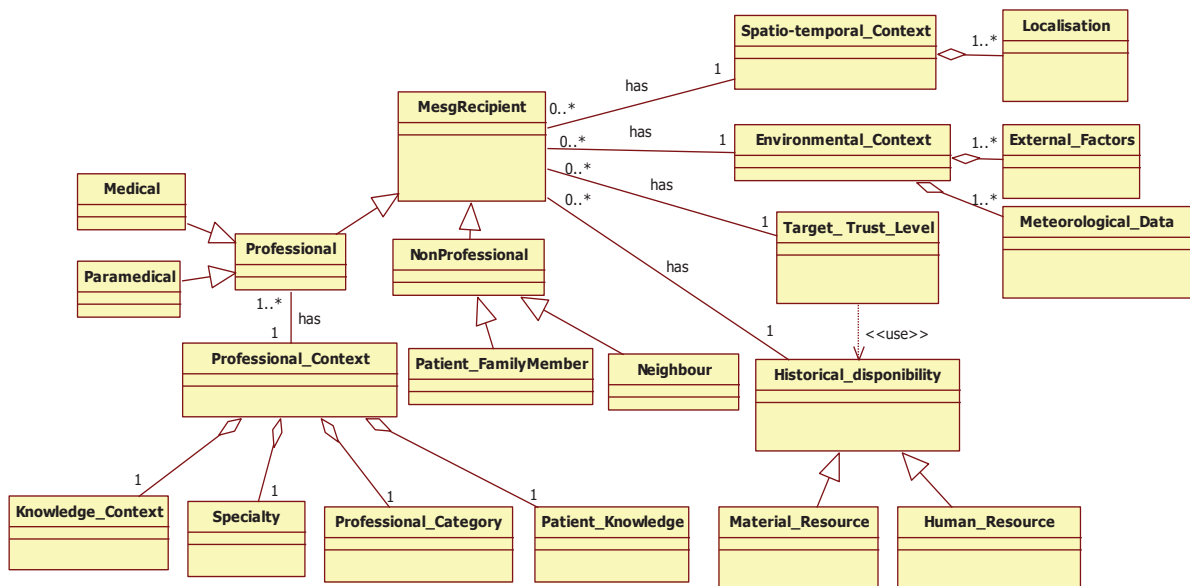


Figure 50. Diagramme de classes modélisant l'écosystème cible

Modélisation UML de l'écosystème message

L'écosystème message regroupe principalement l'ensemble des attributs qui existent dans le HL7 RIM, à partir desquels le contenu du message HL7 version 3 est construit. Pour une raison de simplicité, nous avons extrait seulement les attributs auxquels la stratégie d'acheminement peut être sensible. Dans l'écosystème message, nous rajoutons quelques propriétés (*Message_Type*, *Gravity_level*, *Priority_Level*) qui ne sont pas dans HL7, mais qui peuvent influencer la politique d'acheminement d'un message. La modélisation de ces propriétés est fournie dans la Figure 51.

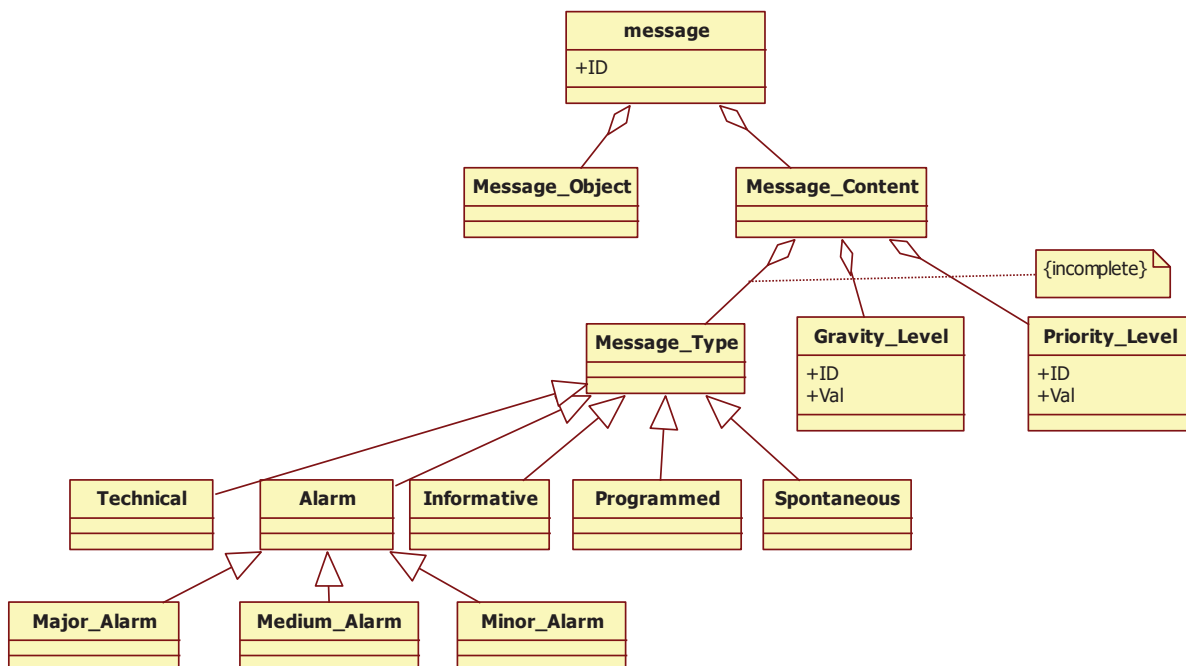


Figure 51. Diagramme de classes modélisant une partie de l'écosystème message

7.2.1.3 Modélisation ontologique de quelques exemples de contextes et d'écosystèmes

Dans cette section, nous avons choisi de concevoir un modèle ontologique des écosystèmes source et message, plus particulièrement dans le domaine de la cardiologie. Nous utilisons les ressources et les recommandations fournies par la Société Européenne de Cardiologie [182].

Nous nous limitons ici à quelques données contextuelles que nous jugeons indispensables pour l'établissement d'une stratégie d'acheminement. Il existe notamment de nombreux autres attributs que ce soit dans le contexte médical, social, géographique, environnemental ou social auxquels une stratégie d'acheminement peut être sensible mais que nous n'avons pas modélisés dans notre ontologie pour des raisons de simplicité.

Contexte médical

Nous avons organisé le concept de l'état des signes cliniques cardiaques (*CardiacClinical Signs state*) en une hiérarchie (Figure 52). Pour chaque signe clinique en cardiologie, nous avons défini un ensemble de sous classes. Les nausées peuvent être *normal*, *below low normalNausea*, *below lower panic limitsNausea*, *above high normalNausea*, *above higher panic limitsNausea*, *abnormalNausea*. Pour les douleurs thoraciques, elles peuvent être normales, sévères, ou très sévères. La respiration peut être une respiration normale ou critique signalant une dyspnée. Nous trouvons également la syncope / perte de connaissance comme étant un signe clinique des maladies cardiovasculaires.

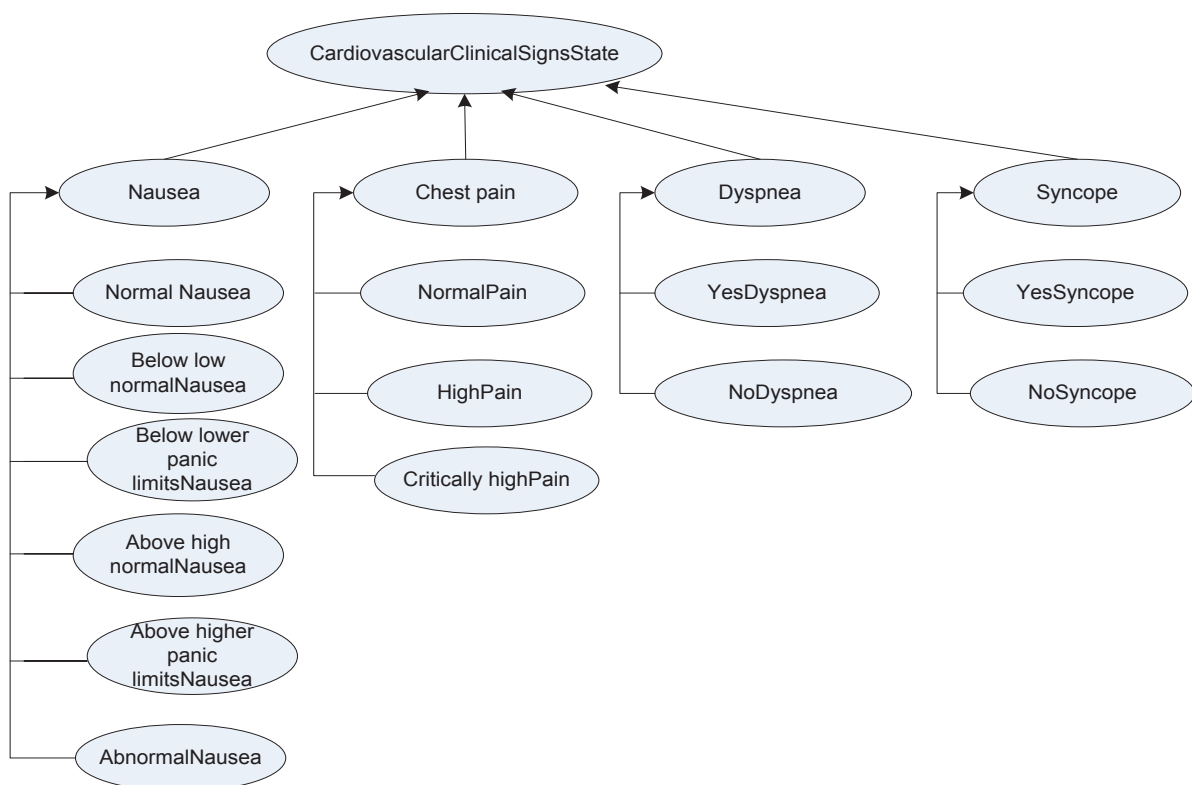


Figure 52. Classification de l'état des signes cliniques en cardiologie

Nous avons opté pour la même stratégie de classification pour différencier les différents états des paramètres biomédicaux (Figure 53). Nous classons les paramètres (Température, ECG, pression artérielle, SpO2) selon leurs degrés de gravité ou de normalité. La tension par exemple est classée en tension normale, hypertension, hypertension sévère, hypotension et hypotension sévère.

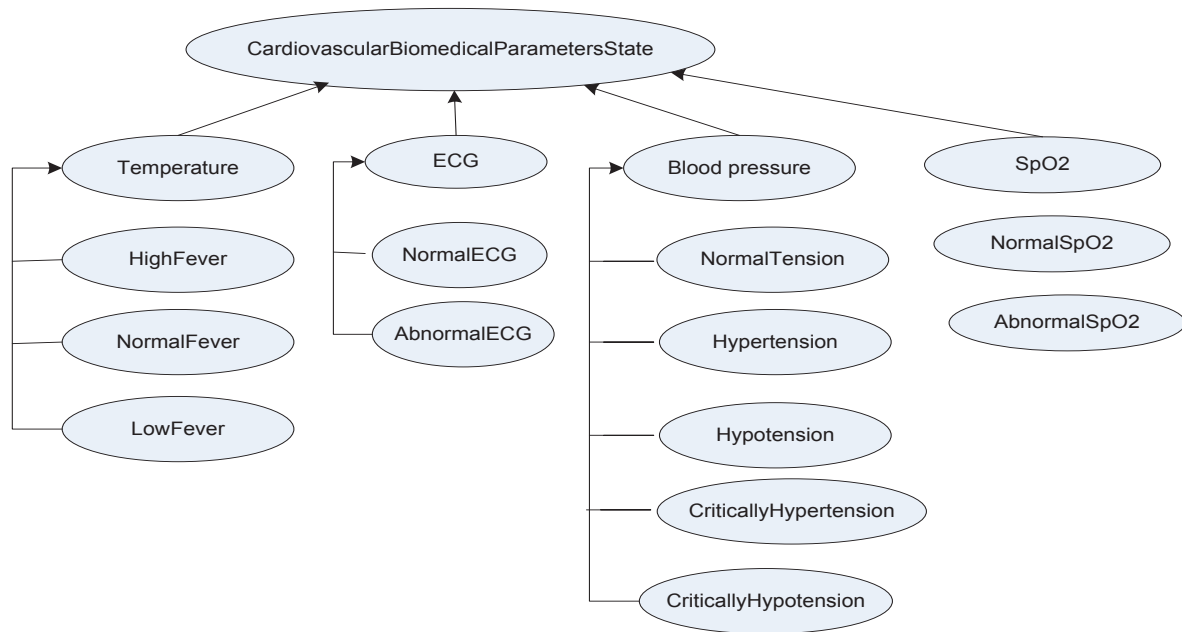


Figure 53. Classification de l'état des paramètres biomédicaux en cardiologie

De même, la hiérarchie du concept « *CardiacRiskFactorsState* » est représentée Figure 54. Nous distinguons les états de normalité du cholestérol et du glucose. D'autres facteurs de risques comme l'obésité, le tabagisme, le diabète sont représentés dans notre modèle.

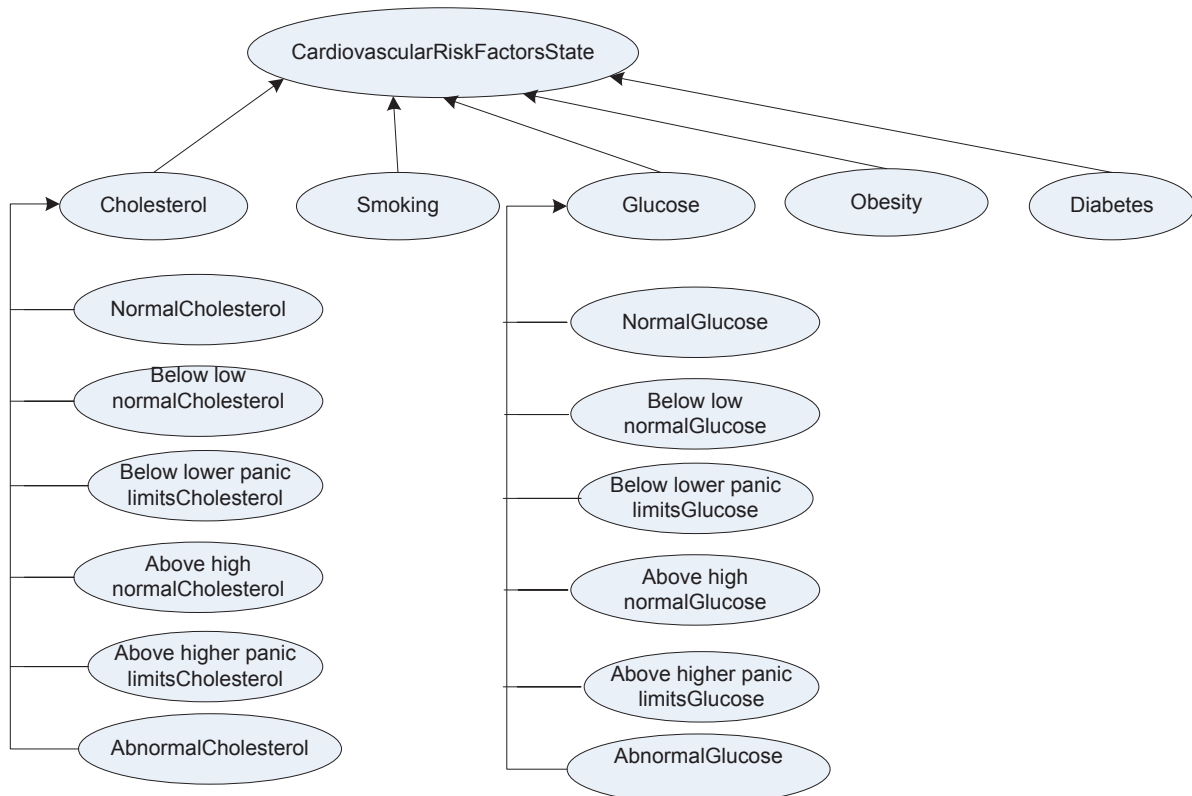


Figure 54. Classification des facteurs de risques en cardiologie

L'état des facteurs physiologiques peut être différent d'un genre à un autre (homme et femme) et d'une catégorie d'âge à une autre (enfant, adulte, personne âgée). Les données proviennent de différents types de dispositifs médicaux formant les sources des données contextuelles médicales tels que : le thermomètre, le glucomètre, l'oxymétrie.

Contexte environnemental

La connaissance de données environnementales du site de l'expéditeur du message est nécessaire surtout en cas d'urgence et de possibilité de transfert du patient. Nous distinguons deux types d'environnement qui caractérisent le site de l'expéditeur du message : un environnement de type « *hostile* » (montagnard par exemple) et l'autre de type « *non hostile* ». Nous classons l'état du climat en climat favorable et l'autre non favorable. De même pour le concept accessibilité, nous distinguons quatre sous-classes : Accès difficile, très difficile, facile, très facile. Nous distinguons 3 types d'états du trafic : fluide, dense et chargé. La hiérarchisation des concepts « *TrafficState* », « *EnvironnementType* », « *ClimateState* » et « *AccessibilityState* » et ses différentes sous classes est représentée Figure 55. Toutes ces données contextuelles sont jugées utiles pour la détermination par exemple des types de ressources matérielles requises pour transférer le patient (hélicoptère, une ambulance, etc.), pour la recherche du destinataire dont le contexte est le plus favorable pour se déplacer, etc.

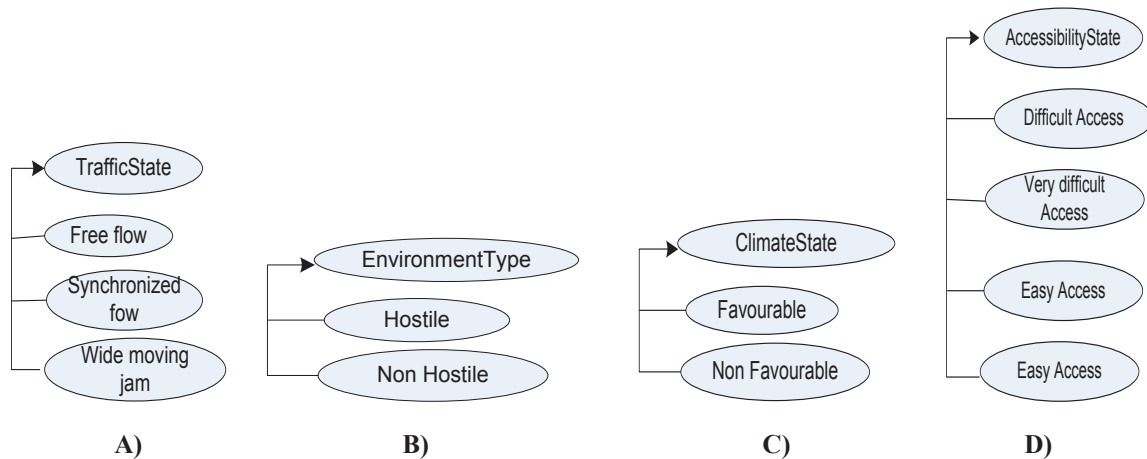


Figure 55. Classification de A) Etat du trafic routier, B) Type de l'environnement, C) Etat du climat, D) Etat d'accès

Contexte social

Il est très important pour nous de connaître si le patient est accompagné par une personne ou non et si cette personne est un professionnel de santé car cette information nous aide à attribuer un certain niveau de confiance à l'expéditeur du message. Elle contribue au choix du destinataire le plus adéquat à la situation médicale du patient. Pour cela, nous intégrons dans notre ontologie, le concept « accompagnement » qui peut prendre les valeurs « oui » ou « non » en fonction des valeurs brutes issues de capteurs de présence, par exemple (Figure 56).

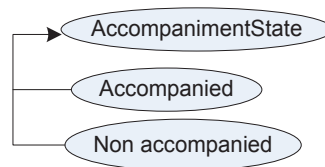


Figure 56. Classification de l'état d'accompagnement

Contexte géographique

Connaître la caractérisation de la distance qui sépare le patient expéditeur du message de son médecin traitant, destinataire du message joue un rôle très important dans le choix du destinataire surtout en cas d'urgence. Nous modélisons pour cela le concept « DistanceState » dans notre ontologie (Figure 57).

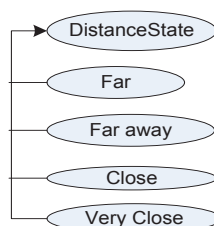


Figure 57. Classement des distances

Ontologie de l'écosystème source

Nous représentons maintenant l'ensemble de l'ontologie de l'écosystème source qui regroupe l'ensemble des concepts, leurs hiérarchisations et les relations qui les relient appelées *Object Properties* (Figure 58).

Comme le montre la Figure 58, le concept « Personne » est relié au concept « location » via la propriété « *located in* ». La localisation de la personne expéditrice du message est reliée aux concepts « *traffic State* », « *accessibilityState* » et « *locationState* » via les propriétés « *hasTrafficState* », « *hasAccessState* » et « *hasLocState* ». La personne est caractérisée par un état d'accompagnement (accompagné ou non accompagné). Elle possède une catégorie d'âge « *hasAgeCat* » qui peut prendre les valeurs « *elderly* » ou « *child* » ou « *adult* ». Chaque patient a un état clinique « *hasClinicalState* » et son historique clinique (facteurs de risques, paramètres biomédicaux, etc.). Nous proposons des hiérarchisations des différents concepts formant le contexte médical, tels que *Cardiac ClinicalSigns State*, *CardiacRiskFactors State*, *CardiacBiomedicalParameters*, et nous distinguons les différents états de normalité.

Des signes cliniques et des paramètres biomédicaux sont obtenus (*assessedBy*) par des appareils médicaux tels qu'un thermomètre, un électrocardiographe, un glucomètre, etc. D'autres données comme l'état climatique, l'état d'accompagnement, l'état du trafic sont plutôt déduits à partir de données brutes issues de dispositifs environnementaux tels que les capteurs de présence, les capteurs du trafic routier, etc. Nous modélisons pour cela dans notre ontologie deux catégories de dispositifs : dispositifs médicaux et dispositifs environnementaux.

