



HAL
open science

Apports de la Notion de Contexte à Différents Systèmes

Manuele Kirsch Pinheiro

► **To cite this version:**

Manuele Kirsch Pinheiro. Apports de la Notion de Contexte à Différents Systèmes. Informatique ubiquitaire. Université Paris 1 Panthéon Sorbonne, 2021. tel-03122073

HAL Id: tel-03122073

<https://theses.hal.science/tel-03122073v1>

Submitted on 26 Jan 2021

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.

Habilitation à Diriger des Recherches

Spécialité : Informatique

Présentée par

Manuele KIRSCH PINHEIRO

Maître de Conférences – Section CNU 27

Apports de la Notion de Contexte à Différents Systèmes

Habilitation soutenue publiquement le 13 Janvier 2021 devant le jury composé de :

Philippe Lalanda, Professeur, Université de Grenoble, Rapporteur
Philippe Roose, Maître de Conférences HDR, Université de Pau, Rapporteur
Florence Sedes, Professeur, Université de Toulouse, Rapporteur
Yolande Berbers, Professeur, Katholieke Universiteit Leuven, Président du jury
Agnès Front, Professeur, Université de Grenoble, Examinatrice
Chantal Taconet, Maître de Conférences HDR, Telecom Sud Paris, Examinatrice
Massimo Villari, Professeur, Université de Messine, Examineur
Bénédicte Le Grand, Professeur, Université Paris 1 Panthéon Sorbonne
Carine Souveyet, Professeur, Université Paris 1 Panthéon Sorbonne

*La différence entre un rêve et la réalité peut être,
tout simplement,
notre envie de le rendre réel...*

*“Disciplina é liberdade
Compaixão é fortaleza
Ter bondade é ter coragem...”
Há Tempos, Legião Urbana*

*à ma mère, qui m'a appris à croire ...
à mon mari, qui m'a appris à aimer ...
... au pouvoir des rêves ...*

Remerciements

Ce document résume 12 ans d'une carrière qui s'est construite au gré des rencontres, des opportunités et des échanges, sans lesquels rien de tout ça n'aurait été possible. La liste de remerciements est donc forcément longue, car j'ai toujours eu la chance d'avoir pu côtoyer des personnes qui m'ont inspirée et motivée. Ce document ne serait pas possible sans toutes ces personnes.

Je tiens tout d'abord à remercier chaleureusement les membres du jury, qui ont tous promptement accepté de participer avec enthousiasme. Il était important pour moi de pouvoir vous présenter ce document et d'avoir votre avis sur toutes les questions et les propositions qu'il contient.

Mes premiers remerciements vont donc à mes rapporteurs, Mme Florence Sedès, M. Philippe Roose et M. Philippe Lalanda. Je tiens ainsi remercier Mme Florence Sedès pour avoir accepté de rapporter ce document, mais également par l'exemple qu'elle représente à toutes les jeunes chercheuses de notre communauté. Je tiens aussi à remercier M. Philippe Roose non seulement pour avoir accepté d'être rapporteur de ce document, mais également pour m'avoir ouvert la voie avec ses recherches impliquant les communautés *middleware* et Système d'Information. Je tiens aussi à remercier Philippe par sa bonne humeur et pour m'avoir fait découvrir Biarritz, où des nombreuses étapes de ce document ont été rédigées. Du même, je remercie chaleureusement M. Philippe Lalanda d'avoir accepté de réaliser le rapport, mais aussi de vos travaux dans la communauté de l'Informatique Pervasive, qui ont toujours m'inspiré dans cette communauté.

Au-delà des rapporteurs, je tiens également à remercier les autres membres du jury, à commencer par Mme Yolande Berbers. Yolande m'a accueilli à la K.U.Leuven pendant mon année de post-doc. Cette expérience m'a profondément marqué et a été certainement décisive pour la suite de ma carrière. Sans cette incroyable opportunité, je n'aurais jamais pu occuper le poste que j'occupe aujourd'hui. Merci, Yolande, de m'avoir tant appris.

I would like also to thanks M. Massimo Villari for accepting to participate to this jury, but also for the fruitful collaborations we have started in these last years. All our discussions have enriched this document, as well as my future research. I hope this collaboration will last long. *Grazie*, Massimo!