Ecosystème message

Dans cette partie, nous proposons une modélisation ontologique de l'écosystème message (Figure 59). Nous mettons en lumière les attributs du HL7 RIM auxquels une politique d'acheminement peut être sensible. Ils apparaissent en italique dans la description qui suit. Comme la montre la figure, le concept « personne » est lié au concept « *ObservationHealthStatus* » via la propriété « *hasHealthStatus* ». Dans les spécifications du vocabulaire du HL7 RIM, il existe trois types de « *ObservationHealthStatus* » que nous modélisons dans l'ontologie : *Disable*, *Drug use*, *Pregnant*. Pour choisir le destinataire le plus adéquat à la situation médicale du patient, il est très important pour nous de savoir si le patient expéditeur du message est handicapé et de quel type d'handicap il souffre : un handicap mental, physique (handicap visuel, auditif, problème d'élocution). Par exemple, un message d'alarme moyenne issu d'un patient ayant des problèmes mentaux peut être envoyé à une centrale d'écoute pour vérifier l'exactitude de sa demande avant de l'envoyer au médecin traitant. Le même message peut être transmis directement au médecin traitant si la personne a des problèmes d'élocution. Pour cela, nous intégrons dans notre ontologie le concept « *PersonDisability* » ainsi que ses sous classes « *Vision impaired* », « *Hearing impaired* » « *Speaching impaired* » « *Mentally impaired* ».

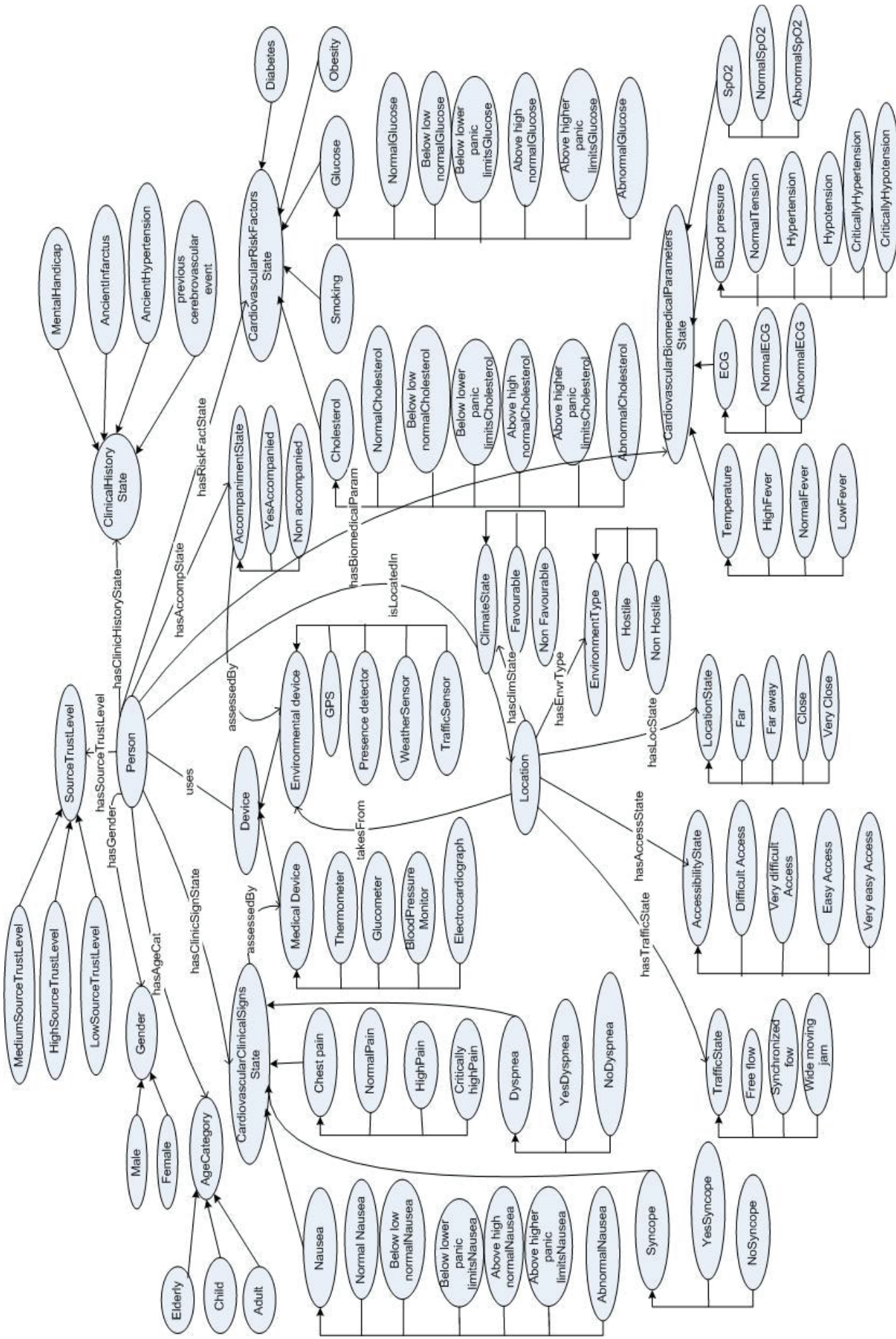


Figure 58. Ontologie de l'écosystème source

Dans notre ontologie, nous modélisons deux autres concepts « *LivingDependency* » et « *LivingSituation* » que nous jugeons aussi importants à connaître pour définir une politique d'acheminement personnalisable qui correspond exactement aux situations de l'expéditeur du message. En effet, savoir si le patient est dépendant de quelqu'un dans sa vie ou s'il vit tout seul ou en famille peut nous permettre de déterminer un certain niveau de confiance du patient. Par exemple, le délai requis pour le traitement d'un message issu d'un patient qui vit tout seul peut être beaucoup plus court que celui d'un patient qui vit avec sa famille.

Ces deux concepts représentent des attributs dans le HL7 RIM. Nous représentons leur hiérarchisation dans notre ontologie. Par exemple, sous le concept « *LivingSituation* », nous retrouvons les sous-classes « *alone* », « *Dependent children* », « *Dependent spouse* », « *Live with family* », « *Unrelated person* ».

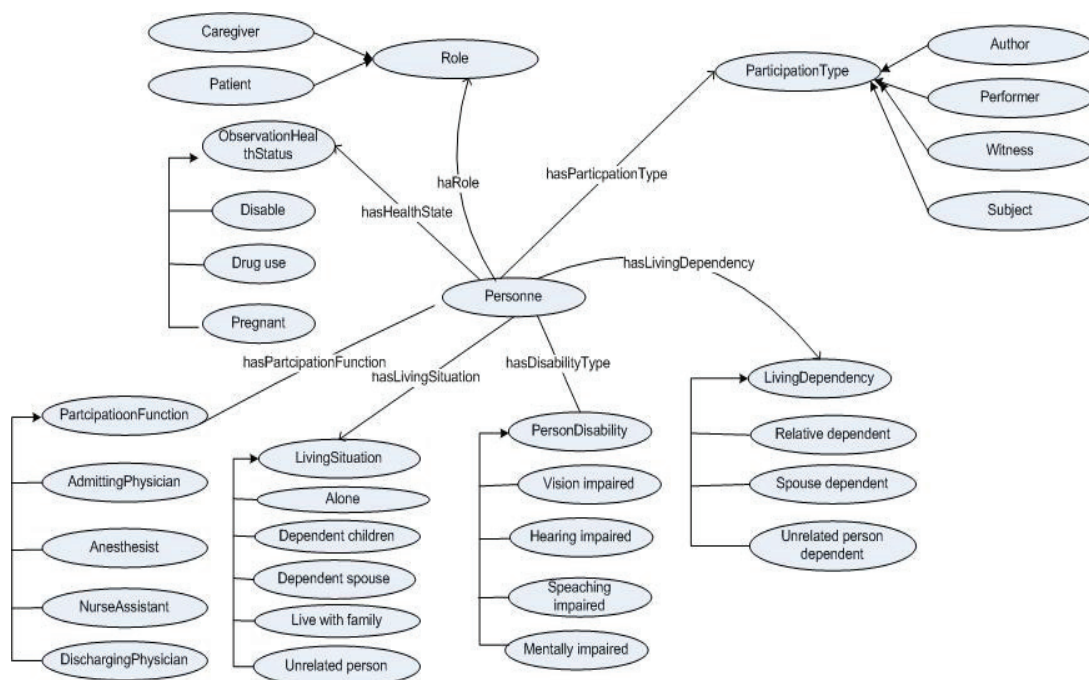


Figure 59. Ontologie de l'écosystème message

7.2.2 Raisonnement et inférence ontologique du contexte

Notre choix de l'approche ontologique était motivé par la possibilité d'effectuer un raisonnement sur le contexte et de pouvoir interpréter des données contextuelles élaborées à partir des données brutes issues directement des dispositifs ou des capteurs médicaux. Toute interprétation d'un état clinique d'un individu dépend de tout un ensemble de paramètres. Ce raisonnement est effectué via des règles logiques que nous formalisons avec le langage SWRL (*Semantic Web Rule Language*) [183].

Le niveau de confiance de l'acteur source est un attribut interprété qui appartient à l'écosystème source, mais il peut être déduit en fonction (f) des attributs de l'écosystème message appartenant au HL7 RIM (en italique), et en fonction d'autres attributs appartenant au même écosystème source. Le délai requis pour la lecture du message, par exemple, peut dépendre de l'expéditeur du message, s'il est un professionnel de santé (cardiologue) ou non,

s'il est habitué à envoyer des fausses alarmes, s'il est accompagné par un professionnel de santé ou non, etc. L'interprétation du niveau de confiance va ainsi dépendre du rôle joué par l'expéditeur du message dans le déclenchement de l'acte.

- C_1 : Si (*participation type = Author and Role class = patient*) Alors
Trust level = f (living dependency, person disability, living situation, accompanying)
- C_2 : Si (*participation type = Author and Role class = caregiver*) Alors
Trust level = f (participation function, number of false alarm, knowledge about patient illness)

L'attribut du RIM « *Participation function* » spécifie la fonction jouée par l'acteur participant dans l'acte. Il peut être un médecin traitant, un(e) aide-soignant(e), un(e) infirmier(ère), etc. Notons que le patient lui-même peut être un professionnel de santé. Dans ce cas, le niveau de confiance est calculé en fonction de la combinaison des deux conditions C_1 et C_2 .

7.3 Présentation des principaux flux d'information de notre architecture appliquée à l'e-santé

Dans cette partie, nous détaillons le contenu des flux d'interaction et des exceptions (Figure 60) entre les différents composants de l'architecture proposée dans le chapitre 5. Notre objectif est double : (1) prévoir le maximum de scénarios d'échange possibles (2) modéliser par la suite une vue dynamique de transfert des messages dans quelques exemples de scénarios.

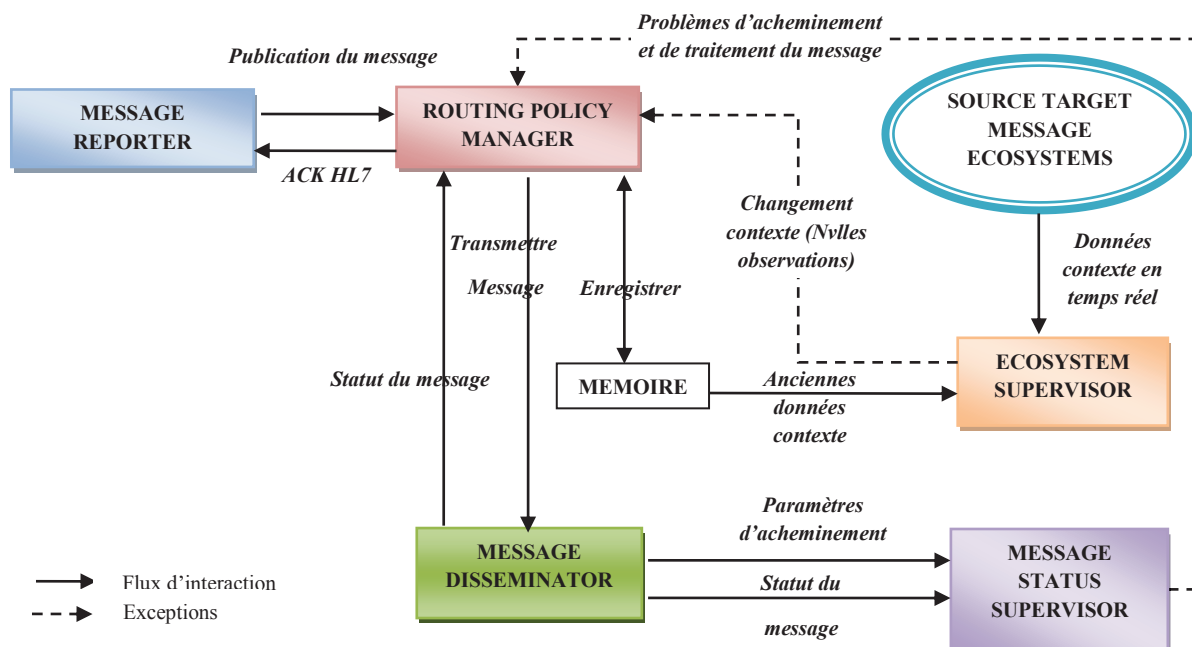


Figure 60. Représentation des flux d'interaction et des exceptions

Comme la montre la Figure 60, la publication du message HL7 représente la transaction à communiquer du *Message Reporter* au *Routing Policy Manager*. La structure du message est conforme au standard HL7 version 3.

Nous représentons dans ce qui suit un exemple d'extrait de message HL7 version 3 [184] qui concerne des résultats de laboratoire. L'acte dans ce cas représente une observation, le résultat du taux de glucose dans le sang qui est égale à 128 mg/dl. L'expéditeur du message GHH_LAB localisé à E-LAB3. Le destinataire est le système GHH_OE se trouvant à Bldg4. Le message HL7 version 3 est basé sur la notation XML où chaque label de sa structure constitue un attribut dans le RIM.

Ce message HL7 est composé de trois parties « *transmission wrapper* », « *Trigger Event Control act* » et « *HL7 domain content* ».

La « *Transmission wrapper* » représente toutes les informations concernant le message lui-même, sa transmission, et son acquittement, etc. L'élément racine indique l'identifiant du message qui est dans ce cas POLB_IN224200. Cet identifiant indique le type de message (résultat laboratoire), l'évènement déclencheur (*Trigger event*) et les responsabilités du récepteur du message. L'expéditeur et le destinataire du message sont décrits par les éléments *sender/device* et *receiver/device*.

```
<POLB_IN224200 ITSVersion="XML_1.0" xmlns="urn:h17-org:v3"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <id root="2.16.840.1.113883.19.1122.7" extension="CNTRL-3456"/>
  <creationTime value="200202150930-0400"/>
  <!-- The version of the datatypes/RIM/vocabulary used is that of May 2006 -->
  <versionCode code="2006-05"/>
  <!-- interaction id= Observation Event Complete, w/o Receiver Responsibilities -->
  <interactionId root="2.16.840.1.113883.1.6" extension="POLB_IN224200"/>
  <processingCode code="P"/>
  <processingModeCode nullFlavor="OTH"/>
  <acceptAckCode code="ER"/>
  <receiver typeCode="RCV">
    <device classCode="DEV" determinerCode="INSTANCE">
      <id extension="GHH LAB" root="2.16.840.1.113883.19.1122.1"/>
      <asLocatedEntity classCode="LOCE">
        <location classCode="PLC" determinerCode="INSTANCE">
          <id root="2.16.840.1.113883.19.1122.2" extension="ELAB-3"/>
        </location>
      </asLocatedEntity>
    </device>
  </receiver>
  <sender typeCode="SND">
    <device classCode="DEV" determinerCode="INSTANCE">
      <id root="2.16.840.1.113883.19.1122.1" extension="GHH OE"/>
      <asLocatedEntity classCode="LOCE">
        <location classCode="PLC" determinerCode="INSTANCE">
          <id root="2.16.840.1.113883.19.1122.2" extension="BLDG24"/>
        </location>
      </asLocatedEntity>
    </device>
  </sender>
  <!-- Trigger Event Control Act & Domain Content -- >
</POLB_IN224200>
```

Le *Trigger Event Control Act* » représente le déclencheur de l'évènement «*Trigger event*» POLB_TE224200. Cette partie du message donne des informations sur l'évènement déclenché, la date et le temps de sa transmission et les parties responsables de son déclenchement.

```
<controlActProcess classCode="CACT" moodCode="EVN">
  <code code="POLB_TE224200" codeSystem="2.16.840.1.113883.1.18"/>
  <subject typeCode="SUBJ" contextConductionInd="false">
  </subject>
</controlActProcess>
```

Le **HL7 domain content** décrit le contenu du message. La racine du message commence par l'élément « *observation event* ». Le reste des éléments indique le type de l'observation, son ID, son temps de détection, son statut « *status code* », sa priorité et les résultats. L'élément « *interpretation code* » indique la valeur (H), c'est-à-dire que la valeur de l'observation « *cholesterol* » qui est égale à 182 mg/dl est importante (*High*). Le **HL7 domain content** contient également des informations concernant la personne auteur de l'observation (identifiant, nom, prénom, etc.), qui est dans cet exemple de message « Mr. Harold H Hippocrates ». On trouve aussi des informations concernant la personne sur qui l'observation a été faite « Eve E. Everywoman » sous les éléments « *patientClinical* » et « *patientPerson* » du message HL7.

```
<observationEvent>
  <id root="2.16.840.1.113883.19.1122.4" extension="1045813"
    assigningAuthorityName="GHH LAB Filler Orders"/>
  <code code="1554-5" codeSystemName="LN"
    codeSystem="2.16.840.1.113883.6.1"
    displayName="GLUCOSE^POST 12H CFST:MCNC:PT:SER/PLAS:QN"/>
  <statusCode code="completed"/>
  <effectiveTime value="200202150730"/>
  <priorityCode code="R"/>
  <confidentialityCode code="N"
    codeSystem="2.16.840.1.113883.5.25"/>
  <value xsi:type="PQ" value="182" unit="mg/dL"/>
  <interpretationCode code="H"/>
  <referenceRange>
    <interpretationRange>
      <value xsi:type="IVL_PQ">
        <low value="70" unit="mg/dL"/>
        <high value="105" unit="mg/dL"/>
      </value>
      <interpretationCode code="N"/>
    </interpretationRange>
  </referenceRange>
  <author>
    <time value="200202150730"/>
    <modeCode code="WRITTEN"/>
    <signatureCode code="S"/>
    <assignedEntity>
      <id root="2.16.840.1.113883.19.1122.3" extension="444-444-4444"/>
      <assignedPerson>
        <name>
          <given>Harold</given>
          <given>H</given>
          <family>Hippocrates</family>
          <suffix qualifier="AC">MD</suffix>
        </name>
      </assignedPerson>
    </assignedEntity>
  </author>
  <recordTarget>
    <patientClinical>
      <id root="2.16.840.1.113883.19.1122.5" extension="444-22-2222"
        assigningAuthorityName="GHH Lab Patient IDs"/>
      <statusCode code="active"/>
      <patientPerson>
        <name use="L">
          <given>Eve</given>
          <given>E</given>
          <family>Everywoman</family>
        </name>
      <asOtherIDs>
```

```

        <id extension="AC555444444" assigningAuthorityName="SSN"
            root="2.16.840.1.113883.4.1"/>
    </asOtherIDs>
</patientPerson>
</patientClinical>
</recordTarget>

```

7.3.1 Acquittement de publication du message

Le message d'acquittement HL7 (ACK HL7) est une réponse d'acquittement sur le message d'origine précédemment envoyé par le Message Reporter au *Routing Policy Manager*. Il indique si le *Routing Policy Manager* a bien reçu le message HL7 du Message Reporter. La structure de cet ACK est conforme au standard HL7 version 3.

Des éléments d'acquittement s'ajoutent à la *structure du message HL7* comme *Acknowledgement detail code*, *Acknowledgement type*. Dans les tableaux qui suivent, nous représentons quelques valeurs possibles de ces deux éléments.

<i>Acknowledgement detail code</i>	Valeurs possibles
SYN	Erreur de syntaxe
. SYN102	Erreur de type de données
.. SYN101	Élément requis manquant
. SYN106	Erreur de terminologie
NOSTORE	Pas d'espace suffisant pour le stockage du message
....	

Tableau 5. Valeurs possibles de l'élément « *Acknowledgement detail code* » du message d'acquittement HL7

<i>Acknowledgement type</i>	Valeurs possibles
AA	Application Acknowledgement Accept
AE	Application Acknowledgement Error
....	

Tableau 6. Quelques valeurs possibles de l'élément « *Acknowledgement type* » du message d'acquittement HL7

7.3.2 Transmission du message vers le destinataire

Le *Routing Policy Manager* demande au *Message Disseminator* de transmettre le message vers le destinataire final sur son dispositif de communication (PC, téléphone portable, etc.).

7.3.3 Notifications du statut du message

L'état du message de communication correspond au statut du message après sa transmission par le *Message Disseminator*. Cette transaction est transmise du *Message Disseminator* vers le *Routing Policy Manager* et vers le *Message Status Supervisor*. Elle permet au RPM d'adapter ses décisions d'acheminement en conséquence si nécessaire.

Les valeurs possibles de cette transaction sont énumérées dans le tableau qui suit.

Valeurs possibles
Reçu par le <i>Message disseminator</i>
Non reçu par le <i>Message disseminator</i>
Reçu par le destinataire
Annulé par le destinataire
Rejeté par le destinataire
Accepté par le destinataire
Lu par le destinataire

Tableau 7. Enumération des valeurs du statut du message

La structure de la transaction « Etat du message de communication » est la suivante :

Usage	Structure de la transaction Statut du message
O	Identifiant du message
O	Valeur de l'état du message de communication

L'usage de l'« identifiant du message » et de la « Valeur de l'état du message de communication » est obligatoire (O).

7.3.4 Notifications des paramètres de gestion de l'acheminement du message

Outre le statut du message, le *Message Disseminator* transmet également au *Message Status Supervisor* la date et l'heure précise d'envoi du message au destinataire ainsi que les délais requis pour sa réception, sa lecture et son traitement. La structure de la transaction « Paramètres d'acheminement du message » est décrit dans le tableau ci-après.