Je tiens aussi à remercier Mme Agnès Front d'avoir accepté de participer à ce jury. J'ai eu le privilège de côtoyer Agnès depuis mes années de thèse à Grenoble et d'avoir pu enrichir ma compréhension des Systèmes d'Information grâce à ses travaux. Je tiens également à la remercier de sa profonde gentillesse et de m'avoir tant inspiré.

Tous mes remerciements aussi à Mme Chantal Taconet, dont les travaux sur la QoC représentent pour moi une véritable source d'inspiration. Je la remercie aussi pour m'avoir soufflé, au détour d'une conversation, l'argument qu'il me fallait pour pouvoir me lancer dans cette aventure. Merci.

Ces remerciements ne seraient pas complets sans mes profonds remerciements à mes deux acolytes, sans qui ce document n'aurait jamais vu le jour et sans qui le travail à Paris ne serait sûrement pas aussi fun : Mme Bénédicte Le Grand et Mme Carine Souveyet. Bénédicte m'a toujours apporté son soutien, ce pourquoi je la remercie énormément. Je la remercie aussi pour son amitié, sa patience, ainsi que pour sa gentillesse et le fait d'être cette personne si solaire. Pouvoir partager les amphes avec toi, avec qui je partage une certaine idée de l'enseignement, c'est un privilège incalculable. Pouvoir compter sur toi parmi mes amis, aussi.

Je tiens aussi à remercier Carine de m'avoir accompagné depuis le début de cette aventure. Nous avons pu discuter chaque détail de ce document. Sans tes remarques et tes questionnements, ce document n'aurait certainement pas vu le jour. Je la remercie également pour nous si nombreuses échanges et collaborations. Il s'agit d'autant de moments de plaisir et de partage, que j'espère pourront encore continuer longtemps. Des nombreuses perspectives de recherche nous attendent après ce document, et il nous reste encore un certain temps avant la retraite.

J'ai aussi une pensée toute particulière à mes encadrants de thèse et de master, Hervé, Jérôme, Marlène et Valdeni. Je leur remercie chaleureusement, car c'est grâce à eux et à leurs enseignements que j'ai pu me lancer dans le chemin de la recherche.

Impossible pour moi de ne pas remercier également tous mes collègues du CRI, qui doivent supporter mon caractère depuis tout ce temps. Je les remercie de m'avoir accueillie (coucou Rebecca ! coucou Raul !) et de m'avoir donné l'opportunité d'accueillir, à mon tour, d'autres chercheurs (coucou Irina !). Je les remercie tous pour tous ces moments de partage, aussi bien autour du labo que de la MIAGE.

Au-delà de mes collègues au sein du CRI, je tiens également à remercier mon équipe de TD avec qui je partage ma passion pour l'enseignement : Emille, Meledje, Amina, Déborah, Bader et tant d'autres. Je les remercie d'être là pour nos étudiants, de nous faire confiance, à Bénédicte et à moi-même. Je n'aurais pas pu rédiger ce document sans avoir une équipe si géniale pour m'aider dans mes cours.

En parlant d'enseignement, impossible de ne pas remercier également mes étudiants. Tout d'abord, je tiens à exprimer ici toute ma gratitude à mes étudiants en thèse (Salma, David, Ali et Assia), sans qui ce document serait certainement beaucoup moins épais. Je voulais aussi remercier les autres doctorants et ATERs du laboratoire (ceux qui sont encore là et ceux qui sont déjà partis), Floriane, Nourhène, Sana, Kahina et tant d'autres, pour tous ces moments d'échange et de partage. Je tiens également à remercier à tous mes étudiants de licence et de master, qui sont mon moteur, ma source de motivation, des véritables rayons de soleil qui illuminent mes journées. Merci à mes « petites pestes adorées » d'avoir permis de partager ce bout de chemin avec vous, tant en cours qu'à travers vos mémoires. J'ai beaucoup appris grâce à vous (ou à cause de vous). Je ne sais pas si j'ai réussi à vous apprendre quelque chose, mais si j'ai pu au moins vous montrer qu'on peut être passionnée par son métier, ça aurait déjà valu la peine.

Je dois aussi remercier à tous mes amis. Ils ne sont pas nombreux, mais ils comptent beaucoup pour moi. Je les remercie tous, ceux qui sont très loin (promis Audrey, cette année je viens au Canada), à ceux qui sont un peu moins loin (Céline, Marius et les enfants Romain et Olivia que j'adore tant), à ma deuxième famille (Léo, Marina et les princesses Sophie et Alice). Je suis probablement la pire amie qu'on puisse imaginer tant je communique peu, mais vous êtes toujours là dans mes pensées.

Tous mes remerciements aussi à ma famille, sans qui absolument rien n'aurait été possible. Je remercie mes sœurs (Damaris et Esther), mon frère Ramiro et sa famille (Mari et les filles Luiza, Bianca et Carolina), à ma mère Sandra. Vous êtes ma plus grande source d'inspiration. Même si on est loin, vous êtes toujours là pour moi. Merci surtout à ma mère. Si je suis ici, si j'ai pu devenir quelqu'un, c'est grâce à toi, à ton exemple, à tes valeurs qui sont aujourd'hui les miennes. Merci. J'espère pouvoir te rendre toujours fière.

Enfin, un grand merci à mon mari Angelo, mon Angelo. Tu as su m'aider, tu m'as portée sur tes épaules, tu m'as supportée dans tous les sens du terme. Ça n'a pas été toujours simple, mais on est là, et on sera toujours là... Merci.

Table de Matières

I	Introduction	13
II	État de l'art	17
1	Avant-propos	17
2	Exemples d'applications	18
3	Public cible	21
4	Roadmap pour l'ingénierie de contexte	23
4.1	Dimension Finalité.....	26
4.2	Dimension Sujet	28
4.3	Dimension Modèle	31
4.4	Dimension Acquisition.....	34
4.5	Dimension Interprétation	37
4.6	Dimension distribution	39
5	Synthèse des questions	40
6	Considérations finales	42
III	Contributions	45
1	Contexte dans les collecticiels	45
1.1	Rappel de la problématique	45
1.2	Contributions & Impact	46
1.3	Bibliométrie	50
2	Contexte dans les environnements pervasifs	53
2.1	<i>Context grouping</i>	53
2.1.1	Rappel de la problématique	53
2.1.2	Contributions & Impact	54
2.1.3	Bibliométrie	57
2.2	Projet PER-MARE.....	59
2.2.1	Rappel de la problématique	59
2.2.2	Contributions	61
2.2.3	Originalité & Impact	72
2.2.4	Bibliométrie	75
2.3	Bilan.....	77
3	Contexte dans l'orientation service	79
3.1	Fouille de contexte (« <i>Context Mining</i> »)	79
3.1.1	Rappel de la problématique	79
3.1.2	Contribution & Impact	80
3.1.3	Bibliométrie	85
3.2	Sélection de services par comparaison de graphes	87

3.2.1	Rappel de la problématique	87
3.2.2	Contribution & Impact	88
3.2.3	Bibliométrie	90
3.3	Sélection de service par l'intention et le contexte.....	92
3.3.1	Rappel de la problématique	92
3.3.2	Contribution & Impact	92
3.3.3	Bibliométrie	97
3.4	Prédiction de services.....	99
3.4.1	Rappel de la problématique	99
3.4.2	Contribution & Impact	100
3.4.3	Bibliométrie	103
3.5	Bilan.....	104
4	Contexte dans les Systèmes d'Information Pervasifs.....	107
4.1	Espace de services.....	112
4.1.1	Rappel de la problématique	112
4.1.2	Contribution & Impact	112
4.1.3	Bibliométrie	118
4.2	Gestion opportuniste de ressources dans les SIP	119
4.2.1	Rappel de la problématique	119
4.2.2	Contribution & Impact	120
4.2.3	Bibliométrie	131
4.3	Bilan.....	132
IV	Conclusion & Perspectives.....	133
V	Références	137
	Annexes.....	153
	Annexe I - <i>Curriculum Vitae</i>	155
	Annexe II – Liste de Travaux et Publications	169