Usage	Structure de la transaction « Paramètres d'acheminement du message »
O	ID Message
O	Délai requis pour la réception
O	Délai requis pour la lecture
O	Délai requis pour le traitement
O	Date et heure d'envoi AAAA : MM : JJ - HH :MM :SS

Tableau 8. Structure de la transaction « paramètres d'acheminement du message »

7.3.5 Notification des problèmes d'acheminement et de traitement du message

Cette transaction est transmise du *Message Status Supervisor* vers le *Routing Policy Manager*. Les valeurs possibles de cette transaction sont décrites dans le tableau ci-après.

Usage	Valeurs possibles de la transaction « problèmes d'acheminement et de traitement du message »
F	Non reçu dans le délai requis
F	Non lu dans le délai requis
F	Non traité (accepté/rejeté/annulé) dans le délai requis

Tableau 9. Valeurs possibles de la transaction « problèmes d'acheminement et de traitement du message »

L'usage de ces valeurs est facultatif (F). Le RPM ne sera notifié que lorsqu'il détecte des dépassements des délais.

7.3.6 Notification de changements de données contextuelles

Cette transaction est émise de l'*Ecosystem Supervisor* vers le *Routing Policy Manager* lorsqu'il y a un changement du contexte. L'*Ecosystem Supervisor* collecte les données contextuelles en temps réel, les compare avec les anciennes et détecte les changements s'il y en a. La structure de cette transaction contient la donnée contextuelle (l'attribut) et sa nouvelle valeur.

Usage	Structure de la transaction « changement du contexte »
F	Attribut
F	Valeur

7.4 Exemple de diagrammes de séquence dans le cadre d'un scénario d'acheminement des messages pour la surveillance des personnes âgées

Dans cette section, nous présentons un scénario de surveillance des personnes âgées illustrant le besoin d'implémenter une politique d'acheminement auto-adaptative en e-santé. Le scénario illustre la collaboration entre les composants de notre architecture dans le cas d'une situation normale, où la requête du patient est traitée par le destinataire (Figure 61) et de la survenue d'une exception : indisponibilité du professionnel de santé et changement de contexte (Figure 62). L'objectif de cette section est de représenter une vue dynamique en modélisant les flux de messages échangés entre les objets de notre architecture selon un point de vue temporel et en mettant l'accent sur la chronologie d'envoi. Pour cela, nous utilisons les diagrammes de séquence UML.

Scénario : Patient A est une personne âgée, télé-surveillée à domicile grâce à un système détecteur de chutes ou de fugues. Le fonctionnement de ce système est le suivant :

L'habitat du patient est instrumenté par des capteurs infrarouges détecteurs de mouvement, de passage, de présence, etc. Les informations issues des capteurs sont transmises à un ordinateur installé au domicile du patient. Le système fusionne ces informations multi-sensorielles. Il permet par la suite de diagnostiquer les mouvements du patient, de détecter tout événement anormal et prévenir le professionnel de santé (médecin, infirmier(ère)) ou l'entourage du patient.

Cas 1 : Acceptation de la requête

Pendant qu'il se déplaçait dans son habitat, le patient A a senti un malaise et est tombé par terre. Le système publie un message d'alarme au « RPM ». Dans ce cas, le système opère en tant que *Message Reporter*. Ce dernier envoie le message au « RPM » qui à son tour confirme la réception du message. Le « RPM » invoque l'*Ecosystem Supervisor* pour collecter des données contextuelles interprétées et/ou non interprétées, « *background* » ou « *real-time* » à partir de l'écosystème source (exemple d'attribut interprété et « *real-time* » : localisation géographique ; attribut interprété et *background* : historique de chute (fréquent)) et à partir de l'écosystème message (exemple d'attribut interprété et *real-time* : niveau d'importance du message). Le RPM combine les observations collectées et calcule la distribution de probabilités nécessaire pour inférer un profil type satisfaisant certaines critères, par exemple, le type de destinataire : médical, les ressources matérielles requises et les délais requis pour la réception, la lecture du message et sa réponse. Dans une liste préconfigurée de destinataires, le RPM cherche le profil le plus proche du profil type et associe le message du patient A à un professionnel de santé B. Le *Message Dissiminator* envoie un message au PDA du professionnel de santé B. B confirme la réception, la lecture et le traitement de message. Le patient A est notifié de l'acceptation de sa requête. Le *Message Status Supervisor* enregistre le niveau de confiance cible du médecin B et tout le cycle de vie du message avec les temps de réception, de lecture et d'acceptation.

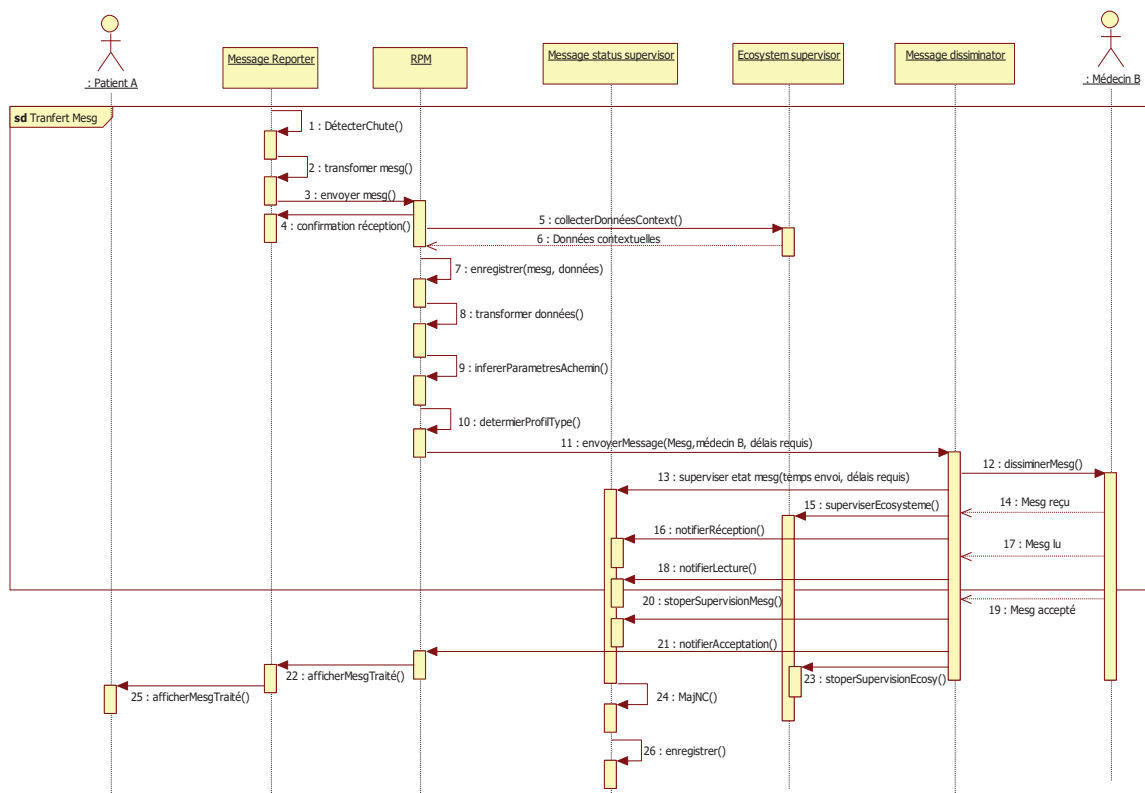


Figure 61. Diagramme de séquence d'acheminement d'une requête de prise en charge en cas de son acceptation par le médecin

Cas 2 : Indisponibilité du professionnel de santé + changement du contexte

Le médecin B confirme la réception du message dans le délai requis, mais il a dépassé le temps requis pour répondre à la requête du patient A. Entre-temps, une aggravation des signes cliniques a été détectée par l'Ecosystem supervisor. Le Message status supervisor et l'Ecosystem supervisor notifient le « RPM ». De nouvelles observations sont transmises au RPM qui adapte ses décisions en augmentant le niveau de priorité de message et en redirigeant le message vers la nouvelle destination d'urgence. Le « RPM » informe le patient sur ce changement. Le dépassement du délai de réponse à la requête du patient est enregistré dans l'historique (background) du médecin B. Ceci peut ainsi influencer son niveau de confiance et donc, des choix d'acheminement ultérieurs. On doit notamment prendre en compte les situations et les raisons pour lesquelles un médecin rejette la requête. Par exemple, le médecin B peut être indisponible à cause de son engagement dans un autre acte médical ou parce qu'il est en congé, etc.

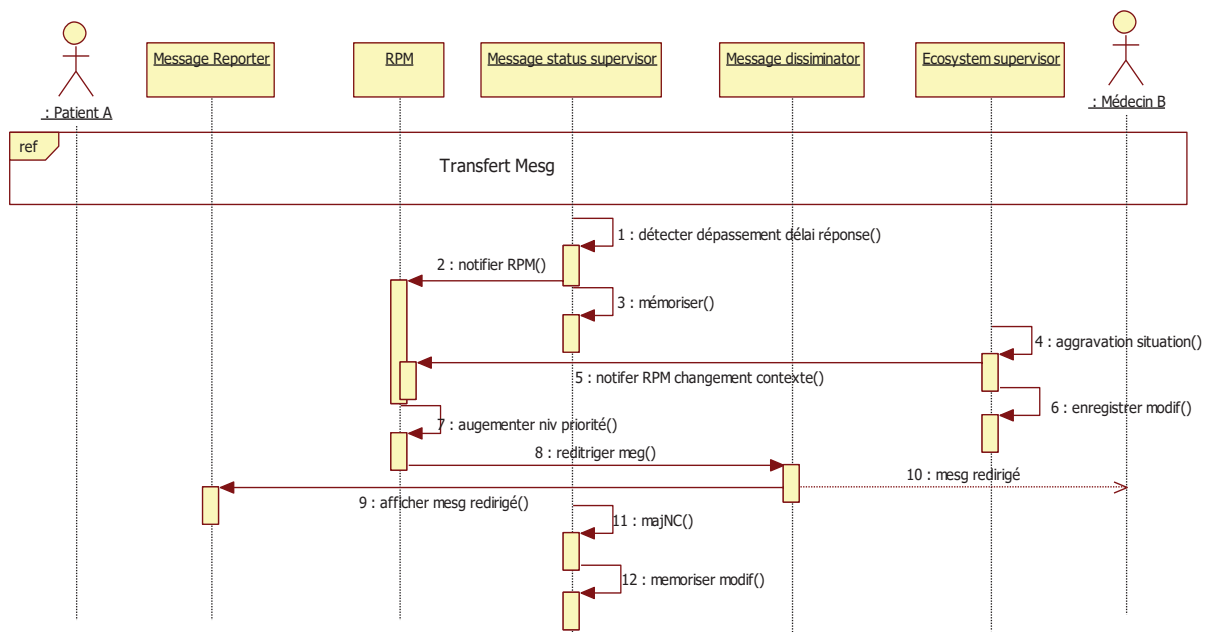


Figure 62. Diagramme de séquence d'acheminement d'une requête de prise en charge en cas d'indisponibilité du professionnel de santé et de changements du contexte.

7.5 Conclusion

En résumé, dans ce chapitre, nous avons proposé un cadre qui est compatible avec les spécifications du vocabulaire HL7 RIM. Pour mieux comprendre la notion d'écosystème et spécifier son contenu, auquel la politique d'acheminement est sensible, nous avons proposé une modélisation en couches, en UML et ontologique des écosystèmes sources, cible et message, appliquée au domaine de l'e-santé. Nous avons explicité le contenu des différentes transactions échangées entre les composants du cadre. Pour illustrer ces interactions, nous avons recouru à la modélisation des diagrammes de séquence UML appliqués à deux scénarios dans le domaine de l'e-santé. Le chapitre suivant sera dédié à évoquer l'aspect implémentation. Nous verrons ce que les réseaux bayésiens dynamiques et les réseaux de

Petri temporels peuvent nous apporter pour un « bon » pilotage de transfert de messages, intelligent, fiable et sûr.

Chapitre 8

Réseaux bayésiens dynamiques et réseaux de Petri temporels pour la gestion de l'acheminement des messages en e-santé

8.1 Introduction

Ce chapitre est afférent à l'aspect implémentation de l'architecture proposée dans le chapitre 5 dans le domaine de l'e-santé. Nous décrivons les méthodes les plus appropriées pour implémenter les services composants de l'architecture.

La première partie est consacrée à l'utilisation des Réseaux Bayésiens Dynamiques (RBD) pour l'inférence des paramètres d'acheminement dans le cadre de l'e-santé. La deuxième partie s'intéresse, quant à elle, à la modélisation dynamique des statuts des messages en utilisant les réseaux de Petri temporels. Nous expliquons, sans être exhaustif, les raisons de notre choix méthodologique via une étude comparative sur les méthodes que nous avons présentées dans le chapitre 4, comme les systèmes experts, les arbres de décision, l'algorithme des colonies de fourmis et les réseaux bayésiens.

8.2 Réseaux bayésiens dynamiques pour l'inférence des paramètres d'acheminement

8.2.1 Pourquoi les réseaux bayésiens dynamiques ?

Dans cette section, nous allons justifier notre choix méthodologique des réseaux bayésiens dynamiques pour implémenter le processus « inférence des paramètres d'acheminement ». Pour cela, nous présentons dans ce qui suit une étude comparative sur des méthodes déjà décrites dans le chapitre 3 eu égard aux besoins et contraintes qui découlent de la problématique d'acheminement des messages. Cette étude consiste à définir les avantages et les inconvénients de chaque méthode (réseaux bayésiens, systèmes expert, arbre de décision, réseaux bayésiens dynamiques) vis-à-vis de notre problématique. Le tableau suivant récapitule notre évaluation des méthodes. Les requis et contraintes de notre problématique que nous considérons sont : le temps et l'inférence à l'instant t sachant l'instant $t-1$, le raisonnement sur des données incomplètes, incertaines et imprécises, la représentation graphique de la dépendance conditionnelle entre les données contextuelles et les capacités de raisonnement flexible non déterministe.

Requis et contraintes	Méthodes d'aide à la décision				
	Réseaux bayésiens	Système expert	Arbre de décision	Colonies de fourmis	Réseaux bayésiens dynamiques
Temps et inférence à l'instant t sachant l'instant $t-1$					×
Raisonnement sur des données incomplètes, incertaines et imprécises	×			×	×
Représentation graphique de la dépendance conditionnelle entre les données contextuelles	×		×		×
Raisonnement flexible non déterministe	×			×	×

Tableau 10. Etude comparative des méthodes d'aide à la décision pour l'inférence des paramètres d'acheminement

Le temps : L'acheminement des messages est un processus dynamique et les décisions d'acheminement peuvent changer tout au long du cycle de vie du message. Les paramètres d'acheminement requis, comme la catégorie professionnelle du destinataire (médicale ou paramédicale), les délais requis pour le traitement de messages, évoluent également dans le temps dès l'entrée d'une nouvelle observation, telle que l'aggravation des signes cliniques, l'indisponibilité des ressources matérielles, etc. Les paramètres d'acheminement à l'instant t dépendent de ceux à l'instant $t-1$. Dans notre travail, nous nous n'intéressons pas au transfert du message seulement vers sa destination, mais nous prévoyons le cas du contexte dynamique qui fait que le message peut être redirigé à tout instant suite à la détection d'un changement des attributs observés (disponibilité, dépassement du délai requis pour la réception, la lecture ou bien le traitement de message, le contexte géographique ou climatique, etc.). Le processus est donc continu et l'estimation de la valeur de paramètres de la politique d'acheminement à un instant t dépend de la valeur de ces paramètres à l'instant $t-1$ et de leurs variations.

Les techniques basées sur des règles classiques telles que les systèmes experts, les arbres de décision peuvent être utilisées pour inférer les paramètres d'acheminement. Cependant, ces méthodes sont plus appropriées pour les applications statiques que pour les applications dynamiques. La modélisation stochastique des processus dynamiques, et plus particulièrement le formalisme des RBDs intègrent parfaitement la notion du temps. Ceci nous permet de faire apparaître des tendances dans l'évolution des données contextuelles dans le temps, un aspect très important dans notre problématique.

L'incertitude : Pour raisonner en fonction du contexte, le service d'inférence des paramètres d'acheminement doit tenir compte du problème de l'incertitude des données contextuelles. En effet, les données sources peuvent être sensibles et bruitées (données issues

de capteurs par exemple). Les données contextuelles peuvent donc être incertaines incomplètes ou imprécises.

Les systèmes experts et les arbres de décision ne tiennent malheureusement pas compte de cette contrainte d'« incertitude » vu qu'ils s'appuient sur un formalisme logique qui requiert l'observation de tous les faits pour la génération des conséquences. Un expert humain peut faire un jugement même lorsque les observations sont incomplètes ou imprécises. Un système expert ne peut pas effectuer cela alors qu'un modèle probabiliste le permet. Les RBs présentent ainsi un outil très puissant pour la représentation de connaissances incertaines et le raisonnement à partir d'informations incomplètes, grâce à son formalisme stochastique et à la modélisation probabiliste qu'il fournit.

Dépendance conditionnelle entre les données contextuelles : Les données contextuelles sont très dépendantes conditionnellement, et cette dépendance peut influencer le pilotage de l'acheminement des messages en e-santé. Par exemple, la donnée contextuelle « distance géographique entre patient et professionnel de santé » peut être influencé par l'existence ou non de la donnée climatique « neige ». Des données du contexte médical peuvent influencer le contexte social (patient accompagné par un professionnel de santé ou non). Ces deux données contextuelles peuvent affecter les règles d'acheminement. Par exemple, la compétence et la spécialité du professionnel de santé requis dépendent fortement du contexte clinique du patient. De plus, la valeur de la donnée contextuelle à l'instant t peut être influencée par la valeur de la même donnée, mais à l'instant $t-1$. Par exemple, le niveau de gravité des signes cliniques à l'instant t dépend de sa valeur à l'instant $t-1$.

Il serait donc intéressant d'avoir un (ou plusieurs) modèle(s) effectuant le lien entre les différentes observations contextuelles dans le temps et les attributs relatifs à la problématique d'acheminement. L'objectif est de permettre une représentation intuitive et claire du problème d'acheminement et de sa sensibilité au contexte, et cela, même lorsque les données contextuelles sont incomplètes et imprécises.

Les relations de dépendance existent dans les systèmes experts, mais d'une façon implicite. En effet, les règles (**Si** prémisses **alors** conséquence) dans les systèmes experts traduisent implicitement une relation de cause à effet. Un système expert ne fournit pas une représentation graphique claire de dépendance entre les différentes variables de la règle. De même, les colonies de fourmis ne se dotent pas d'un aspect graphique qui représente les relations entre les attributs du contexte. Au contraire, les arbres de décisions offrent une représentation graphique simple et compréhensible reflétant les relations logiques qui existent entre les différents nœuds de l'arbre.

Contrairement aux systèmes experts et aux colonies de fourmis, les réseaux bayésiens se sont révélés être des moyens extrêmement intuitifs et puissants pour la description des dépendances entre les différentes variables aléatoires du réseau grâce à la notion de séparation et aux tables de probabilités conditionnelles. Les réseaux bayésiens forment, comme nous l'avons vu dans le chapitre 3, une combinaison des approches probabilistes et de la théorie des graphes. Le réseau bayésien est avant tout un graphe dirigé acyclique. Le

modèle graphique probabiliste des réseaux bayésiens représente les relations entre les différents attributs clairement et intuitivement. Les orientations des arcs reliant les nœuds (s'ils existent) peuvent présenter une relation causale (de cause à effet) entre les différentes variables du réseau. Dans notre problématique, les graphes jouent un rôle très important. Ils permettent de fournir un moyen simple et efficace d'exprimer nos hypothèses, de donner une représentation intuitive et mathématique et des fonctions des probabilités conjointes et de faciliter l'inférence à partir d'observations. Outre l'aspect modélisation des dépendances entre les observations contextuelles, les réseaux bayésiens dynamiques permettent de relier les variables d'un pas de temps à un autre.

Modèle de raisonnement flexible et non déterministe : Nous avons besoin de raisonner en fonction du contexte afin de déduire la stratégie d'acheminement la plus adéquate à la situation à la fois du patient et du professionnel de santé. Ce modèle de raisonnement doit être flexible en fonction du changement des données contextuelles. En effet, Il n'y a pas toujours de règles strictes pour obtenir une stratégie d'acheminement. Par exemple, une même combinaison de données contextuelles peut aboutir à la même destination du message. Un intérêt majeur des réseaux bayésiens et des colonies de fourmis est qu'ils s'appuient tous les deux sur un formalisme probabiliste permettant un raisonnement flexible, non déterministe, ce que ne permettent pas les arbres de décisions ou les systèmes experts.

Dans un réseau bayésien, les relations entre les différentes variables sont probabilistes, basées sur le théorème de Bayes. Le modèle probabiliste ne modélise pas le mode de raisonnement que fait l'expert mais les connaissances qu'a l'expert sur les observations pouvant influencer le système. Ce mode de raisonnement est appelé raisonnement fondé. Ceci rend les réseaux bayésiens un outil très puissant pour un raisonnement flexible et non déterministe. Un tel modèle n'est donc pas un système expert au sens où est habituellement utilisé ce terme, les raisonnements effectués n'étant, par ailleurs, pas logiques, mais probabilistes. Ainsi, l'observation d'une ou plusieurs observations n'aboutit pas systématiquement à la conséquence mais modifie sa probabilité d'apparition. Au contraire, les systèmes experts sont plutôt moins flexibles. Ils sont plutôt fondés sur un raisonnement dit causal qui modélise le mode de raisonnement de l'expert en essayant de reproduire ce raisonnement sur de nouvelles requêtes. En pratique, le moteur d'inférence dans les systèmes experts effectue des déductions logiques basées sur des règles déterministes définies par l'expert lui-même. Dans notre problématique, il n'est pas évident de déterminer toutes les combinaisons possibles du contexte et prévoir toutes les règles d'acheminement qui en découlent. De plus, les données contextuelles peuvent évoluer continuellement.

8.2.2 Construction d'un réseau bayésien dynamique pour l'inférence du paramètre d'acheminement « type de destinataire »

Nous présentons dans la Figure 63 un exemple de réseau bayésien dynamique utilisé pour inférer la catégorie professionnelle requise dans le cadre d'un scénario en e-santé [179]. Comme illustré, la catégorie professionnelle « *Required staff Type* » (indiqué par le nœud E dans le RBD) est inférée en fonction de différentes données contextuelles provenant des écosystèmes source et message : le niveau de gravité des signes cliniques (fort/ faible/

moyen), le type du message (technique, alarme, informatif), le statut du message (redirigé, initial), et le niveau de gravité des antécédents cliniques (fort/ faible/ moyen). Ces données sont modélisées en tant que variables aléatoires dans le réseau, et peuvent changer de valeurs d'un pas de temps à un autre, comme le niveau de gravité des signes cliniques par exemple. La connaissance de la valeur de l'une des observations (A, B, D ou E) se propage sur le réseau, en générant une mise à jour des probabilités conditionnelles de chaque variable aléatoire (nœud) dans le graphe.