Liste des Figures

Figure 1. Positionnement des contributions présentées dans une vision schématique d'un Système d'Information Pervasif.....	14
Figure 2. Cartographie des contributions et de leur évolution.	15
Figure 3. (a) Une installation GridStix [223] et (b) un prototype de système d'observation de stress hydrique [130].	19
Figure 4. Observation simultanée d'un même tracé par deux terminaux distincts.	21
Figure 5. Kit distribué aux étudiants comportant un RaspberryPi ZeroW, un capteur de température I2C BMP 280 et un triple LED [130].....	22
Figure 6. Dimensions de la roadmap pour l'ingénierie de contexte [132].	24
Figure 7. Cycle de vie pour la gestion de contexte proposé par [195].....	24
Figure 8. La Qualité de Contexte vue comme une dimension transversale, affectant chaque dimension de la roadmap [133].	26
Figure 9. Meta-modèle pour l'information de contexte [136].	29
Figure 10. Ontologie (a) et méta-modèle orienté objets utilisés pour la modélisation de l'information de contexte [262].	33
Figure 11. Influence et flot de données entre les dimensions modèle, acquisition et interprétation [130].	39
Figure 12. Premier niveau du modèle de contexte proposé dans [125].	47
Figure 13. Détails du modèle de contexte proposé dans [125].	47
Figure 14. Définition de la mesure de similarité <i>SimO</i> d'après [125].	48
Figure 15. Architecture conceptuelle pour un traitement commun de la notion de contexte et de conscience de groupe [128].	50
Figure 16. Évolution des citations dans le temps.	52
Figure 17. Requête CQL définissant un groupe basé sur la localisation de son utilisateur [134].	55
Figure 18. Architecture mise en place de pour le mécanisme de distribution de contexte proposé [134].	55
Figure 19. Illustration de différents nœuds appartenant à deux groupes distincts [134].	56
Figure 20. Graphique illustrant l'évolution des citations au cours des années.	58
Figure 21. Illustration du principe MapReduce sur un exemple d'annuaire, où on comptabilise le nombre de téléphones fixes (« 123... ») par adresse postal (« XYZ Str... »).	60
Figure 22. Ensemble d'outils pour le Big Data basés sur le framework Hadoop (illustration d'après [12]).	61
Figure 23. Overview de l'architecture Hadoop (d'après [45])	62
Figure 24. Scénarios d'exécution d'un benchmark Big Data (application TestDFSIO) dans un cluster homogène dédié (cas A) et dans un environnement pervasif, sans prise en compte du contexte (cas B) et avec prise en compte du contexte d'exécution (cas C et D) [46].	63
Figure 25. ContextCollector responsable par l'observation du contexte d'exécution d'un nœud [45].	64
Figure 26. Représentation de l'entrée d'un nouveau nœud sur une communauté CloudFit.	66
Figure 27. Comparaison entre les temps d'exécution du benchmark WorldCount avec la plateforme Hadoop et la bibliothèque de calcul réparti CONFIIT [235].	68
Figure 28. Comparaison des temps d'exécution entre Hadoop et CONFIIT modifié avec l'usage de la bibliothèque P2P FreePastry [234].	68
Figure 29. Comparaison entre l'exécution d'une même application dans un cluster dédié et dans une grille pervasive [233].	70
Figure 30. Architecture logicielle de la plateforme CloudFIT.	71
Figure 31. Exécution d'une application sur une grille pervasive, avec un nœud qui quitte la communauté et un autre qui l'intègre à posteriori [236].	71
Figure 32. Exécution de l'application WordCount sur un environnement hétérogène [236].	72
Figure 33. Gartner Hype Cycle de technologies émergentes pour l'année 2012.	73

Figure 34. Gartner Hype Cycle de technologies émergentes pour l'année 2019.	74
Figure 35. Graphique illustrant la distribution des citations dans le temps.....	77
Figure 36. Application de l'ACF pour l'analyse du contexte d'usage d'applications mobiles [117].	81
Figure 37. Treillis associant l'usage de certaines applications à leur contexte d'usage [116].....	82
Figure 38. Évolution des citations liées à la fouille de contexte.....	87
Figure 39. Comparaison entre deux graphes de contexte à travers les mesures de similarité locales et globale [256].	88
Figure 40. Citations par publication.....	91
Figure 41. Illustration de la relation entre un service, une intention et un contexte [175].....	93
Figure 42. Description d'une intention (a) et d'un contexte requis (b) associés à un service [172].	94
Figure 43. Extrait de l'ontologie d'intention [170].....	94
Figure 44. Méta-modèle de contexte proposé dans [170].	95
Figure 45. Ontologie multi-niveaux de contexte issu de [170].	95
Figure 46. Vision globale du mécanisme de sélection de service par intention et contexte [181].	96
Figure 47. Évolution des citations dans le temps par référence.....	99
Figure 48. Différentes étapes du processus de prédiction de service proposé [178].....	101
Figure 49. Citations par publication.....	104
Figure 50. Illustration d'un espace de service (a) et des multiples espaces (b) composant un SIP (d'après [131, 175]).	114
Figure 51. Axes de développement autour d'un SIP [175].	115
Figure 52. (a) Le micro-business containers en tant qu'élément de liaison entre le niveau technique et le niveau conceptuel, et (b) la composition d'un micro-business container et sa relation avec les microéléments [232].	116
Figure 53. Correspondance entre l'espace de services, les micro-business containers et le niveau technique.....	117
Figure 54. Résultats sur les temps de lecture (a) et d'écriture (b) sur un RaspberryPi 3 [238].	122
Figure 55. Résultats obtenus en transférant des données distantes vers un équipement de proximité.	122
Figure 56. Exécution d'une simulation Monte Carlo sur la plateforme CloudFIT avec surcharge de ressources et sans observation du contexte pour le placement des tâches.....	123
Figure 57. Exécution d'une simulation Monte Carlo dans la plateforme CloudFIT avec surcharge des machines m3 et m4, et contexte d'exécution pris en considération pour le placement des tâches.	124
Figure 58. Exécution simulant un partitionnement du réseau, avec la machine m2 perdant tout contact avec les autres.	125
Figure 59. Dimensions de la gestion de ressources identifiées à partir de l'étude.	126
Figure 60. Représentation simplifiée de l'architecture conceptuelle pour la gestion de ressources d'un SIP.	130
Figure 61. Context facility au service d'un SIP.	135

Liste des Tableaux

Tableau 1. Les dimensions de gestion de contexte avec les principales questions et quelques exemples [130].	41
Tableau 2. Analyse bibliométrique de publications choisies.....	51
Tableau 3. Nombre de citations observées pour les publications relatives à la thématique distribution de contexte.....	58
Tableau 4. Indications sur les dispositifs utilisés lors des expérimentations.	70
Tableau 5. Nombre de citations par article, organisées par année de citation.....	76
Tableau 6. Analyse bibliométrique des publications liées à la fouille de contexte.	86
Tableau 7. Nombre de citations des publications concernées par ces travaux.	91
Tableau 8. Analyse bibliométrique des publications liées à la sélection de services par intention et contexte.....	98
Tableau 9. Citations par publication et par intervalle.....	103
Tableau 10. Ensemble de définitions formelles définissant un espace de services (d'après [129, 131]).	113
Tableau 11. Analyse bibliométrique des publications présentant l'espace de services.	118
Tableau 12. Mots-clés utilisés pour l'étude de littérature autour de la gestion de ressources.	126
Tableau 13. Analyse bibliométrique des publications liées la gestion de ressources dans les SIP.....	131

I Introduction

Le présent document synthétise une partie importante de mes travaux de recherches, lesquels peuvent être caractérisés par l'apport de la notion de contexte à différents systèmes (*groupware systems* ou collecticiels en français, *middleware* ou intergiciel en français, ou encore Systèmes d'Information). Ces recherches sont le fruit d'une carrière démarrée en 2006, date de soutenance de ma thèse, et qui se poursuit jusqu'à aujourd'hui (2020), à l'Université Paris 1 Panthéon Sorbonne.

Tout au long de ma carrière, j'ai eu l'opportunité et le privilège d'avoir intégré des nombreuses équipes, travaillant sur différents domaines de l'Informatique. Tout d'abord, entre octobre 2002 et septembre 2006, j'ai réalisé ma thèse à l'Université de Joseph Fourier – Grenoble I, au sein de l'équipe SIGMA, une équipe reconnue dans le domaine des Systèmes d'Information, dont l'axe « multimédia » (actuelle équipe STEAMER) était spécialisé sur l'adaptation. Au sein de l'axe « multimédia », j'ai pu mener des recherches autour de l'adaptation des collecticiels, et plus particulièrement l'adaptation de l'information de conscience de groupe (« *group awareness* »), dont le support est une caractéristique marquante de ce type d'application. C'est à ce moment que mes recherches autour de la notion de contexte ont démarré, avec notamment la proposition d'un modèle de contexte tenant compte des aspects organisationnels.