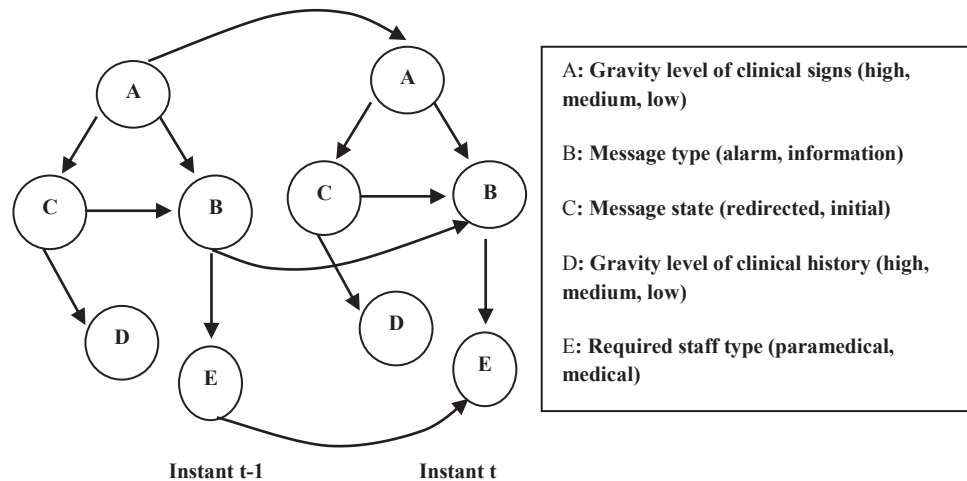


Figure 63 . Réseau bayésien dynamique pour l'inférence du type de staff

8.2.3 Construction d'un réseau bayésien dynamique pour l'inférence du paramètre d'acheminement « délai requis pour la lecture du message »

La Figure 64 présente un réseau bayésien dynamique que nous utilisons pour inférer le paramètre d'acheminement « délai requis pour la lecture du message » en fonction des autres variables constituant les données contextuelles du message : le niveau de gravité de l'historique clinique et des signes cliniques, la priorité de l'acte, le statut d'acheminement du message, le niveau de confiance [185]. Le réseau a deux pas temporels décrivant l'évolution des variables aléatoires (des nœuds x du réseau) du pas de temps $t-1$ au pas t , c'est-à-dire $P(X_t | X_{t-1})$. La dépendance entre les variables du RBD est représentée par les tables de probabilités conditionnelles. Le RBD reflète le processus d'évaluation temporelle de la stratégie d'acheminement en fonction de la dynamique caractérisant les données contextuelles. Par exemple, le niveau de gravité des signes cliniques (douleurs thoraciques, pression artérielle systolique et diastolique, température, interprétation ECG, etc.) à l'instant t peut dépendre de niveau de gravité des signes cliniques à l'instant $t-1$. Le délai requis pour la lecture du message à l'instant t peut aussi dépendre de sa valeur à l'instant $t-1$. Les liens reliant deux nœuds indiquent qu'il existe des relations probabilistes entre les états de ces deux nœuds. Par exemple, le niveau de gravité de l'historique clinique (ancien infarctus) augmente la probabilité du niveau de gravité des signes cliniques.

8.2.3.1 Choix des variables du RBD basé sur les attributs du HL7 RIM

Nous distinguons deux types de variables du RBD : interprétées (niveau de confiance, le niveau de gravité de l'historique clinique, les facteurs de risques et les signes cliniques), et non interprétées (la priorité de l'acte, le statut de l'acheminement du message). Ces variables regroupent un ensemble d'autres variables provenant de l'écosystème source et du contenu du message lui-même. Afin d'être conforme aux spécifications internationales du vocabulaire, nous avons basé le plus possible notre modèle sur le HL7 RIM. Dans la Figure 64, seule la variable « *Actpriority* » existe en tant qu'attribut dans le HL7 RIM. Mais, plusieurs variables du réseau sont interprétées à partir des attributs de HL7 RIM, comme le niveau de confiance (*Trust Level*), le niveau de gravité des signes cliniques et de l'historique clinique.

En outre, le délai requis pour la lecture du message va aussi dépendre de l'état du message (redirigé, initial, annulé) et de la priorité de l'acte « *act priority* » (facultatif, routine, urgent, très urgent).

8.2.3.2 Scénario de simulation du RBD dans le domaine de cardiologie

L'un des avantages du RBD est qu'il permet l'adaptation. Dans le scénario suivant, nous démontrons comment l'insertion d'une nouvelle connaissance (évidence) d'une variable dans le réseau peut ajuster les probabilités affichées dans la Figure 64, et donc influencer la valeur du paramètre d'acheminement.

Scénario : Le patient X a un historique de problèmes cardiaques. Il est allé en haute montagne pour skier. Il amène avec lui son dispositif médical intelligent (le CARDIAN® [1], par exemple). En arrivant, le patient ne se sent pas bien. Il enregistre son ECG et le dispositif détecte et envoie une alarme moyenne à son médecin traitant.

Puis, le patient X ressent des douleurs thoraciques et une ischémie cardiaque est détectée par son dispositif. Le niveau de gravité des signes cliniques a donc augmenté. Cette nouvelle connaissance constitue une évidence dans le RBD. Ainsi, cette évidence se propage dans le réseau et provoque des ajustements dans les tables de probabilités des nœuds du RBD. Cette mise à jour est réalisée grâce à l'algorithme du passage de message « *Forward-backward* ». Comme le montre la Figure 64, le niveau de gravité des signes cliniques devient important à 100%. Ainsi, les probabilités que le délai requis pour la lecture du message à l'instant t soit faible augmente.

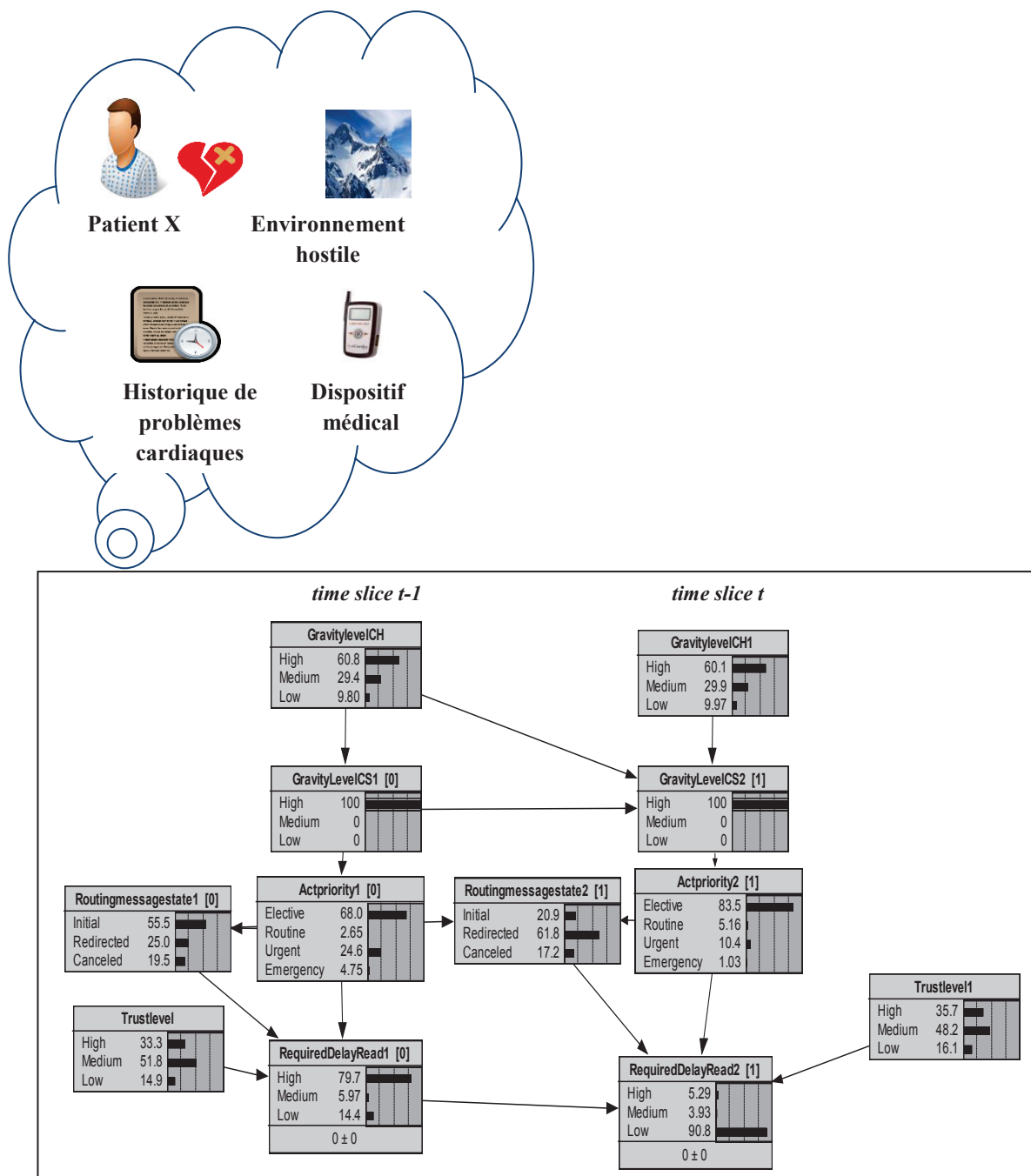


Figure 64. Simulation du RBD pour l'inférence du délai requis pour la lecture d'un message en e-santé dans le domaine de la cardiologie

8.3 Les réseaux de Petri temporels pour la supervision du statut des messages

8.3.1 Pourquoi les réseaux de Petri temporels ?

Pour superviser les statuts des messages (reçu, lu, acquitté, etc.) et pouvoir détecter les impasses, c'est-à-dire le blocage du message dans un état précis, par exemple en attente de réponse, nous avons besoin de modéliser la dynamique de l'état des messages dans le temps.

Cette modélisation va nous permettre par la suite de prendre les décisions de routage adéquates.

Différentes méthodes pour la modélisation dynamique des objets ont été développées, mais elles sont peu nombreuses à intégrer des méthodes d'analyse dans leurs propres descriptions, comme c'est le cas des réseaux de Petri temporels. Les diagrammes d'états-transitions, par exemple, bien qu'ils démontrent un aspect dynamique de l'état de message, elles ne permettent pas l'analyse des propriétés comportementales et structurelles essentielles à la validation du système.

Nous adoptons les réseaux de Petri temporels pour implémenter le service « Supervision du statut du message ». En fait, l'historique de l'enchaînement du message peut être modélisé par un Réseau de Petri temporel où les places représentent les statuts du message (en attente de réception / de lecture / de réponse) et les jetons représentent le message lui-même. Nous choisissons les réseaux de Petri temporels pour deux raisons principales : (1) Pour leur capacité d'éviter des impasses d'acheminement et de permettre ainsi la possibilité de rediriger les messages ; (2) Les RdPs temporels permettent, comme nous le verrons, de déterminer « le niveau de confiance » des professionnels de santé, qui est considéré comme l'un des critères importants à prendre en compte lors de la recherche du « bon » destinataire.

Dans les sections qui suivent, nous présentons dans un premier temps un diagramme d'états-transitions modélisant les différents états par lesquels passe un message. Nous intégrons par la suite dans cette modélisation un aspect analytique au moyen des réseaux de Petri temporels pour la supervision des statuts du message.

8.3.2 Modélisation du statut des messages

Pour détecter les événements imprévus de dépassement des délais de traitement du message, nous devons modéliser tout d'abord les états du message depuis sa création jusqu'à son traitement. Le diagramme d'états-transitions UML, Figure 65, décrit le comportement du message à l'aide d'un automate à états finis.

Comme la montre la Figure 65, à l'état initial, le message est en mode création. Lorsqu'un expéditeur (patient, professionnel de santé, etc.) transmet le message à un destinataire, le message passe à l'état « envoyé ». Le message est dit « reçu » si et seulement si le destinataire envoie un acquittement de réception. Si non, le message est dit « non reçu dans le délai requis », et il sera envoyé vers un autre destinataire. A la réception du message, le professionnel de santé (destinataire) peut : répondre directement à la requête, lire le message, ne pas lire le message dans le délai requis. Dans ce dernier cas, le message sera également renvoyé vers un autre destinataire. La réponse peut avoir trois états : accepté, en attente et rejeté. Des changements de contexte peuvent également survenir au moment de l'attente. Dans ce cas, le message peut être renvoyé ou redirigé vers une nouvelle destination. Notons que le message peut être annulé à tout moment quel que soit son statut.

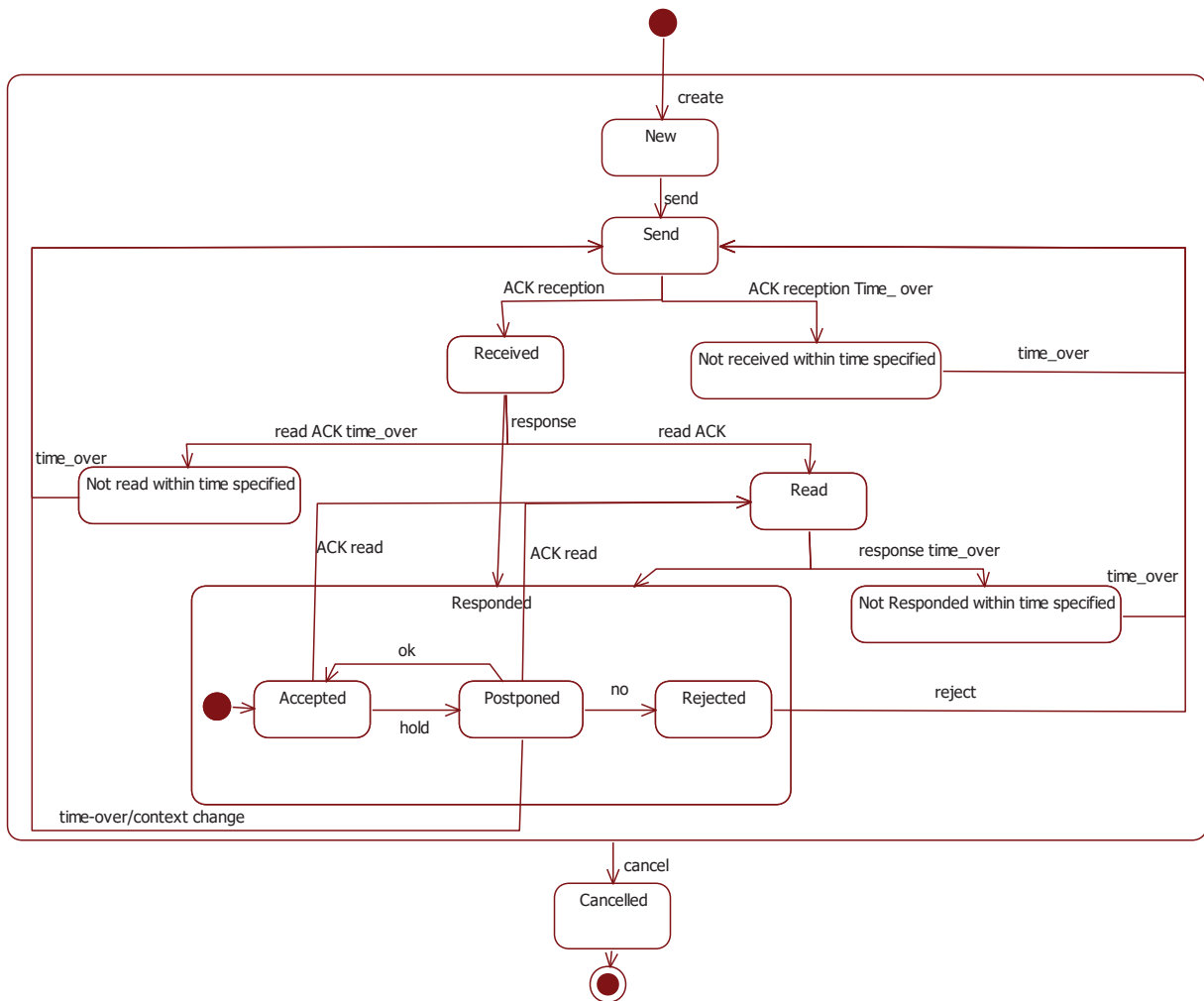


Figure 65. Diagramme UML d'états-transitions pour la modélisation des états du message

8.3.3 Modélisation du Réseau de Petri temporel

La Figure 66 représente un extrait du RdP temporel que nous avons modélisé illustrant les différences du temps d'attente entre trois professionnels de santé. Au temps t_0 , le jeton est déposé dans la place P, mais il reste indisponible. Après d unités de temps, le jeton devient disponible et la transition t' devient franchissable. Dans cet exemple, nous supposons que le temps requis pour recevoir une réponse positive est au minimum 30 min et il ne doit pas dépasser 2 heures. Comme illustré, le premier professionnel de santé prend moins de temps pour répondre positivement à la requête de son patient (le jeton quitte la place P après une heure) en comparaison avec les deux autres professionnels de santé. Contrairement au troisième qui dépasse le délai requis pour lire le message, ce qui diminue son « niveau de confiance » pour des transmissions ultérieures de requêtes de prise en charge médicale. Evidemment, nous devons aussi considérer les raisons pour lesquelles les médecins ont des réponses tardives (en raison d'engagement dans un autre acte médical, de congé, etc.).

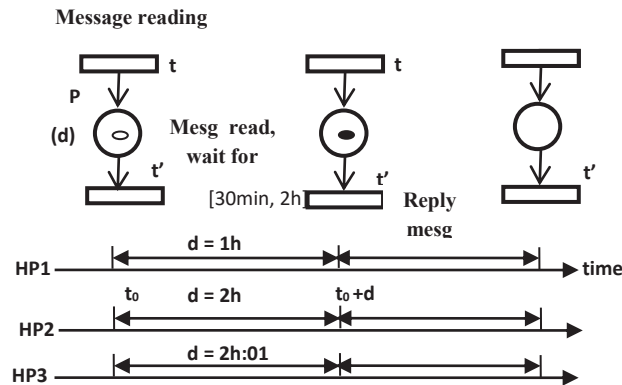


Figure 66. Réseau de Petri temporel pour la modélisation du statut du message

8.4 Conclusion et discussion

En résumé, ce chapitre constitue un premier pas vers l'implémentation d'un prototype fondée sur l'architecture proposée dans le chapitre 3. Nous avons effectué une étude approfondie et comparative de quelques méthodes d'aide à la décision. Les réseaux bayésiens dynamiques semblent être une voie prometteuse pour l'inférence des paramètres d'acheminement et l'amélioration de la qualité de services de transmission de messages en e-santé. Nous avons également montré l'intérêt des réseaux de Petri temporels en tant que modèle mathématique de la dynamique du statut des messages en e-santé et de détection des impasses qui peuvent survenir lors de l'acheminement d'un message.

Chapitre 9

Application de notre méthodologie à l'e-cardiologie et illustration de son fonctionnement. Discussion

9.1 Introduction

L'objectif de ce chapitre est de concrétiser notre solution à travers un scénario réel issu du domaine de la cardiologie, et d'apporter des éléments de discussion quant à la mise en œuvre opérationnelle de notre approche méthodologique. Nous proposons l'exemple de l'acheminement d'un message d'alarme moyenne généré par le CARDIAN[®] (ou PEM) présenté dans le chapitre 2, un dispositif médical intelligent qui permet de détecter l'occurrence d'un événement électrique cardiaque anormal lors de l'enregistrement par le patient de son propre ECG, et de générer des alarmes en conséquence. Le déroulement de l'acheminement de ce message d'alarme est démontré, dans la section suivante, à travers un diagramme de collaboration BPMN. Un diagramme de collaboration se compose d'un ensemble de participants représentés par des bassins, appelés *pools*. Les *pools* peuvent inclure des processus métiers et interagissent entre eux avec des flux de messages. A travers ce diagramme, nous illustrons, le déroulement de l'acheminement d'un message d'alarme depuis sa génération par le CARDIAN[®] jusqu'à son traitement par un professionnel de santé, tout en montrant les différents participants impliqués dans la gestion d'acheminement du message ainsi que les flux de messages échangés entre eux. Les participants représentent, dans notre cas, les modules composant notre architecture, modules que nous avons présentés dans le chapitre 5 de ce manuscrit.

La mise en œuvre de notre système soulève des défis sur les plans médico-social, juridique, socio-économique, sécurité et préservation de la vie privée, etc. Nous discutons tous ces aspects et présentons notre vision eu égard à l'implémentation de notre approche dans la troisième section de ce chapitre.

9.2 Instanciation de notre méthode de gestion d'acheminement d'un message d'alarme moyenne dans un contexte de « self-care » en cardiologie

Dans cette section, nous présentons, tout d'abord, un exemple de scénario de message d'alarme moyenne généré par le CARDIAN[®] dans le contexte d'utilisation décrit dans le projet EPI-MEDICS. Nous montrons par la suite le déroulement de l'acheminement de ce

message par notre système - le fruit de nos travaux de thèse -au moyen d'un diagramme de collaboration modélisant les processus métiers et les échanges de messages entre les différents modules de notre architecture.

Le diagramme de collaboration présenté Figure 67 se compose de 6 participants ou bassins : (1) l'écosystème source (*Source Ecosystem*) comporte le CARDIAN[®] (*Message Reporter*) et le patient, (2) l'écosystème cible (*Target ecosystem*) contient les destinataires figurés dans le scénario ci-après, (3) le *Routing Policy Manager* (RPM), (4) le *Message Disseminator*, (5) le *Message Status Supervisor* et (6) l'*Ecosystem Supervisor*. Chaque participant exécute un processus. Ici, nous montrons seulement les processus exécutés par le patient, le CARDIAN[®], le RPM, et le *Message Disseminator*. Les processus exécutés par le *Message Status Supervisor* et l'*Ecosystem Supervisor* ont déjà été présentés dans le chapitre 6 de ce mémoire. Pour des raisons de lisibilité, nous détaillons le sous-processus « *search for relevant healthcare professionnal* » et ses interactions en Figure 68, dans un sous-diagramme de collaboration à part du diagramme principal Figure 67. Ces diagrammes montrent comment notre système de gestion de routage des messages se comporte en cas de refus de la requête par le professionnel de santé, de dépassement de délai requis pour le traitement du message et enfin de changement du contexte clinique du patient. Ils permettent de répondre aux questions suivantes : quels sont les participants impliqués dans cette gestion ? et quels sont les flux de messages qu'ils échangent ?