Après ma thèse, j'ai pu continuer mes recherches dans la communauté CSCW (« *Computer Supported Cooperative Work* »), dont sont issus les travaux sur les collecticiels et la conscience de groupe, en intégrant l'équipe ECOO (actuelle COAST) au LORIA à Nancy, pendant une année d'ATER (2006-2007) à l'IUT Nancy Charlemagne de l'Université Nancy 2.

Suite à cette année d'ATER, j'ai eu l'opportunité d'intégrer la prestigieuse K.U.Leuven pour un poste de post-doc au sein du laboratoire DistriNet, dans le cadre du projet européen IST-MUSIC. Pendant cette période (2007-2008), j'ai été confrontée à un nouvel environnement, mais surtout à un nouveau domaine, celui des Systèmes Répartis. Alors résolument intégrée dans la communauté de l'Informatique Ubiquitaire, j'ai dû m'imprégner de nouveaux concepts, comme les réseaux pair-à-pair et la notion de service, afin de mieux m'adapter et d'ainsi mieux apporter une véritable plus-value à mon équipe et au projet IST-MUSIC.

A partir de septembre 2008, j'ai pu intégrer le Centre de Recherche en Informatique de l'Université Paris 1 Panthéon Sorbonne en tant que Maître de Conférences. Là encore, j'ai dû faire preuve d'adaptation, en intégrant une équipe Système d'Information réputée par ses compétences en Ingénierie des Exigences et Ingénierie de Services. J'ai pu alors m'imprégner des nouveaux concepts, telle la notion d'intention, que j'ai pu intégrer à mes recherches, tout en apportant à l'équipe mes compétences autour de la notion de contexte.

Tout au long de ces années, j'ai ainsi pu intégrer plusieurs équipes appartenant à différentes communautés de l'Informatique, en incorporant à mes recherches des concepts issus de ces communautés et en leur apportant en retour mes propres contributions. La notion de contexte apparaît ainsi comme véritable fil conducteur de mes recherches, puisque cette notion a été mise à contribution dans tous les domaines traversés par mes travaux (CSCW, Informatique Ubiquitaire, Systèmes d'Information). Tel un fil d'Ariane, la notion de contexte a été appliquée dans ces différents domaines, avec des contributions pour chaque communauté.

Toutes ces expériences et ces contributions convergent aujourd'hui dans l'évolution des Systèmes d'Information (SI), dans ce qu'on appelle ici Systèmes d'Information Pervasifs (SIP). En effet, l'introduction des nouvelles technologies et tendances dans les SI conduit inéluctablement à leur évolution vers une nouvelle génération de Systèmes d'Information, les Systèmes d'Information Pervasifs. Ces nouvelles technologies impactent en premier lieu les infrastructures utilisées par ces systèmes, mais leur influence ne se limite pas à ce niveau purement technique. Tous les niveaux d'un Système d'Information peuvent être impactés. Dans une vision schématique (pour ne pas dire

simpliste), on peut considérer que ces nouvelles technologies et les opportunités qu’elles apportent sont susceptibles d’influencer les infrastructures, mais également les services offerts, les applications et processus métiers, voire même le pilotage de ces systèmes (voir Figure 1).

Les nouvelles technologies, comme l’IoT, le *Cloud* et *Fog Computing*, apportent plus de dynamisme aux Systèmes d’Information et permettent ainsi d’envisager des SI plus souples, capables de mieux s’adapter aux changements. La notion de contexte peut alors contribuer à atteindre cette souplesse, devenue nécessaire pour mieux tenir compte de l’environnement dynamique vers lequel se dirigent petit à petit les Systèmes d’Information. L’information de contexte peut ainsi être capturée et remontée, niveau par niveau, tels des événements, et contribuer à l’adaptation de chaque niveau et du système dans sa totalité.