9.2.1 Scénario d'alarme moyenne

Nous supposons le scénario suivant :

Un patient A dispose du CARDIAN[®] lui permettant de détecter des alarmes en cas de problèmes cardiaques. Le dossier médical du patient A est embarqué dans le CARDIAN[®] et est composé des données médicales suivantes :

- Antécédents cliniques : présence d'un infarctus ancien.
- Facteurs de risques : fumeur, diabétique, obèse.
- Paramètres biomédicaux : tension dans la limite des seuils de normalité, présence d'un ECG de référence annoté et revu par le cardiologue du patient.

Nous supposons que le patient A possède une liste préconfigurée de destinataires dont les *background attributes* et les *real-time attributes* sont décrits dans les tableaux 11 et 12 présentés et commentés ci-après. Cette liste de destinataires est enregistrée dans la carte mémoire du CARDIAN[®] et provient des préférences du patient, de ses contacts, de son dossier médical, etc.

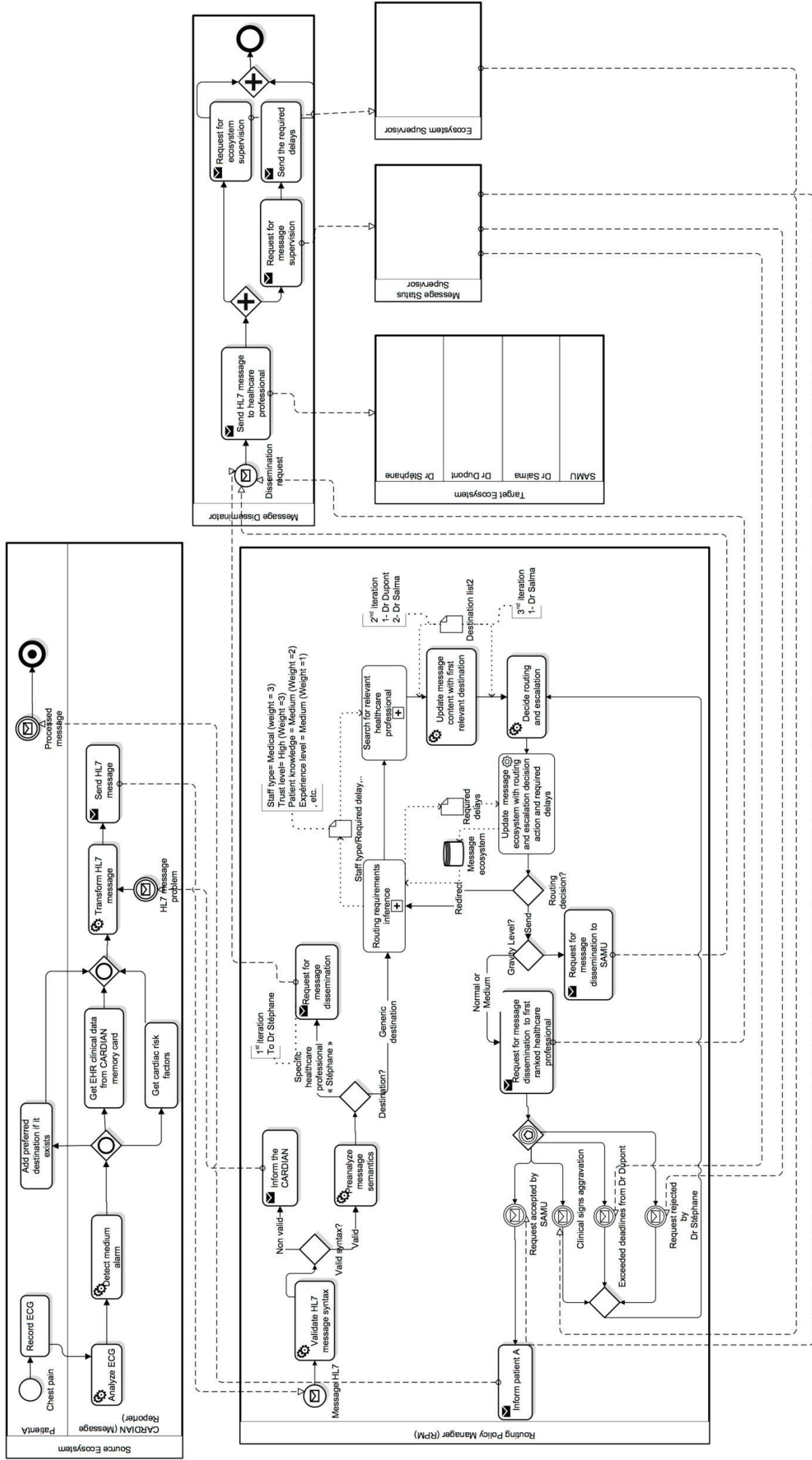


Figure 67. Diagramme de collaboration du processus de gestion d'acheminement d'une alarme moyenne en cardiologie (alarme issue d'un CARDIAN®)

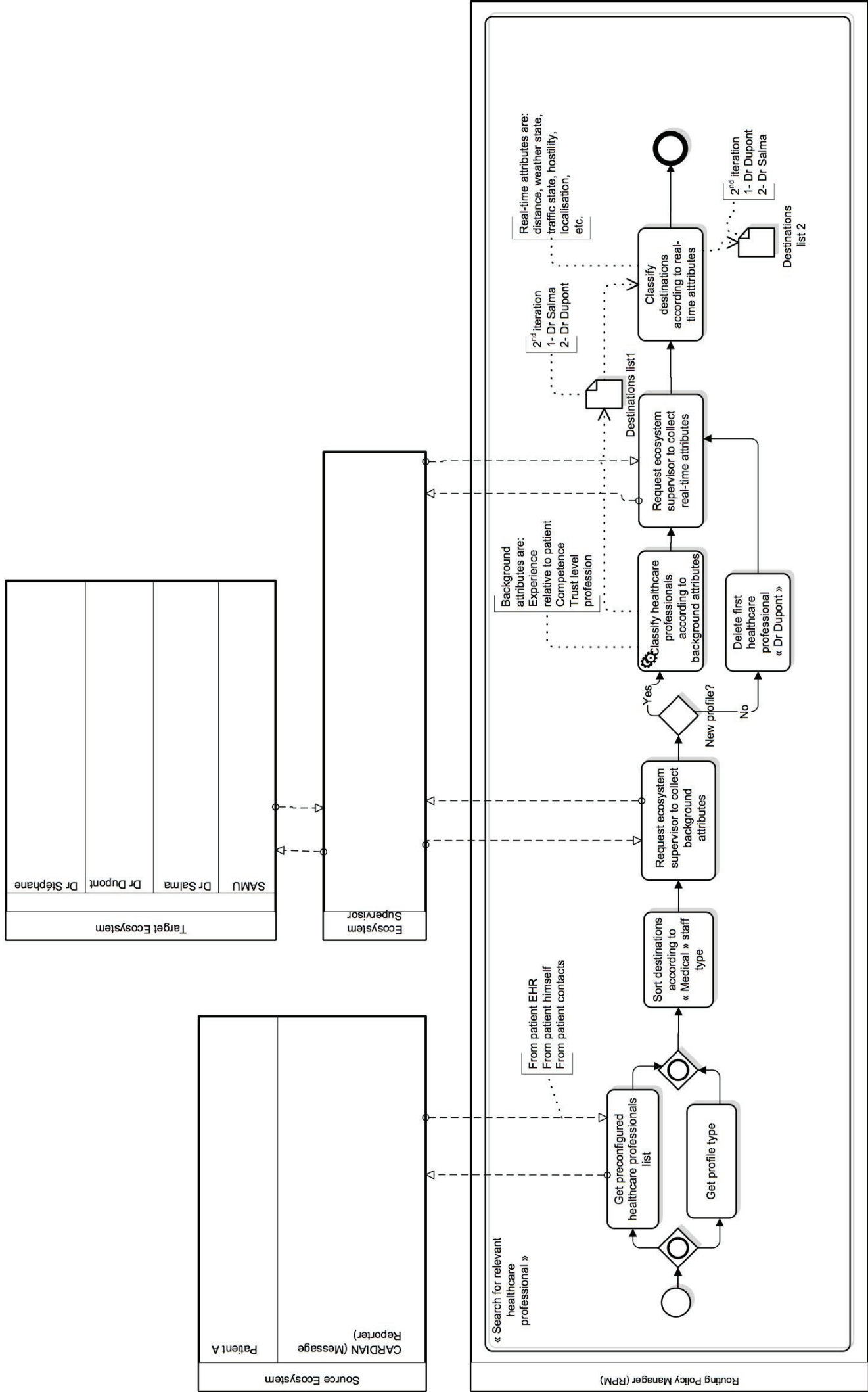


Figure 68. Diagramme de collaboration du sous-processus « Search for relevant healthcare professional » du RPM appliqué à un scénario en cardiologie

Destinataires	Critères de type « <i>Background attributes</i> »				
	Profession	Catégorie professionnelle	Niveau de connaissance du patient	Niveau d'expérience (ancienneté)	Niveau de confiance ³
Dr Stéphane	Médecin traitant	Médical	Fort	Fort	Fort
Dr Dupont	Cardiologue	Médical	Fort	Fort	Moyen
Dr Salma	Médecin généraliste	Médical	Moyen	Fort	Fort
Ambre	Infirmier	Paramédical	Moyen	Moyen	Moyen

Tableau 11. Liste préconfigurée des destinataires du patient A et leurs critères de sélection de type « *Background attributes* »

Destinataires	Critères de type « <i>Real-time attributes</i> »	
	Distance	Accessibilité
Dr Stéphane	Très proche	Très facile
Dr Dupont	Proche	Très facile
Dr Salma	Loin	Difficile
Ambre	Proche	Difficile

Tableau 12. Liste préconfigurée des destinataires des messages provenant du CARDIAN® du patient A et leurs critères de sélection de type « *Real-time attributes* »

Notons, ici, que le choix de ces critères n'est pas exhaustif et qu'il doit être établi et/ou validé par des médecins spécialistes du domaine pour chaque type de scénario. Notre cadrage est en effet ouvert à l'implémentation de différents ensembles de critères pour répondre à la diversité des cas d'usage. Les valeurs des critères indiquées dans les tableaux 11 et 12 sont fortuites et ont été choisies pour leur seule potentialité à montrer le fonctionnement de notre système sur un exemple.

Dans le Tableau 11, le critère « profession » indique si le destinataire est un médecin traitant, cardiologue, médecin généraliste, infirmier ou autre, etc. La catégorie professionnelle détermine si le destinataire potentiel appartient au personnel médical ou paramédical. Le niveau de connaissance du professionnel de santé est relatif à sa connaissance de la situation médicale du patient, en d'autres termes au nombre de consultations qu'il a effectuées pour ce patient et depuis combien d'années. La médecine, en effet, n'est pas une science exacte, et est connue pour être un domaine extrêmement complexe, face à l'hétérogénéité du vivant et des signes cliniques. La précision diagnostique est de façon notoire grandement améliorée lorsque le médecin dispose de tout l'historique clinique du patient, ce qui est d'ailleurs l'un des intérêts majeurs de l'e-santé, afin que l'interprétation des résultats d'un examen puisse être réalisée en référence aux données antérieures du patient, et donc de façon spécifique à sa situation clinique. Le niveau d'expérience relève, ici, de l'ancienneté du professionnel de santé dans le domaine de la cardiologie. Le niveau de confiance assigné au professionnel de santé dans cet exemple est uniquement déduit à partir de son historique comportemental eu égard à sa disponibilité et à ses temps de réponse observés antérieurement. Un médecin ayant un niveau de confiance très fort, comme « Dr Salma » dans cet exemple, a l'habitude de

³ eu égard à la capacité de répondre instantanément à une demande si nécessaire

répondre fréquemment et rapidement aux requêtes de prise en charge qu'il reçoit de ses patients.

Dans le Tableau 12, les critères de type *real-time attributes* sont : la distance qui sépare le professionnel de santé du patient A et l'accessibilité du site du professionnel de santé. Le choix de ces critères est justifié par le fait que nous supposons qu'il peut y avoir une forte probabilité de transfert du patient A, et qu'il est donc important que les facteurs environnementaux et de géolocalisation soient favorables. L'accessibilité de l'environnement dépend de plusieurs facteurs dont le type de l'environnement (isolé ou non, qualité des infrastructures routières, densité du trafic routier, état climatique, etc.). Un professionnel de santé distant peut arriver plus tôt pour la prise en charge de son patient A qu'un autre qui est proche s'il n'y a pas de problèmes d'embouteillage ou d'autres obstacles environnementaux qui empêchent son déplacement.

Le scénario ou cas d'utilisation que nous avons choisi est le suivant : le patient A, qui a des antécédents cardiovasculaires et vit à son domicile, dispose du CARDIAN[®], sur les recommandations de son cardiologue, pour s'auto-enregistrer son propre ECG au moins deux fois par mois, et surtout dès qu'il ne se sent pas très bien afin de transmettre son ECG numérique pour contrôle. Le patient A éprouve des douleurs thoraciques, alors qu'il est en train de faire un peu de gymnastique d'entretien dans son jardin. Suite à l'enregistrement de l'ECG, un message d'alarme moyenne est généré par son CARDIAN[®]. Suivons maintenant le déroulement de l'acheminement de ce message en présence de notre système.

9.2.2 Diagramme de collaboration appliqués à un scénario d'acheminement d'un message d'alarme dans un contexte de « *self-care* » en cardiologie (Figure 67).

- Le patient A ressent des douleurs thoraciques de façon transitoire.
- Il enregistre son ECG.
- Le CARDIAN[®] analyse l'ECG qui vient d'être enregistré en faisant référence à un ECG antérieur, les antécédents cliniques, les facteurs de risque, les signes cliniques et les paramètres biomédicaux du patient A (ensemble d'éléments d'information qui sont stockés dans la carte mémoire du dispositif), et détecte des modifications des signaux ECG pouvant être, avec une probabilité de 50%, significatives d'un dysfonctionnement électrique cardiaque, et donc de l'apparition d'un nouvel incident cardiovasculaire. Les résultats d'analyse du CARDIAN[®] correspondent à une alarme qualifiée de « moyenne », c'est-à-dire ne nécessitant pas l'appel immédiat du SAMU, mais néanmoins le contrôle par un médecin de l'ECG enregistré dans des délais convenables, afin qu'il oriente en conséquence le patient vers une prise en charge médicale pour d'éventuels examens complémentaires et pour infirmer ou confirmer la survenue d'une anomalie d'origine cardiovasculaire.

- Le CARDIAN[®] collecte toutes les données cliniques du dossier médical du patient enregistrées dans la carte mémoire du CARDIAN[®] et construit un message HL7 correspondant à une alarme moyenne.
- Le CARDIAN[®] transmet le message HL7 au RPM.
- Le patient A, contrarié par ses symptômes, n'ajoute pas au message de destinataire préféré.
- Le RPM valide la syntaxe du contenu du message HL7 (contrôle d'erreurs sur les types de données, vérification de la non-existence de données manquantes, etc.) et, en cas de détection de problèmes, en informe le CARDIAN[®] pour qu'il invoque la vérification de son processus de construction du message.

1^{ère} itération :

- Le message étant validé, le RPM invoque le *Message Disseminator* pour envoyer le message d'alarme à « Dr Stéphane », le médecin traitant du patient A.
- Une fois le message envoyé, le *Message Disseminator* invoque l'*Ecosystem Supervisor* et le *Message Status Supervisor* pour superviser l'état du message et pour détecter d'éventuels changements de contextes en cours d'acheminement.

Occurrence d'une exception : le Dr Stéphane refuse le message parce qu'il est occupé par une consultation dans son cabinet qui nécessite une attention particulière qu'il ne peut interrompre.

- Le *Message Status Supervisor* informe le RPM.
- Le RPM décide alors de rediriger le message HL7 vers un autre destinataire. Mais vers quel destinataire ? et quel est le temps requis pour répondre à ce message ?
- Le sous processus « *Routing requirements inference* » du RPM collecte des données contextuelles (antécédents cliniques, signes cliniques, facteurs de risques, le type du message (alarme moyenne)) du patient A. Il infère au moyen de l'algorithme *forward-backward* et d'un réseau bayésien dynamique, des probabilités P d'estimation des paramètres d'acheminement requis tels que : P (catégorie professionnelle requise = médical) = 1, P (délai requis pour la réponse au message = Faible) = 0,8 (ceci correspond à 1/4 d'heure de délai requis), P (niveau d'expérience requise = Fort) = 0,3, P (niveau de connaissance requis = fort) = 0,3, P (niveau de confiance requis = très fort) = 1. Nous choisissons dans cette exemple ces valeurs arbitrairement mais néanmoins conformes à la nature du type du scénario et au type de message (message d'alarme moyenne). Ces probabilités contribuent à affecter des poids à chaque critère de sélection des destinataires. Le niveau de confiance et la catégorie professionnelle requis sont associés au poids le plus élevé (poids = 3 par exemple) car il y a un risque à confirmer ou infirmer pour le patient, et l'interprétation automatique de l'ECG est

fournie par le CARDIAN[®] avec une probabilité de 50%. Rappelons en effet que l'interprétation automatique comme l'interprétation humaine de signes vitaux est une tâche extrêmement difficile et complexe et n'est jamais garantie à 100% d'exactitude [40]. Le niveau requis de la connaissance relative au patient est moyenne mais néanmoins souhaitable (poids = 2) et enfin le niveau d'expérience demandé au médecin peut être celui d'un médecin généraliste, non spécialisé en cardiologie (poids = 1). Cette inférence bayésienne suppose en résumé que dans le cadre de ce scénario, le message d'alarme moyenne soit reçu par un professionnel de santé qui soit médecin, habituellement reconnu apte à traiter les messages du patient A dans des délais assez brefs (d'après l'historique de son activité de prise en charge télé-médicale par notre système), ayant une excellente connaissance du dossier médical du patient A, et enfin une expérience (expertise en cardiologie/ancienneté) plus ou moins importante.

- Au niveau du RPM, le sous processus « *Search for relevant healthcare professional* » reçoit à partir du sous processus « *Routing requirements inference* » les paramètres d'acheminement requis préalablement inférés. Il invoque alors le CARDIAN[®] pour recevoir la liste préconfigurée des destinataires potentiels des messages du patient A avec leurs attributs de type *background* et *real-time* de l'écosystème source (Tableau 11 et Tableau 12).
- A partir de cette liste de destinataires potentiels, le sous processus « *Search for relevant healthcare professional* » sélectionne les professionnels de santé dont la catégorie professionnelle est : « médical », c'est-à-dire « Dr Dupont » et « Dr Salma ». Dr Stéphane est supprimé de la liste parce qu'il n'a pas accepté la requête de prise en charge. Ambre est aussi exclue de la liste puisqu'elle est une infirmière (catégorie professionnelle = paramédical) (Figure 68).
- Le sous-processus « *Search for relevant healthcare professional* » du RPM invoque l'*Ecosystem Supervisor* pour collecter les *background attributes* à partir de l'écosystème cible et classe la liste des destinataires selon leurs attributs. Après exécution du service « *Classify healthcare professional according to background attributes* », Dr Salma arrive en tête de la liste des destinataires puisqu'il possède un niveau de confiance plus fort que Dr Dupont (Figure 68).

Liste des professionnels de santé appartenant à la catégorie « médical » classée selon les valeurs des « <i>background attributes</i> » (<i>Destinations list1</i>)	
1	Dr Salma
2	Dr Dupont

- Le sous-processus « *Search for relevant healthcare professional* » du RPM classe ensuite cette dernière liste selon les *real-time attributes*. Dr Dupont passe alors en tête de la liste, car il est plus proche en temps de déplacement du patient A que Dr Salma (Figure 68), au cas où une consultation serait nécessaire.

	Liste des professionnels de santé de type « médical » classés selon les valeurs des <i>background attributes</i> (<i>Destinations list2</i>)
1	Dr Dupont
2	Dr Salma

2^{ème} itération :

- Le RPM ré-invoque le *Message Disseminator* pour transférer le message à « Dr Dupont » et re-invoque le *Message Status Supervisor* et l'*Ecosystem Supervisor*.

Occurrence d'une deuxième exception : Au bout d'un quart d'heure (le délai requis pour la réponse au message préalablement inféré par le réseau bayésien dynamique dans le sous processus « *Routing parameters inference* »), Dr Dupont n'a toujours pas répondu au message d'alarme du patient A.

- Le *Message Status Supervisor* informe le RPM de cette nouvelle exception.
- Le RPM décide alors de rediriger le message.
- Le RPM invoque le service « *Routing parameters inference* » pour inférer de nouvelles valeurs de paramètres. Nous supposons que le réseau bayésien infère les mêmes valeurs de paramètres d'acheminement, c'est-à-dire le même niveau de confiance (très fort), et la même catégorie professionnelle (Médical).
- Le service « *Search for the relevant healthcare professional* » supprime Dr Dupont de la liste des destinataires puisqu'il n'a pas répondu.

3^{ème} itération

- Le RPM invoque le *Message Disseminator* pour transmettre le message à « Dr Salma ».

Occurrence d'une troisième exception : une aggravation des signes cliniques est détectée par l'*Ecosystem Supervisor*.

- Le RPM augmente le niveau de priorité du message et appelle le *Message Disseminator* pour transmettre le message directement au SAMU.
- Le RPM informe le CARDIAN[®] et un message de traitement du message sera affiché sur le CARDIAN[®] du patient A.

Le déroulement de l'acheminement de messages utilisant notre système de contrôle autonome et intelligent, que nous venons de présenter, met en lumière l'intérêt de composition de services d'intelligence ambiante pour accélérer en temps réel la prise en charge d'un patient de façon pertinente, c'est-à-dire personnalisée et adaptée aux informations et à la connaissance disponibles au sein des écosystèmes source, cible et message. Sa description

textuelle peut paraître un peu longue, mais c'était à des fins de clarté. On peut supposer, d'après l'état des connaissances, que le temps d'exécution de ces étapes d'acheminement par les technologies actuelles sera bien inférieur au délai de réponse attendu, dans notre exemple d'un quart d'heure, délai qui était requis dès la première itération. Or, une gestion « manuelle » d'orientation du patient par lui-même, ou par l'un de ses proches (à condition qu'il ne soit pas seul à son domicile), en utilisant le courrier électronique, son téléphone et/ou son propre véhicule peut prendre une heure, et donc le mettre en danger (voire mettre la vie des autres en danger s'il tente de conduire son véhicule pour aller consulter un médecin).

9.3 Discussions et perspectives

Nos travaux de recherche se sont concrétisés par la proposition d'une méthodologie originale de conception d'architecture d'orchestration de services, pilotés par diverses méthodes d'intelligence artificielle et d'aide à la décision. Il s'agit d'une solution permettant le développement d'un système de gestion de l'acheminement de messages, entièrement autonome, personnalisable dynamiquement aux cas d'usage, et dirigé par des modèles de fusion de données provenant de divers contextes. Aucun des travaux que nous avons cités dans notre revue de la littérature ne remplit l'ensemble des fonctionnalités de notre système de gestion d'acheminement. L'avantage de notre architecture est de pouvoir, en outre, s'appliquer à de nombreux domaines afférant à l'Internet des objets connectés, où l'échange pertinent et autonome d'informations et de données provenant de dispositifs communicants en intelligence ambiante revêt un intérêt sociétal majeur. Les cas d'utilisation sont nombreux en santé (vêtements et habitat intelligents, logistique et planification de soins, gestion des urgences, hospitalisation à domicile, systèmes personnels de santé, ...), mais aussi en gestion de risques environnementaux (pollution, occurrence de phénomènes météorologiques ou géologiques anormaux, etc.), en gestion anticipée de pannes et d'incidents domestiques, en gestion de la sécurité routière, etc. En gestion de la sécurité routière, par exemple, la gestion d'acheminement peut être appliquée sur les données issues de différents types de capteurs installés dans ou près des routes et dans les véhicules pour envoyer des informations sur la circulation, sur des accidents routiers, sur la chute d'arbres, d'objets ou de pierres, etc. Pour chaque scénario, notre système peut déterminer le destinataire le plus approprié (police, SAMU, pompier, services techniques, usagers de la route, etc.) et adapter ses décisions d'acheminement en fonction du changement de contexte.

En matière d'échange de données en e-santé, le besoin d'interopérabilité et de conformité aux standards internationaux d'échanges numériques de données médicales mérite une attention particulière. Pour cela, nous avons veillé à ce que notre architecture soit compatible avec HL7, qui est le standard le plus utilisé dans le monde dans le domaine de la santé. Nous avons ainsi étendu le modèle d'information de référence d'HL7 (HL7 RIM), qui constitue le dictionnaire analytique source à partir duquel tous les messages HL7 sont construits, pour aller au-delà de la représentation sémantique et syntaxique de l'information à véhiculer. Nous avons extrait les classes du RIM qui sont afférentes à notre problématique. Nous les avons fusionnées et mises en relation de façon cohérente avec de nouvelles classes de données contextuelles provenant des écosystèmes source et cible. Le modèle d'information global

résultant de nos travaux représente ainsi une vue plus étendue du domaine, en prenant en compte l'information ambiante pertinente pour une meilleure prise en charge de la santé en environnement ubiquitaire. L'ensemble des processus de communication d'information est ainsi pilotable, comme nous l'avons montré, par des méthodes d'intelligence artificielle pour promouvoir une gestion d'acheminement intelligente, tout en restant compatible aux spécifications d'HL7 afin de préserver l'interopérabilité.

La conception des divers modèles et bases de connaissances liés aux concepts de l'acheminement des messages en e-santé nécessite l'implication de différents acteurs, experts médicaux du domaine (urgentistes, professionnels de santé, centres de télésurveillance, etc.). La modélisation du contexte médical, par exemple, de notre écosystème source auquel une politique d'acheminement est sensible, devrait être le fruit d'un consensus multi-acteurs, spécialistes des cas d'utilisation considérés, en conformité aux spécifications du vocabulaire et de la syntaxe définies par les standards internationaux en e-santé. L'architecture dirigée par des modèles que nous proposons est ainsi une architecture ouverte, instanciable dans le contexte de diverses applications, nécessitant néanmoins encore de nombreuses personnes-mois de développement.

La mise en œuvre d'un système de gestion d'acheminement des messages, intelligent et autonome, tel que celui qui est proposé dans cette thèse, nécessite de nombreuses expérimentations, tests et évaluations, à la fois sur les plans technique, médico-légal, socioéconomique, etc. avant d'être industrialisé. Ces phases de validation depuis la preuve de concepts jusqu'au passage à l'échelle dépassent très largement le cadre de ma thèse. Mais, elles sont essentielles pour les applications en e-santé, qui est un domaine extrêmement vaste et très sensible. Il est toutefois reconnu internationalement que la durée de ces étapes peut être de plus de dix ans en santé. Cet aspect est d'autant plus crucial dans notre problématique de recherche que le système proposé est doté d'une forte potentialité d'autonomie. Bien que l'autonomie soit un concept très puissant surtout dans le cadre de l'informatique ubiquitaire, sa mise en œuvre dans le domaine de l'e-santé revêt un critère complexe et soulève plusieurs problèmes : fiabilité, sécurité, etc. Tout système autonome à vocation médicale peut être à risque pour le patient. Il nécessite des évaluations multicentriques sur des bases de données importantes et parfaitement validées qui requièrent l'implication de plusieurs acteurs et beaucoup de temps pour être construites. Des problèmes de responsabilité du concepteur du système peuvent aussi être évoqués en cas de non délivrance d'un message au « bon » professionnel de santé et au « bon » moment, surtout en cas d'urgence. Néanmoins, l'existence d'un système autonome de gestion d'acheminement de messages en e-santé est un besoin majeur en e-santé. Dans le cadre de la téléassistance des personnes handicapées, par exemple, un tel système serait particulièrement utile, car il peut gérer l'acheminement de messages sans aucune intervention humaine. Toutefois, notre système, bien qu'il puisse fonctionner de manière entièrement autonome, n'exclut pas de par sa conception les interventions humaines d'interaction homme-machine. Il offre une aide pour l'assurance d'une bonne qualité d'acheminement des messages. Ainsi, dans la conception de notre système, nous avons bien sûr pris en compte les cas d'extrême urgence, où notre système intervient pour envoyer les messages directement au centre d'urgence (SAMU), mais nous

avons aussi pris en considération les préférences d'acheminement à la fois du patient et du professionnel de santé (l'interlocuteur peut avoir une destination préférée, par exemple), ainsi que les possibilités d'annulation du transfert du message, que ce soit à la demande du patient ou du professionnel de santé. Le système proposé doit ainsi être considéré comme une nouvelle solution complémentaire à l'existant, sans avoir vocation de s'y substituer, mais répondant à un besoin de confort accru dans notre société d'information et de communication.

Dans le cadre du projet EPI-MEDICS présenté dans le chapitre 2, notre système peut être implanté dans les serveurs d'alarme avec lesquels communiquent les CARDIAN[®] ou les PEM. Après détection d'une alarme moyenne par un CARDIAN[®], le message d'alarme et les données concernant le patient (sa localisation, l'ECG qui est la cause de la génération de l'alarme, l'ECG de référence, les symptômes cliniques du patient) sont envoyés via Internet au serveur d'application qui embarque notre système de gestion d'acheminement. Les composants de notre système interagissent avec une base de connaissances construite à partir d'un ensemble d'ontologies présentées chapitre 3, et que nous avons illustré chapitre 7 par quelques exemples provenant du domaine de la cardiologie.. Ces ontologies modélisent l'ensemble des connaissances de nos trois écosystèmes, source, cible et messages. Le traitement au niveau du serveur d'application permet de déterminer la destination la plus appropriée et les délais requis pour le traitement du message. Selon les préférences et la situation médicale du patient, le serveur peut envoyer un SMS, un mail ou un message spécialisé vers le PC, téléphone, PDA, etc., du professionnel de santé le plus approprié, indiquant l'URL du serveur d'application, un mot de passe et un identifiant générés aléatoirement. Le professionnel de santé peut ainsi accéder à distance aux données du patient hébergées par le serveur d'alarme et prendre ses décisions en conséquence.

Notre système peut aussi être déployé selon une architecture distribuée sur des serveurs d'application si le message est expédié à l'issue d'une intervention humaine via des interfaces homme/machine, comme dans le scénario de téléconsultation du projet Télé-infirmière présenté dans le chapitre 2. Nous rappelons que dans ce projet, l'infirmière peut transmettre les paramètres biomédicaux du patient vers un médecin sur un serveur par l'intermédiaire d'un chariot médical. Pour cela, nous pouvons envisager une architecture distribuée trois-tiers où la couche « client » représente la partie interface qui permet à l'infirmière d'interagir avec les médecins, la couche « application » embarque notre système de gestion d'acheminement de messages et la dernière couche correspond aux connaissances du contexte qui sont modélisées sous forme d'ontologies. Notre système peut lui-même être distribué sur plusieurs serveurs d'application connectés selon un protocole donné, tels que SOAP (*Simple Object Access Protocol*) [186] utilisé dans le cadre des architectures orientées services, etc.

Notre problématique s'inscrivant dans le domaine de la santé et concernant plus particulièrement l'échange de données médicales, la prise en compte des aspects de sécurité, d'éthique, de protection de la vie privée et juridiques s'imposent. La plupart des services composant notre architecture recueillent et traitent une grande quantité de données à caractère personnel appartenant à la fois au patient, à savoir son dossier médical (antécédents cliniques, facteurs de risque, etc.) et au professionnel de santé (localisation géographique, niveau

d'expérience, etc.). Cette gestion de données personnelles doit être normalement réalisée après consentement exprès de la personne concernée. Tout un cadre légal strict est mis en place en France, à cet effet, avec la loi Informatique et Libertés [187] qui décrit les droits d'accès, d'opposition, de modification, de suppression, etc. Un aspect important à ne pas oublier dans la phase de développement logiciel est la nécessité, surtout dans le domaine de la santé, de veiller à stocker la traçabilité d'acheminement de tous les messages pour des questions juridiques. Cette fonctionnalité de mémoire de la gestion d'acheminement est d'ailleurs déjà intégrée dans notre architecture afin de doter notre système de caractéristiques d'auto-adaptation en fonction de l'historique d'acheminement, fonctionnalités pouvant notamment être réalisées au moyen d'algorithmes utilisant l'approche des colonies de fourmis, par exemple.

Plusieurs perspectives de mise en œuvre découlent de notre travail de thèse, que nous discutons ci-après. La première concerne le classement des professionnels de santé, du plus pertinent au moins pertinent eu égard à la situation contextuelle du patient. Nous avons cité dans notre mémoire la méthode des plus proches voisins qui permet d'appréhender de façon multidimensionnelle une analyse de tendance vers la « meilleure » décision. A cet effet, les méthodes d'aide à la décision multicritères, telles que les méthodes PROMETHEE et GAIA [188], peuvent être des pistes prometteuses. Ainsi, la méthode PROMETHEE permet le classement basé sur la pondération des critères de sélection des candidats. La méthode GAIA apporte un complément descriptif à PROMETHEE. Elle pourrait être utilisée pour mieux comprendre le choix décisionnel des destinataires par notre système, grâce à une représentation graphique permettant l'analyse des données d'un problème multicritères et la visualisation de l'impact d'un jeu de poids sur le choix du destinataire.

L'accès au contenu du message et aux données médicales dans notre système doit être fait de manière très sécurisée afin de garantir la confidentialité, respecter le principe déontologique du secret professionnel et le droit au respect de la vie privée. Pour cela, des contrôles d'accès au contenu du message devront être intégrés dans notre système en fonction des rôles et des responsabilités des interlocuteurs.

Les prises de décisions d'acheminement des messages doivent non seulement être exactes mais aussi exécutées dans un délai minimum surtout en cas d'urgence où la vie du patient est mise en jeu. Notre système est a priori pénalisé par le raisonnement sur un grand nombre de données contextuelles pour prendre des décisions d'acheminement, ce qui peut influencer son temps d'exécution. Pour cela, nous suggérons, par exemple, l'utilisation des techniques de parallélisme qui permettent de traiter un grand nombre de données de manière simultanée en un temps minimal. Proposé par Google, le modèle de programmation MapReduce [189], par exemple, permet les calculs parallèles, l'utilisation et le traitement d'un nombre important de données (Big Data) en les distribuant sur des clusters de nœuds. MapReduce consiste en deux fonctions Map () et Reduce (). Dans l'étape Map, le nœud parent découpe un problème en sous problèmes et les distribue sur d'autres nœuds. Dans l'étape Reduce, après le traitement des sous problèmes, les nœuds remontent leurs résultats au nœud parent qui les avait

sollicités. A la fin, le problème devient traité d'une façon parallèle et distribuée et dans un temps raisonnable.

Pour automatiser le service du processus décisionnel d'acheminement, une piste possible peut être de s'appuyer sur les technologies d'intelligence artificielle, plus spécifiquement sur les algorithmes d'apprentissage par renforcement tels que *TD-learning* ou *Q-learning*. Le processus décisionnel deviendra ainsi capable d'apprendre à partir d'expériences et de chercher un comportement décisionnel optimal qui maximise la somme des récompenses au cours du temps. Une récompense dans notre problématique peut être attribuée au service « *Routing and escalation decision* » de notre architecture, si sa décision de routage a permis de traiter un message par le destinataire dans le temps requis, et donc une prise en charge efficace du patient dans le cadre de l'e-santé.

9.4 Conclusion

Dans ce chapitre, nous avons illustré et démontré les fonctionnalités et l'applicabilité de notre système en e-santé pour le suivi de patients à risque cardiovasculaire, qui est l'un des domaines phares où le temps de prise en charge des patients par des équipes médicales spécialisées après l'apparition des premiers symptômes, est crucial pour leur survie et leur qualité de vie. Nous avons montré, au moyen d'un diagramme de collaborations BPMN, le déroulement du processus de l'acheminement d'un message d'alarme moyenne d'un scénario réel issu du projet Européen EPI-MEDICS. Cette instanciation nous a permis de montrer les interactions entre les différents modules « participants » de notre architecture et les capacités d'auto-adaptation de notre système en cas de déclenchement d'exceptions (refus de prise en charge ou dépassement de délai requis pour le traitement du message) et/ou de changement de contexte (aggravation des signes cliniques).

Nous avons ensuite discuté différents aspects afférents à notre problématique qui ont porté sur l'applicabilité de notre solution à divers scénarios en e-santé, mais aussi dans d'autres domaines, son interopérabilité et sa conformité aux standards internationaux d'échange de données en santé, sa prise en compte de l'analyse de risques en intégrant la possibilité d'interaction homme-machine en cas d'artefacts d'origine technique ou humaine, et l'historique des processus d'acheminement. Puis, nous avons évoqué les aspects sécurité, éthique, protection de la vie privée et juridiques soulevés par notre problématique de recherche. Enfin, nous avons proposé des solutions de déploiement des services de notre architecture, et présenté quelques perspectives de mise en œuvre de notre système.

Conclusion Générale

L'avènement des technologies de réseaux sans fils et des dispositifs médicaux communicants a bouleversé les techniques d'échange d'informations médicales en e-santé. L'un des défis dans le domaine de l'e-santé est à présent d'assurer, surtout en cas d'urgence, une qualité de service intelligente, optimale et pertinente de gestion des messages, dans le but de promouvoir le bien-être des patients n'importe où et n'importe quand.

Tel est le défi lancé par notre sujet de thèse. Nos travaux ont permis de proposer une méthode originale d'orchestration de services intelligents visant à optimiser la qualité de services de transmission des messages, applicable dans divers domaines dont celui de la santé. Notre objectif a été de rendre le système d'acheminement des messages capable de superviser son environnement, de détecter tout changement de contexte (indisponibilité de destinataires ou de ressources matérielles, changement de localisation géographique, etc.) ou les exceptions événementielles imprévues qui peuvent survenir en temps réel (dépassement du délai requis pour le traitement du message, etc.) et, en conséquence, de s'adapter d'une manière « intelligente ».

Bilan sur les approches

L'étude que nous avons menée a abouti dans un premier temps à identifier les défis qui découlent de notre problématique (Chapitre 2). Ces défis sont :

- la diversité et la multiplicité des scénarios d'usage en e-santé : téléassistance à domicile, téléconsultation, télé-expertise, etc. Un système de gestion d'acheminement de messages doit fournir une solution générique qui s'applique aux différents cas d'usage et aux différents types de message (informatif, alarme, technique, etc.) ;
- la diversité des acteurs pouvant être impliqués dans un acte médical en e-santé : patients, infirmiers, médecins, hébergeurs de données médicales, ambulanciers, aides-soignants, etc. Le défi est de pouvoir transmettre le message vers l'acteur le plus approprié en fonction de la situation contextuelle (médicale, géographique, environnementale, etc.) de l'expéditeur, qui peut être un patient, un professionnel de santé ou non.

La nature ubiquitaire qui caractérise l'environnement de notre système soulève plusieurs contraintes que notre solution doit prendre en compte, liées notamment à la mobilité des utilisateurs, à l'hétérogénéité de l'information et des ressources, à l'évolution du contexte, etc. Pour illustrer ces contraintes, nous avons choisi de présenter deux scénarios issus de deux projets réalisés dans notre laboratoire (Chapitre 2). Ceci nous a permis par la suite de déduire les besoins requis en gestion d'acheminement des messages : **personnalisation, sensibilité au contexte, auto-adaptation, interopérabilité, supervision, autonomie, sécurité et protection de la vie privée, intelligence**, et d'une manière générale la **qualité de service** qui englobe les points cités précédemment (Chapitre 2). De notre point de vue, un système de pilotage du convoiement des messages doit pouvoir transmettre les messages d'une manière intelligente vers le destinataire le plus approprié (personnalisation), superviser son environnement et s'adapter aux changements du contexte (supervision, sensibilité au

contexte) avec une moindre intervention humaine (autonomie), tout en respectant les standards internationaux d'échanges de données médicales (interopérabilité).

En vue de ces besoins, nous avons orienté l'état de l'art de notre thèse vers l'exploration de différentes thématiques afférentes à notre problématique : l'e-santé et la télémédecine citoyenne (Chapitre 1), la gestion de l'acheminement des messages en e-santé (Chapitre 2), les systèmes autonomes et sensibles au contexte et les méthodes d'aide à la décision. Dans la littérature, plusieurs travaux de recherche traitant la problématique d'acheminement de messages sous différents points de vue ont été réalisés en e-santé, mais aussi dans d'autres domaines. Néanmoins, aucune des solutions proposées ne couvre l'intégralité des besoins que nous avons identifiés dans le Chapitre 2.

Pour traiter notre problématique, nous avons tout d'abord adopté une approche *bottom-up* (Chapitre 2) afin de déduire les spécifications d'acheminement en e-santé à partir de scénarios concrets, et ensuite une approche *top-down* qui va d'une vision générique (Partie III) applicable à tout domaine vers une vision plus spécifique, adaptée au domaine de l'e-santé (Partie IV).

La première approche envisagée est de proposer des solutions sur la conception d'un système de gestion d'acheminement des messages sensible au contexte, qui s'applique à différents scénarios d'usage et dans différents domaines d'application (Chapitre 5). Pour cela, nous avons proposé :

- un modèle générique qui met en relation les objets liés au paradigme « sensibilité au contexte » et ceux de la gestion d'acheminement dans tout domaine. Ce modèle reflète l'aspect multidimensionnel du contexte auquel une stratégie d'acheminement est sensible. Nous avons ainsi introduit le terme « écosystème » pour désigner un ensemble de systèmes d'informations complexes qui sont reliés à une entité (objet/personne) au sein d'un environnement donné. Nous avons distingué trois types d'écosystèmes : écosystèmes source, cible et message ;
- une architecture orientée services et dirigée par des modèles qui répond aux besoins que nous avons préalablement déduits. Elle a pour objectif de promouvoir une gestion autonome, personnalisable, intelligente et auto-adaptative d'acheminement des messages. Notre architecture vise à être la plus générique possible pour répondre aux différents scénarios d'application. Elle comporte les composants logiciels suivants :
 - le *Message Reporter* constitue la source du message, responsable de la publication du message ;
 - le *Routing Policy Manager* forme le noyau de l'architecture. Il est doté d'un processus décisionnel intelligent autonome et auto-adaptatif pour la prise de décision de routage (r/envoyer, rediriger, annuler), pour l'inférence des besoins requis pour l'acheminement des messages (moyens d'acheminement,

ressources matérielles, spécialité/expérience) et pour le choix de la « bonne » destination auquel le message doit être envoyé ;

- le *Message Disseminator* est responsable du transfert du message au destinataire final ;
- le *Message Status Supervisor* permet le suivi du cycle de vie des messages et la détection des dépassements de délai de réception, de lecture et de réponse aux requêtes ;
- l'*Ecosystem Supervisor* supervise les données contextuelles source, cible et message, et détecte les changements pour adapter le processus décisionnel d'acheminement, si nécessaire ;
- les écosystèmes source, cible et message constituent des bases de connaissances qui regroupent un ensemble de données contextuelles concernant l'origine, la destination et le contenu du message.

Ces composants interagissent en exécutant un ensemble de processus métiers complexes. Pour mieux comprendre le fonctionnement de notre architecture, nous avons proposé, dans le Chapitre 6, une modélisation des processus métiers BPM décrivant d'une manière détaillée et structurée les activités de chaque composant qu'elle comporte.

Notre deuxième approche est appliquée au domaine de l'e-santé (Partie IV). Dans un souci de préserver l'interopérabilité de notre solution, nous avons proposé un synoptique applicatif de notre architecture pour le pilotage de l'acheminement des messages, compatible avec HL7 version 3 (Chapitre 7). Nos travaux de recherche contribuent ainsi à proposer une extension du standard international le plus usité dans le monde de la santé, en intégrant le paradigme de l'intelligence ambiante. La complexité des écosystèmes composant le synoptique nous a mené à proposer trois types de modélisation de ces écosystèmes, présentés Chapitre 7 : UML, en couche et ontologique. L'ensemble de modélisations complémentaires nous permet d'avoir une vue détaillée et structurée du contenu de nos écosystèmes.