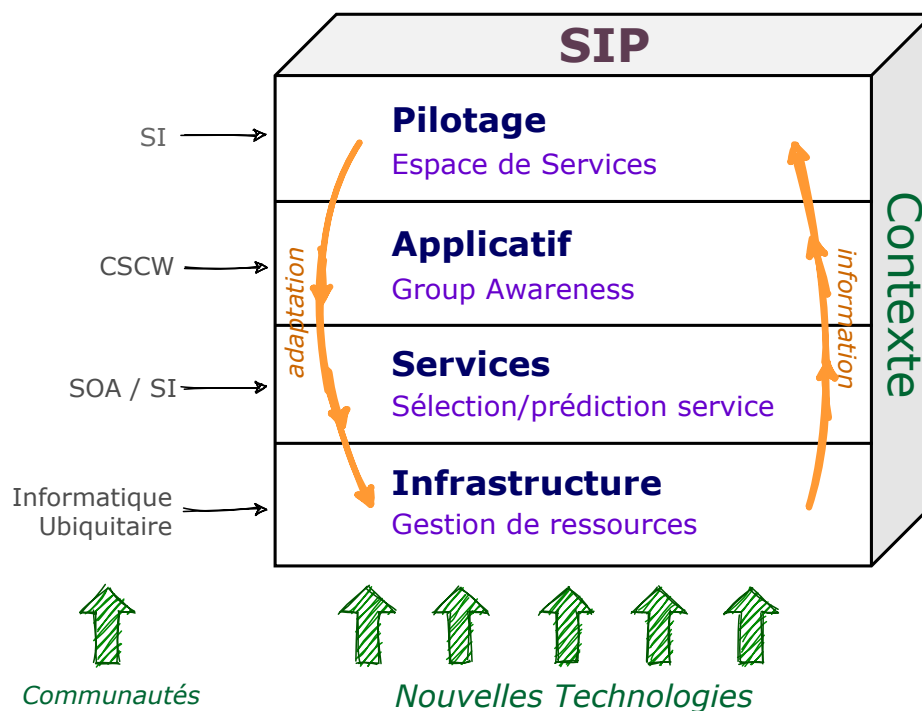


Figure 1. Positionnement des contributions présentées dans une vision schématique d’un Système d’Information Pervasif.

Au cours de ces années, les contributions que j’ai pu réaliser sur les différentes communautés retrouvent aujourd’hui une application directe dans cette vision, illustrée par la Figure 1. Chaque contribution présentée dans ce document (notée en violet à la Figure 1) va ainsi pouvoir s’appliquer à un des différents niveaux de ces systèmes. Dans chaque contribution, la notion de contexte apparaît comme l’élément clé permettant notamment l’adaptation à chaque niveau.

Chacune des contributions présentées dans ce document converge ainsi vers cette nouvelle génération de SI. Ces contributions sont présentées par la suite, non dans un ordre chronologique, mais plutôt dans un ordre logique, organisées en fonction de la communauté dans laquelle la contribution a été élaborée. La notion de contexte représentant le fil conducteur de ces travaux, il est important de bien comprendre ce concept et ses caractéristiques. Nous commençons donc ce document par un état de l’art sur la notion de contexte et son ingénierie. Cette « *roadmap* », issu de [130], synthétise mon expérience et ma vision de la notion de contexte et de son application (ce qu’on entend ici par son ingénierie). Cet effort de vulgarisation, d’abord destiné à l’enseignement, établit les bases nécessaires pour la compréhension de ce concept à travers un ensemble de dimensions considérées nécessaires à son ingénierie. Ces dimensions trouvent ensuite leur application sur les différentes contributions.