Nous avons présenté par la suite les principaux flux d'information échangés entre les composants de notre architecture appliquée à l'e-santé. Pour illustrer l'échange de ces flux, nous avons proposé dans le Chapitre 7, un exemple de diagrammes de séquence dans le cadre d'un scénario d'acheminement de messages pour la surveillance des personnes âgées.

L'un des atouts majeurs de notre approche consiste à embarquer de l'intelligence artificielle au niveau du processus décisionnel d'acheminement pour le pilotage automatique des messages. Pour ce faire, nous avons mené, dans le Chapitre 8, une étude comparative sur plusieurs méthodes d'intelligence artificielle, notamment les systèmes experts, les réseaux bayésiens, les arbres de décision et les réseaux bayésiens dynamiques, méthodes que nous avons présenté dans l'état de l'art (Chapitre 4), etc. Au vu des résultats de cette comparaison, nous avons choisi d'utiliser les réseaux bayésiens dynamiques pour inférer les paramètres

d'acheminement, principalement à cause de leur capacité à faire face aux différents contraintes soulevées par le problème d'acheminement (temps et inférence, incertitude et incomplétude des données contextuelles, raisonnement flexible et non déterministe dans la définition de la politique d'acheminement, dépendance entre données contextuelles, etc.). Pour implémenter le service « Supervision du statut du message » de notre architecture, nous avons proposé d'utiliser des réseaux de Petri temporels présentés Chapitre 4. Les réseaux de Petri nous ont permis de modéliser la dynamique du statut des messages et incluent dans leurs propres descriptions des éléments d'analyse quant aux problèmes d'impasses des messages.

Enfin, nous avons montré dans le Chapitre 9 l'applicabilité de notre système en e-santé, plus particulièrement dans le domaine de la cardiologie. Nous avons décrit, au moyen d'un diagramme de collaboration BPMN le déroulement de l'acheminement d'un message d'alarme moyenne d'un scénario réel depuis sa génération jusqu'à son traitement. Cette instanciation nous a permis d'avoir une vue détaillée des fonctionnalités de notre système, de son comportement lors de l'occurrence des exceptions (aggravation des signes cliniques, refus de la requête de prise en charge, dépassement du délai requis pour le traitement du message), ainsi que sur les interactions entre les différents composants de notre architecture. Enfin, nous avons proposé des solutions sur le déploiement des services de notre architecture et discuté les contraintes sur les plans médico-légaux, sécurité de la vie privée, etc., soulevées par notre problématique.

Perspectives

La gestion d'acheminement des messages, comme nous l'avons vu, dépend fortement des données contextuelles modélisées dans les écosystèmes source, cible et message, données qui sont souvent spécifiques à chaque scénario. En effet, contrairement par exemple à celui d'une alarme moyenne, dans un scénario d'alarme mineure, notre système n'a pas besoin de collecter la donnée contextuelle « localisation » des destinataires possibles puisqu'il y a une faible probabilité de transfert du patient. Un axe de recherche très important à explorer serait de trouver des techniques automatisées de collecte de ces données pertinentes appropriées à chaque cas d'usage.

Pour garantir un processus décisionnel intelligent de routage, nous avons exploré quelques méthodes d'intelligence artificielle. Il reste cependant plusieurs autres méthodes d'aide à la décision à évaluer en termes de qualité de service et de temps de réponse. Le raisonnement à base de cas pourrait, par exemple, être utilisé pour implémenter le service *Decide routing and escalation*. Une piste de recherche prometteuse pourrait être aussi d'invoquer dynamiquement des méthodes selon les types des messages et les scénarios dans chaque domaine d'application.

L'un des challenges de la recherche en e-santé est son caractère pluridisciplinaire invoquant de fait un processus itératif de création d'innovations méthodologiques, de réalisation technologique et d'évaluation à grande échelle, depuis l'identification des besoins jusqu'à son industrialisation. Notre problématique de recherche nous a conduit à adopter une démarche de type *bottom-up* puis *top-down*, qui devrait être poursuivie par une nouvelle

phase de type *bottom-up*, et ainsi de suite, pour que nos travaux puissent enfin résulter en une solution améliorant la qualité de vie des patients/citoyens. Tel est notre souhait.

Bibliographie

- [1] FAYN, J. et RUBEL, P. Toward a Personal Health Society in Cardiology. *IEEE Transactions on Information Technology in Biomedicine*, 2010, vol. 14, n° 2, p. 401–409.
- [2] TÉLISSON, D. Méthodologie pour la conception de systèmes d'information pervasifs : application à la pSanté, 118 p. Thèse : Informatique : Lyon : 2006.
- [3] Business Process Model and Notation™ (BPMN™). [en ligne]. Disponible à l'adresse : <http://www.omg.org/spec/BPMN/>.
- [4] Health Level Seven. [en ligne]. Disponible à l'adresse : www.hl7.org.
- [5] World Health Organisation. [en ligne]. Disponible à l'adresse : <http://www.who.int/topics/ehealth/en/>.
- [6] LASBORDES, P. La télésanté : un nouvel atout au service de notre bien-être - Un plan quinquennal éco-responsable pour le déploiement de la télésanté en France [en ligne]. rapport public. 2009. Disponible à l'adresse : <http://www.ladocumentationfrancaise.fr/var/storage/rapports-publics/094000539.pdf>.
- [7] RUBEL, P., FAYN, J., NOLLO, G., et al. Toward personal eHealth in cardiology. Results from the EPI-MEDICS telemedicine project. *Journal of electrocardiology*, 2005, vol. 38, n° 4, p. 100–106.
- [8] Site d'information du Dossier Médical Personnel - DMP. [en ligne]. Disponible à l'adresse : <http://www.dmp.gouv.fr/>.
- [9] MINISTÈRE DE LA SANTÉ ET DES SPORTS. *Décrets, arrêtés, circulaires Décret no 2010-1229 du 19 octobre 2010 relatif à la télémédecine* [en ligne]. 2010. Disponible à l'adresse : http://esante.gouv.fr/sites/default/files/joe_20101021_0013.pdf.
- [10] MOEN, A., HACKL, W.O., HOFDIJK, J., et al. eHealth in Europe - Status and Challenges. *Yearbook of Medical Informatics*, 2013, vol. 8, p. 59-63.
- [11] ASIP SANTÉ. *Rapport d'activité 2014* [en ligne]. 2014. Disponible à l'adresse : <http://esante.gouv.fr/actus/politique-publique/l-asip-sante-publie-son-rapport-d-activite-2014>.
- [12] TOUBERT, K.A. et VANDERSCHULDEN, M. *La démographie médicale à l'horizon 2030 : de nouvelles projections nationales (...)* [en ligne]. 2009. Disponible à l'adresse : <http://www.drees.sante.gouv.fr/la-demographie-medicale-a-l-horizon-2030-de-nouvelles-projections-nationales-et-regionales-detaillees,5488.html>.
- [13] Union Européenne. [en ligne]. Disponible à l'adresse : ec.europa.eu/information_society/activities/health/whatis_ehealth/index_en.htm.
- [14] FAYN, J., GHEDIRA, C., TELISSON, D., et al. Towards new integrated information and communication infrastructures in e-health. Examples from cardiology. In *IEEE Computers in Cardiology*. 2003, vol. 30, p. 113-116.
- [15] RUBEL, P., FAYN, J., SIMON-CHAUTEMPS, L., et al. New paradigms in telemedicine: ambient intelligence, wearable, pervasive and personalized. *Studies in health technology and informatics*, 2004, vol. 108, p. 123-132.
- [16] DZENOWAGIS, J. et KERNEN, G. Connecting for health : global vision, local insight : report for the World Summit on the Information Society : country profiles [en ligne]. 2006.

- Disponible à l'adresse :
http://www.who.int/ehealth/publications/WSISReport_Connecting_for_Health.pdf.
- [17] AL-SHORBAJI, N. et GEISSBUHLER, A. Establishing an evidence base for e-health: the proof is in the pudding. *Bulletin of the World Health Organization*, 2012, vol. 90, n° 5, p. 322-322A.
- [18] ASIP SANTÉ. *Etude sur la télésanté et la Télémedecine en Europe* [en ligne]. 2011. Disponible à l'adresse :
http://www.fieec.fr/iso_album/etude_telesante_fieec_asip_sante_28_mars.pdf.
- [19] BAGAYOKO, C.O., TRAORÉ, D., THEVOZ, L., et al. Medical and economic benefits of telehealth in low- and middle-income countries: results of a study in four district hospitals in Mali. *BMC health services research*, 2014, vol. 14 Suppl 1, p. S9.
- [20] Infarctus du myocarde. [en ligne]. Disponible à l'adresse :
<http://www.inserm.fr/thematiques/physiopathologie-metabolisme-nutrition/dossiers-d-information/infarctus-du-myocarde>.
- [21] DE LUCA, G., SURYAPRANATA, H., OTTERVANGER, J.P., et al. Time delay to treatment and mortality in primary angioplasty for acute myocardial infarction: every minute of delay counts. *Circulation*, 2004, vol. 109, n° 10, p. 1223-1225.
- [22] MCGINN, A.P., ROSAMOND, W.D., GOFF, D.D.C., et al. Trends in prehospital delay time and use of emergency medical services for acute myocardial infarction: experience in 4 US communities from 1987-2000. *American Heart Journal*, 2005, vol. 150, n° 3, p. 392-400.
- [23] MAGLAVERAS, N., BONATO, P. et TAMURA, T. Guest editorial. Special section on personal health systems. *IEEE transactions on information technology in biomedicine*, 2010, vol. 14, n° 2, p. 360-363.
- [24] LAIRES, M.F., LADEIRA, M.J. et CHRISTENSEN, J.P. (éd.). *Health in the New Communications Age*. IOS Press, Amsterdam, The Netherlands, 1995, 733 p. ISBN 978-90-5199-224-3.
- [25] TÉLISSON, D., ATOUI, H., J. FAYN, et al. Ambient intelligence and pervasive architecture designed within the EPI-MEDICS personal ECG monitor. *IJHISI*, 2008, vol. 3, p. 68-80.
- [26] GdR STIC-Santé. [en ligne]. Disponible à l'adresse : <http://www.stic-sante.org/>.
- [27] FAYN, J. et VUILLERME, N. Theme G: eHealth. Results and future works. *IRBM*, 2013, vol. 34, n° 1, p. 18-20.
- [28] Décret n°2006-6 du 4 janvier 2006 relatif à l'hébergement de données de santé à caractère personnel et modifiant le code de la santé publique (dispositions réglementaires). [en ligne]. 2006. Disponible à l'adresse :
<https://www.legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000000264665#LEGIARTI000006248643>.
- [29] GHEDIRA, C., MARET, P., FAYN, J., et al. Adaptive customizable browsing through semistructured cardiology hypermedia data. In *Computers in Cardiology*. Cambridge, MA, 2000, vol. 27, p. 295-298.

- [30] GHEDIRA, C., MARET, P., FAYN, J., et al. Adaptive user interface customization through browsing knowledge capitalization. *International Journal of Medical Informatics*, 2002, vol. 68, n° 1-3, p. 219-228.
- [31] Intelligence artificielle. *Wikipédia* [en ligne]. 2016. Disponible à l'adresse : https://fr.wikipedia.org/wiki/Intelligence_artificielle.
- [32] BUCHANAN, B.G. et SHORTLIFFE, E.H. *Rule Based Expert Systems: The Mycin Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley Longman Publishing, Boston, MA, USA, 1984, 784 p. ISBN 978-0-201-10172-0.
- [33] YU, V.L., FAGAN, L.M., WRAITH, S.M., et al. Antimicrobial selection by a computer. A blinded evaluation by infectious diseases experts. *JAMA*, 1979, vol. 242, n° 12, p. 1279-1282.
- [34] RUBEL, P., FAYN, J., NOLLO, G., et al. Toward personal eHealth in cardiology. Results from the EPI-MEDICS telemedicine project. *Journal of electrocardiology*, 2005, p. 100–106.
- [35] ISO/DIS 11073-91064.2007 et EN 1064:2005. Medical Informatics-Standard Communication Protocol-Computer-assisted Electrocardiography (SCP-ECG). [en ligne]. Disponible à l'adresse : <http://www.centc251.org/>.
- [36] FAYN, J. A classification tree approach for cardiac ischemia detection using spatiotemporal information from three standard ECG leads. *IEEE transactions on bio-medical engineering*, 2011, vol. 58, n° 1, p. 95-102.
- [37] ATOUI, H., FAYN, J. et RUBEL, P. A Novel Neural-Network Model for Deriving Standard 12-Lead ECGs From Serial Three-Lead ECGs: Application to Self-Care. *IEEE Transactions on Information Technology in Biomedicine*, 2010, vol. 14, n° 3, p. 883-890.
- [38] ATOUI, H. Conception de systèmes intelligents pour la télémédecine citoyenne, 157 p. Thèse : Informatique : Lyon : 2006.
- [39] ARNAUD, P., RUBEL, P., MORLET, D., et al. Methodology of ECG interpretation in the Lyon program. *Methods of Information in Medicine*, 1990, vol. 29, n° 4, p. 393-402.
- [40] WILLEMS, J.L., ABREU-LIMA, C., ARNAUD, P., et al. The diagnostic performance of computer programs for the interpretation of electrocardiograms. *The New England Journal of Medicine*, 1991, vol. 325, n° 25, p. 1767-1773.
- [41] ATOUI, H., FAYN, J., GUEYFFIER, F., et al. Cardiovascular risk stratification in decision support systems: A probabilistic approach. application to pHealth. In *IEEE Computers in Cardiology*. Valencia, 2006, vol. 33, p. 281-284.
- [42] RUBEL, P., FAYN, J., AROD, S., et al. Intelligence ambiante et systèmes pervasifs en télémédecine. Exemples d'application en télécardiologies. *Médecine d'urgence*, 2004, vol. 6, p. 521–526.
- [43] Loi n° 2002-303 du 4 mars 2002 relative aux droits des malades et à la qualité du système de santé. [en ligne]. Disponible à l'adresse : <https://www.legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000000227015>.
- [44] Digital Imaging and Communications in Medicine. [en ligne]. Disponible à l'adresse : <http://dicom.nema.org>.

- [45] CHABRIAIS, J. *Les standards pour les systèmes d'Information de Santé : (Série documents d'initiation)* [en ligne]. 2001. Disponible à l'adresse : <http://www.yakaliker.org/@api/deki/files/374/=HL7-1ini1.pdf>.
- [46] ASIP SANTÉ. *Cadre d'Interopérabilité des Systèmes d'Information de Santé (CI-SIS)* [en ligne]. 2009. Disponible à l'adresse : http://esante.gouv.fr/sites/default/files/CI-SIS_DOC-CHAPEAU_V1.3.1_0.pdf.
- [47] Reference Information Model (RIM). [en ligne]. Disponible à l'adresse : <http://www.hl7.org/implement/standards/rim.cfm>.
- [48] ABDULMALIK, S. Health Level Seven Reference Information Model. [en ligne]. 2014. Disponible à l'adresse : <http://fr.slideshare.net/AShakil/hl7-reference-information-model>.
- [49] RAMOS, C., AUGUSTO, J.C. et SHAPIRO, D. Ambient Intelligence-the Next Step for Artificial Intelligence. *IEEE Intelligent Systems*, 2008, vol. 23, n° 2, p. 15-18.
- [50] WEISER, M. The Computer for the 21st Century. *ACM SIGMOBILE Mob. Comput. Commun. Rev.*, 1999, vol. 3, n° 3, p. 3-11.
- [51] WEISER, M. Whatever Happened to the Next-generation Internet? *Commun. ACM*, 2001, vol. 44, n° 9, p. 61-69.
- [52] VASILAKOS, A.V. et LISETTI, C. Guest editorial. Special section on affective and pervasive computing for healthcare. *IEEE transactions on information technology in biomedicine*, 2010, vol. 14, n° 2, p. 183-185.
- [53] DEY, A.K., SALBER, D., FUTAKAWA, M., et al. *An Architecture to Support Context-Aware Applications* [en ligne]. GVU Technical Report;GIT-GVU-99-23. 1999. Disponible à l'adresse : <http://hdl.handle.net/1853/3390>.
- [54] PASCOE, J. Adding generic contextual capabilities to wearable computers. In *IEEE Second International Symposium on Wearable Computers*. Pittsburgh, PA, USA, 1998, p. 92-99.
- [55] RAMADOUR, P., FAKHRI, M. et CAUVET, C. A goal-driven method for business services engineering. In *2010 IEEE 4th International Conference on New Trends in Information Science and Service Science (NISS)*. Gyeongju, 2010, p. 554-558.
- [56] SCHILIT, B., ADAMS, N. et WANT, R. Context-Aware Computing Applications. In *First Workshop on Mobile Computing Systems and Applications (WMCSA'94)*. Santa Cruz, California, USA, 1994, p. 85-90.
- [57] BROWN, P.J., BOVEY, J.D. et CHEN, X. Context-aware applications: from the laboratory to the marketplace. *IEEE Personal Communications*, 1997, vol. 4, n° 5, p. 58-64.
- [58] RYAN, N., PASCOE, J. et MORSE, D. Enhanced Reality Fieldwork: the Context-aware Archaeological Assistant. In GAFFNEY, V., VAN LEUSEN, M. et EXXON, S. (éd.), *Computer Applications in Archaeology*. Oxford, 1998, p. 182-196.
- [59] BOUHADDOU, O., WARNEKAR, P., PARRISH, F., et al. Exchange of Computable Patient Data between the Department of Veterans Affairs (VA) and the Department of Defense (DoD): Terminology Mediation Strategy. *Journal of the American Medical Informatics Association : JAMIA*, 2008, vol. 15, n° 2, p. 174-183.

- [60] GAYNOR, M., MYUNG, D., GUPTA, A., et al. Interoperability of Medical Applications and Devices. In *IEEE Proceedings of the Annual Hawaii International Conference on System Sciences*. Waikoloa, HI, 2008, p. 240.
- [61] HORN, P. *Autonomic Computing: IBM's Perspective on the State of Information Technology* [en ligne]. Technical report. 2001. Disponible à l'adresse : http://www.research.ibm.com/autonomic/manifesto/autonomic_computing.pdf.
- [62] PHAROW, P. et BLOBEL, B. Specific interoperability problems of security infrastructure services. *Studies in Health Technology and Informatics*, 2006, vol. 121, p. 349-363.
- [63] KATSIKAS, S. et KOKOLAKIS, S. High level security policies for Healthcare Information Systems. *Studies in Health Technology and Informatics*, 2003, vol. 96, p. 98-104.
- [64] UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS. X.902 : Technologies de l'information – Traitement réparti ouvert – Modèle de référence: fondements. [en ligne]. 2009. Disponible à l'adresse : <https://www.itu.int/rec/T-REC-X.902-200910-I/fr>.
- [65] MOO-MENA, F.J. Modélisation des architectures logicielles dynamiques : application à la gestion de la qualité de service des applications à base de services web, 219 p. Thèse : Informatique : Toulouse : 2007.
- [66] FRANKEN, L.J.N. et HAVERKORT, B.R.H.M. Quality of service management using generic modelling and monitoring techniques. *Distributed Systems Engineering*, 1997, vol. 4, n° 1, p. 28.
- [67] VOGEL, A., KERHERVE, B., VON BOCHMANN, G., et al. Distributed multimedia and QoS: a survey. *IEEE MultiMedia*, 1995, vol. 2, n° 2, p. 10-19.
- [68] SOUIL, M. et BOUABDALLAH, A. On QoS Provisioning in Context-Aware Wireless Sensor Networks for Healthcare. In *2011 IEEE Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN)*. Maui, HI, 2011, p. 1-6.
- [69] SALEHIE, M. et TAHVILDARI, L. Self-adaptive Software: Landscape and Research Challenges. *ACM Trans. Auton. Adapt. Syst.*, 2009, vol. 4, n° 2, p. 14:1–14:42.
- [70] SUN, Y., TAN, W., LI, L., et al. SLA detective control model for workflow composition of cloud services. In *2013 IEEE 17th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*. 2013, p. 165–171.
- [71] DING, R. et MUNTEAN, G.-M. A context-aware cross-layer energy-efficient adaptive routing algorithm for WLAN communications. In *2012 IEEE 37th Conference on Local Computer Networks (LCN)*. 2012, p. 176-179.
- [72] PENG, S., YANG, S.X., GREGORI, S., et al. An adaptive QoS and energy-aware routing algorithm for wireless sensor networks. In *International Conference on Information and Automation, 2008. ICIA 2008*. 2008, p. 578–583.
- [73] NAGIB, G. et WAHIED, G.A. Network Routing Protocol using Genetic Algorithms. *International Journal of Electrical & Computer Sciences*, vol. 10, n° 2, p. 40.
- [74] KE, Z., LI, L. et CHEN, N. A Crossover Game Routing Algorithm for Wireless Multimedia Sensor Networks. In *IEEE 9th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD '08)*. Phuket, 2008, p. 595-599.

- [75] EHSAN, S. et HAMDAROU, B. A Survey on Energy-Efficient Routing Techniques with QoS Assurances for Wireless Multimedia Sensor Networks. *IEEE Communications Surveys Tutorials*, 2012, vol. 14, n° 2, p. 265–278.
- [76] CHIU, D.K.W., LI, Q. et KARLAPALEM, K. Web interface-driven cooperative exception handling in ADOME workflow management system. *Information Systems*, 2001, vol. 26, n° 2, p. 93-120.
- [77] CHIU, D.K.W., KWOK, B.W.C., WONG, R.L.S., et al. Alerts for healthcare process and data integration. In *IEEE Proceedings of the 37th Annual Hawaii International Conference on System Sciences*. 2004, p. 10.
- [78] DILMAGHANI, R.B. et RAO, R.R. A Systematic Approach to Improve Communication for Emergency Response. In *42nd Hawaii International Conference on System Sciences, 2009. HICSS '09*. 2009, p. 1–8.
- [79] DILMAGHANI, R.B. et RAO, R.R. Supervisory decision making in emergency response application. In *2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*. 2013, p. 553–558.
- [80] HSIEH, F.-S. Context-aware Workflow Driven Resource Allocation for e-Healthcare. In *2007 9th International Conference on e-Health Networking, Application and Services*. 2007, p. 34–39.
- [81] MIRAOU, M., TADJ, C., FATTAHI, J., et al. Dynamic Context-Aware and Limited Resources-Aware Service Adaptation for Pervasive Computing. *Advances in Software Engineering*, 2011, vol. 2011, n° 7, p. 11.
- [82] RODRÍGUEZ, M.D., FAVELA, J., MARTÍNEZ, E.A., et al. Location-aware access to hospital information and services. *IEEE transactions on information technology in biomedicine: a publication of the IEEE Engineering in Medicine and Biology Society*, 2004, vol. 8, n° 4, p. 448-455.
- [83] MITCHELL, S., SPITERI, M.D., BATES, J., et al. Context-aware Multimedia Computing in the Intelligent Hospital. In *Proceedings of the 9th Workshop on ACM SIGOPS European Workshop: Beyond the PC: New Challenges for the Operating System* [en ligne]. 2000, p. 13–18.
- [84] STANFORD, V. Beam me up, doctor McCoy. *IEEE Pervasive Computing*, 2003, vol. 2, n° 3, p. 13-18.
- [85] NAGEBA, E., RUBEL, P. et FAYN, J. Towards an intelligent exploitation of heterogeneous and distributed resources in cooperative environments of eHealth. *IRBM*, 2013, vol. 34, n° 1, p. 79-85.
- [86] NAGEBA, E., FAYN, J. et RUBEL, P. A model driven ontology-based architecture for supporting the quality of services in pervasive telemedicine applications. In *3rd International Conference on Pervasive Computing Technologies for Healthcare, 2009. PervasiveHealth 2009*. 2009, p. 1 –8.
- [87] CARDOSO DE MORAES, J.L., LOPES DE SOUZA, W., FERREIRA PIRES, L., et al. A novel architecture for message exchange in Pervasive Healthcare based on the use of Intelligent Agents. In *2013 ACS International Conference on Computer Systems and Applications (AICCSA)*. 2013, p. 1-8.

- [88] What is openEHR? [en ligne]. Disponible à l'adresse : <http://www.openehr.org/>.
- [89] REENSKAUG, T. *The original MVC reports* [en ligne]. 1979. Disponible à l'adresse : http://folk.uio.no/trygver/2007/MVC_Originals.pdf.
- [90] CUGGIA, M., BAYAT, S., POULAIN, P., et al. Interopérabilité sémantique: comparaison de la représentation du score d'APGAR en HL7V3 et Open EHR. In *Risques, Technologies de l'Information pour les Pratiques Médicales*. Springer, Paris. 2009, p. 125-136.
- [91] ASIP SANTÉ. *Interfaces d'accès au Système de Messageries sécurisées de Santé (MSSanté)* [en ligne]. Rapport des spécifications Fonctionnelles et Techniques. 2014. Disponible à l'adresse : <https://www.mssante.fr>.
- [92] IHE PCD TECHNICAL COMMITTEE. *IHE Patient Care Device (PCD) Technical Framework* [en ligne]. Rapport technique. 2015. Disponible à l'adresse : http://www.ihe.net/uploadedFiles/Documents/PCD/IHE_PCD_TF_Vol1.pdf.
- [93] IHE PCD TECHNICAL COMMITTEE. *IHE Patient Care Device Technical Framework - Alarm Communication Management (ACM)-Supplement Trial Implementation* [en ligne]. Rapport technique. 2012. Disponible à l'adresse : http://www.ihe.net/uploadedFiles/Documents/PCD/IHE_PCD_Suppl_ACM_Rev1.3_TI_2012-08-16.pdf.
- [94] CARDOZO, N., GONZALEZ, S., MENS, K., et al. Modeling and Analyzing Self-Adaptive Systems with Context Petri Nets. In *2013 International Symposium on Theoretical Aspects of Software Engineering (TASE)*. 2013, p. 191-198.
- [95] IBM CORPORATION. *An architectural blueprint for autonomic computing*. [en ligne]. [sans date]. Disponible à l'adresse : [http://www-03.ibm.com/autonomic/pdfs/AC Blueprint White Paper V7.pdf](http://www-03.ibm.com/autonomic/pdfs/AC_Blueprint_White_Paper_V7.pdf).
- [96] GANEK, A.G. et CORBI, T.A. The dawning of the autonomic computing era. *IBM Systems Journal*, 2003, vol. 42, n° 1, p. 5-18.
- [97] HU, P., INDULSKA, J. et ROBINSON, R. An Autonomic Context Management System for Pervasive Computing. In *Sixth Annual IEEE International Conference on Pervasive Computing and Communications, 2008. PerCom 2008*. 2008, p. 213-223.
- [98] NAMI, M.R. et BERTELS, K. A Survey of Autonomic Computing Systems. In *Third International Conference on Autonomic and Autonomous Systems, 2007. ICAS07*. Athens, 2007, p. 26-26.
- [99] TIANFIELD, H. Multi-agent autonomic architecture and its application in e-medicine. In *IEEE/WIC International Conference on Intelligent Agent Technology, 2003. IAT 2003*. 2003, p. 601-604.
- [100] STERRITT, R., PARASHAR, M., TIANFIELD, H., et al. A Concise Introduction to Autonomic Computing. *Adv. Eng. Inform.*, 2005, vol. 19, n° 3, p. 181-187.
- [101] AHUJA, K. et DANGEY, H. Autonomic Computing: An emerging perspective and issues. In *IEEE International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)*. Ghaziabad, 2014, p. 471-475.
- [102] VASSEV, E. et HINCHEY, M. The Challenge of Developing Autonomic Systems. *Computer*, 2010, vol. 43, n° 12, p. 93-96.

- [103] TESAURO, G., CHESS, D.M., WALSH, W.E., et al. A Multi-Agent Systems Approach to Autonomic Computing. In *IEEE Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*. Washington, DC, USA, 2004, p. 464–471.
- [104] FERBER, J. *Les systèmes multi-agents. Vers une intelligence collective*. InterEditions, Paris, 1997, 522 p. ISBN 978-2-7296-0665-7.
- [105] LABROU, Y. et FININ, T. A Semantics Approach for KQML-a General Purpose Communication Language for Software Agents. In *Proceedings of the Third International Conference on Information and Knowledge Management (CIKM'94)*. New York, 1994, p. 447–455.
- [106] WEISS, G. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. New Ed, Cambridge, Mass., 2000, 643 p. ISBN 978-0-262-73131-7.
- [107] BRATMAN, M.E. *Intention, Plans, and Practical Reason*. Harvard University Press, Stanford, Calif, 1999, 212 p. ISBN 978-1-57586-192-0.
- [108] FERBER, J. *Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*. 1e éd., Boston, MA, USA, 1999, 528 p.
- [109] MÉZIANE TANI, I. *Stratégie de rendez-vous dans les systèmes multi-agents*, Mémoire : Informatique et télécommunication : Tlemcen, Algérie : 2007.
- [110] PAVÉ, A. *Modélisation en biologie et en écologie*. Aléas, Lyon, 2002, 560 p. ISBN 2-908016-32-X.
- [111] CAMBIER, C., BOUSQUET, F. et DANSOKO, D. Un univers multi-agents pour la modélisation du système de la pêche du Delta Central du Niger. In TCHUENTE, M. (éd.), *Proceedings of the 1st African conference on research in computer science*. Rocquencourt, 1993, p. 111-122.
- [112] IFFENECKER, C. *Un système multi-agents pour le support des activités de conception de produits*, 145 p. Thèse : Sciences appliquées : Paris, France : 1992.
- [113] SHAKSHUKI, E. et REID, M. Multi-Agent System Applications in Healthcare: Current Technology and Future Roadmap. *Procedia Computer Science*, 2015, vol. 52, p. 252-261.
- [114] SALEEM, R.M., MUHAMMAD, A. et MARTINEZ-ENRIQUEZ, A.M. Remote Patient Monitoring and Healthcare Management Using Multi-agent Based Architecture. In *2010 IEEE Ninth Mexican International Conference on Artificial Intelligence (MICAI)*. Pachuca, 2010, p. 118-123.
- [115] SHEN, Y., COLLOC, J., JACQUET-ANDRIEU, A., et al. Emerging medical informatics with case-based reasoning for aiding clinical decision in multi-agent system. *Journal of Biomedical Informatics*, 2015, vol. 56, p. 307-317.
- [116] BROWN, P. The Stick-e Document: a Framework for Creating Context-aware Applications. BROWN, A., BRÜGGEMENN-KLEIN, A. et FENG, A. (éd.), *EPODD - Electronic Publishing, Origination, Dissemination and Design*, 1996, vol. 8, n° 2&3, p. 259-272.
- [117] SASSE, A. et JOHNSON, C. Plasticity of User Interfaces: Framework and Research Agenda. In *INTERACT'99*. 1999, p. 110-117.

- [118] DEY, A.K., ABOARD, G.D. et SALBER, D. A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-aware Applications. *Hum.Comput. Interact.*, 2001, vol. 16, n° 2, p. 97–166.
- [119] ASTHANA, A., CRAVATTS, M. et KRZYZANOWSKI, P. An indoor wireless system for personalized shopping assistance. In *IEEE Workshop on Mobile Computing Systems and Applications*. Santa Cruz, California, USA, 1994, p. 69-74.
- [120] SCHMIDT, A., AIDOO, K., TAKALUOMA, A., et al. Advanced Interaction in Context. In *Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing*. London, UK, 1999, p. 89–101.
- [121] BELHANAFI-BEHOULI, N. Ajout de mécanismes de réactivité au contexte dans les intergiciels pour composants dans le cadre d'utilisateurs nomades, 170 p. Thèse : Informatique : Evry : 2006.
- [122] FLEUREY, F., DEHLEN, V., BENCOMO, N., et al. Modeling and Validating Dynamic Adaptation. In CHAUDRON, M.R.V. (éd.), *Models in Software Engineering* [en ligne]. 2008, p. 97-108.
- [123] BEHOULI, N.B., TACONET, C. et BERNARD, G. An architecture for supporting Development and Execution of Context-Aware Component applications. In *2006 ACS/IEEE International Conference on Pervasive Services*. 2006, p. 57-66.
- [124] TACONET, C., KAZI-AOUL, Z., ZAIER, M., et al. CA3M: A Runtime Model and a Middleware for Dynamic Context Management. In MEERSMAN, R., DILLON, T. et HERRERO, P. (éd.), *On the Move to Meaningful Internet Systems: OTM 2009* [en ligne]. 2009, p. 513-530.
- [125] KEPHART, J.O. et CHESS, D.M. The vision of autonomic computing. *Computer*, 2003, vol. 36, n° 1, p. 41-50.
- [126] Object Management Group. [en ligne]. Disponible à l'adresse : <http://www.omg.org/>.
- [127] SEIDWITZ, E. What Models Mean. *IEEE Softw.*, 2003, vol. 20, n° 5, p. 26–32.
- [128] BEZIVIN, J. et GERBE, O. Towards a precise definition of the OMG/MDA framework. In *IEEE 16th Annual International Conference on Automated Software Engineering, 2001. (ASE 2001)*. 2001, p. 273-280.
- [129] SHENG, Q.Z. et BENATALLAH, B. ContextUML: a UML-based modeling language for model-driven development of context-aware Web services. In *IEEE International Conference on Mobile Business, 2005. ICMB 2005*. 2005, p. 206-212.
- [130] KRAUSE, M. et HOCHSTATTER, I. Challenges in Modelling and Using Quality of Context (Qoc). In *Proceedings of the Second International Conference on Mobility Aware Technologies and Applications*. Berlin, Heidelberg, 2005, p. 324–333.
- [131] BUCHHOLZ, T. et SCHIFFERS, M. Quality of Context: What It Is And Why We Need It. In *Proceedings of the 10th Workshop of the OpenView University Association (OVUA'03)*. Boston, Massachusetts, USA, 2003, p. 1-14.
- [132] DARGIE, W. et HAMANN, T. A Distributed Architecture for Reasoning about a Higher-Level Context. In *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, 2006. (WiMob'2006)*. Montreal, Que, 2006, p. 268-275.

- [133] HAGHIGHI, P.D., KRISHNASWAMY, S., ZASLAVSKY, A., et al. Reasoning about Context in Uncertain Pervasive Computing Environments. In ROGGEN, D., LOMBRISER, C., TRÖSTER, G., et al. (éd.), *Smart Sensing and Context*. Springer, Berlin Heidelberg. 2008, p. 112-125.
- [134] FERRY, N. Adaptations dynamiques au contexte en informatique ambiante : propriétés logiques et temporelles, 201 p. Thèse : Informatique : Nice : 2011.
- [135] QUINLAN, J.R. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Francisco, CA, USA, 1993, 302 p.
- [136] BREIMAN, L., FRIEDMAN, J., STONE, C.J., et al. *Classification and Regression Trees*. Monterey, CA, 1984.
- [137] KASS, G.V. An Exploratory Technique for Investigating Large Quantities of Categorical Data. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 1980, vol. 29, n° 2, p. 119-127.
- [138] CAPPONI, C. Arbres de décision. [en ligne]. Université Aix-Marseille. Disponible à l'adresse : <http://pageperso.lif.univ-mrs.fr/~cecile.capponi/lib/exe/fetch.php?media=cours-arbres.pdf>
- [139] RAKOTOMALALA, R. Arbres de Decision. *Revue MODULAD*, 2005, N° 33, p. 163-187
- [140] DENIS, F., KADRI, H. et CAPPONI, C. Introduction à l'apprentissage automatique. [en ligne]. Université d'Aix-Marseille. 2014. Disponible à l'adresse : <http://pageperso.lif.univ-mrs.fr/~francois.denis/IAAM1/chap2.pdf>
- [141] DENNING, P.J. Towards a Science of Expert Systems. *IEEE Expert*, 1986, vol. 1, n° 2, p. 80-83.
- [142] ALPHONSE, C. *Knowledge Management et Web 2.0 : Outils, méthodes et applications*. Hermes Science Publications, 2013, 250 p. ISBN 978-2-7462-9549-0.
- [143] JACKSON, P. *Introduction To Expert Systems*. 3e éd., Harlow, England, 1998, 560 p.
- [144] GIARRATANO, J.C. et RILEY, G.D. *Expert Systems: Principles and Programming*. 4e éd., 2004, 288 p. ISBN 978-0-534-38447-0.
- [145] DENIS, F. et MICLET, L. *Intelligence artificielle: Les systèmes experts* [en ligne]. Lannion, 2006. Disponible à l'adresse : <http://people.irisa.fr/Laurent.Miclet/documents/documentsenseignement/SE.pdf>.
- [146] ADAM, E. systèmes experts. [en ligne]. Cours Master 1/ IUP 3 Informatique. Valence. Disponible à l'adresse : <http://emmanuel.adam.free.fr/site/IMG/pdf/SEP.pdf>.
- [147] DRÉO, J., PÉTROWSKI, A., SIARRY, P., et al. *Métaheuristiques pour l'optimisation difficile*. 2003, 368 p. ISBN 2-212-11368-4.
- [148] OSMAN, I.H. et LAPORTE, G. Metaheuristics: A bibliography. *Annals of Operations Research*, 1996, vol. 63, n° 5, p. 511-623.
- [149] DENEUBOURG, J.L., ARON, S., GOSS, S., et al. The self-organizing exploratory pattern of the argentine ant. *Journal of Insect Behavior*, 1990, vol. 3, n° 2, p. 159-168.

- [150] DORIGO, M. et STÜTZLE, T. *Ant Colony Optimization*. Bradford Company, Scituate, MA, USA, 2004, ISBN 978-0-262-04219-2.
- [151] DORIGO, M., MANIEZZO, V. et COLORNI, A. Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 1996, vol. 26, n° 1, p. 29-41.
- [152] DORIGO, M. et GAMBARDILLA, L.M. Ant colony system: a cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, 1997, vol. 1, n° 1, p. 53-66.
- [153] BONABEAU, E., DORIGO, M. et THERAULAZ, G. *Swarm Intelligence: From Natural to Artificial Systems* [en ligne]. New York, NY, USA, 1999, 320 p. ISBN 978-0-19-513159-8.
- [154] Vista Project: Nasa Mission Control. [en ligne]. Disponible à l'adresse : <http://robotics.stanford.edu/~koller/BNtut/sld151.htm>.
- [155] NAÏM, P., WUILLEMIN, P.H., LERAY, P., et al. *Réseaux bayésiens*. 3e éd., Paris, 2007, 423 p. ISBN 978-2-212-11972-5.
- [156] PEARL, J. *Causality: Models, Reasoning, and Inference*. 1e éd., Cambridge, U.K. ; New York, 2000, 400 p. ISBN 978-0-521-77362-1.
- [157] COOPER, G.F. The computational complexity of probabilistic inference using bayesian belief networks. *Artificial Intelligence*, 1990, vol. 42, n° 2, p. 393-405.
- [158] LAURITZEN, S.L. et SPIEGELHALTER, D.J. Local computations with probabilities on graphical structures and their application to expert systems. In SHAFER, G. et PEARL, J. (éd.), *Readings in Uncertain Reasoning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. 1990, p. 415-448.
- [159] JENSEN, F.V. et LAURITZEN, S. Bayesian updating in recursive graphical models by local computations. *Computational Statistics Quarterly*, 1990, vol. 4, p. 269-282.
- [160] MURPHY, K.P. *Dynamic Bayesian Networks: Representation, Inference and Learning*, 225 p. Thèse : Informatique : Berkeley : 2002.
- [161] RABINER, L.R. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 1989, vol. 77, n° 2, p. 257-286.
- [162] FOATA, D. et FUCHS, A. *Processus stochastiques : Processus de poisson, chaînes de Markov et martingales*. 2e éd., Paris, 2004, 236 p. ISBN 978-2-10-048850-6.
- [163] BENAÏM, M. et KAROUI, N. Promenade aléatoire: Chaînes de Markov et simulations: martingales et stratégie. 2004, 316 p. ISBN 2-7302-1168-3.
- [164] DEQUIER, J. *Chaînes de Markov et applications*. Mémoire de probatoire en informatique. Grenoble, 2005.
- [165] BAUM, L.E. An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of Markov Processes. *Inequalities III*, 1972, vol. 3, p. 1-8.
- [166] BAUM, L.E., PETRIE, T., SOULES, G., et al. A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains. *The Annals of Mathematical Statistics*, 1970, vol. 41, n° 1, p. 164-171.

- [167] BELLOT, D. Fusion de données avec des réseaux bayésiens pour la modélisation des systèmes dynamiques et son application en télémédecine, 176 p. Thèse : Informatique : Nancy I : 2002.
- [168] C. ROSE. Modélisation stochastique pour le raisonnement médical et ses applications à la télémédecine, 201 p. Thèse : Informatique : Nancy I : 2011.
- [169] CHOQUET-GENIET, A. *Les réseaux de Petri : Un outils de modélisation Cours et exercices corrigés*. Dunod, Paris, 2006, 240 p. ISBN 978-2-10-049147-6.
- [170] MURATA, T. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 1989, vol. 77, n° 4, p. 541-580.
- [171] SILVA, M. et TERUEL, E. Petri Nets for the Design and Operation of Manufacturing Systems. *European Journal of Control*, 1997, vol. 3, n° 3, p. 182-199.
- [172] VALETTE, R. *Les réseaux de Petri* [en ligne]. Toulouse, 2002. Disponible à l'adresse : <http://homepages.laas.fr/francois/POLYS/pnrv.pdf>.
- [173] RAMCHANDANI, C. *Analysis of asynchronous concurrent systems by timed petri nets*. Rapport technique. Cambridge, MA, USA, 1974.
- [174] MERLIN, P.M. *A Study of the Recoverability of Computing Systems.*, Thèse : Informatique : Irvine : 1974.
- [175] SIFAKIS, J. Le contrôle des systèmes asynchrones : concepts, propriétés, analyse statique, Thèse : Informatique : Grenoble I : 1979.
- [176] SIFAKIS, J. Performance evaluation of systems using nets. In BRAUER, W. (éd.), *Net Theory and Applications*. Springer, Berlin Heidelberg. 1980, p. 307-319.
- [177] KHANSA, W. Réseaux de Pétri P-Temporels: contribution à l'étude des systèmes à événements discrets, 190 p. Thèse : Informatique : Annecy : 1997.
- [178] GUIZANI, N. et FAYN, J. Model-Driven Architecture for Self-Adaptive Context-Aware Message Routing in Pervasive Environments. In *The 11 th International Conference on Autonomic and Autonomous Systems (ICAS 2015)*. Rome, Italy, 2015, p. 44-50.
- [179] GUIZANI, N. et FAYN, J. Framework for Intelligent Message Routing Policy Adaptation. In *12th IEEE International Conference on Autonomic Computing (ICAC)*. Grenoble, 2015, p. 235-238.
- [180] GUTIN, G., YEO, A. et ZVEROVICH, A. Traveling salesman should not be greedy: domination analysis of greedy-type heuristics for the TSP. *Discrete Applied Mathematics*, 2002, vol. 117, n° 1-3, p. 81-86.
- [181] GUIZANI, N. et FAYN, J. Personalized Message Routing Policy in Pervasive eHealth Applications Based on HL7 RIM. In *Studies in Health Technology and Informatics*. 2015, vol. 211, p. 265-270.
- [182] Acute Cardiovascular Care Association. [en ligne]. Disponible à l'adresse : www.escardio.org/ACCA.
- [183] SWRL: A Semantic Web Rule Language Combining OWL and RuleML. [en ligne]. Disponible à l'adresse : <https://www.w3.org/Submission/SWRL/>.

- [184] HL7 Message examples: version 2 and version 3. [en ligne]. Disponible à l'adresse : <http://www.ringholm.com>.
- [185] GUIZANI, N. et FAYN, J. Cardiology eHealth messages routing policies management driven by Dynamic Bayesian Networks. In *IEEE Computing in Cardiology Conference (CinC)*. Nice, 2015, vol. 42, p. 185-188.
- [186] World Wide Web Consortium (W3C). [en ligne]. Disponible à l'adresse : <https://www.w3.org>.
- [187] Loi « Informatique et Libertés ». [en ligne]. Disponible à l'adresse : <https://www.cnil.fr/fr/loi-78-17-du-6-janvier-1978-modifiee>.
- [188] Multicriteria Decision Aid Methods, Modeling and Software. [en ligne]. Disponible à l'adresse : <http://www.promethee-gaia.net/>.
- [189] MapReduce. [en ligne]. Disponible à l'adresse : <https://hadoop.apache.org>.