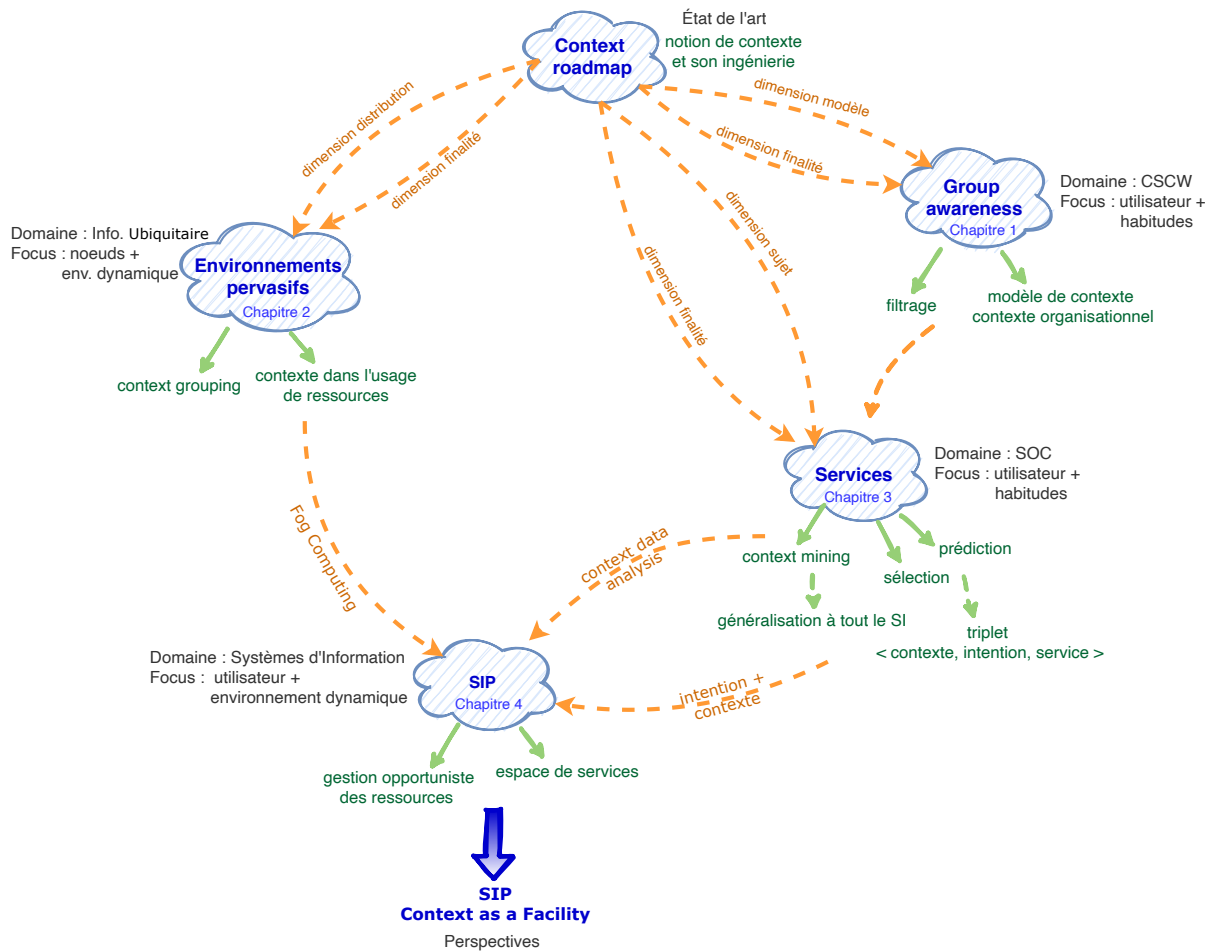


Figure 2. Cartographie des contributions et de leur évolution.

La première de ces contributions (chapitre 1) s'applique à la communauté CSCW. Il s'agit des travaux réalisés majoritairement pendant ma thèse (2002-2006), centrés sur l'adaptation de l'information de conscience de groupe (*group awareness*). Ces travaux, dont le focus principal sont les utilisateurs finaux, organisés en équipes, soulèvent la question des habitudes de travail et proposent deux contributions majeures : un modèle de contexte incluant le contexte dit organisationnel ; et un mécanisme de filtrage pour l'information de conscience de groupe réalisé à partir des profils représentant les habitudes des utilisateurs.

On retrouve ces contributions, ou plutôt leur influence, dans mes travaux réalisés au sein de la communauté SOC (*Service Oriented Computing*). Dans ces travaux (chapitre 3), réalisés entre 2008 et 2014, on retrouve ce focus résolument tourné vers les utilisateurs et le support à leurs habitudes de travail à travers trois contributions autour de la sélection et la prédiction de services, mais également une contribution sur l'analyse de données contextuelles, ce qu'on appelle ici « *context mining* ». Ces derniers considèrent comme une des questions majeures mises en avant par la *roadmap*, la pertinence d'une information de contexte pour un utilisateur, et à *fortiori*, pour une application, et le questionnement sur la généralisation du support au contexte à tout un Système d'Information. Les travaux sur la sélection et la prédiction de services, quant à eux, introduisent le triplet « *< intention, contexte, service >* » selon lequel un service est proposé afin de satisfaire des intentions dans un certain contexte d'usage.

Ces contributions ont ainsi exploré différentes dimensions présentes dans la *roadmap*, et notamment les dimensions « modèle » et « sujet ». Ces contributions ont été complétées par des travaux plus

techniques, ancrés dans le domaine de l'Informatique Ubiquitaire. Ces travaux (chapitre 2), menés entre 2008 et 2016, explorent aussi une autre dimension mise en avant par la *roadmap*, la distribution des données de contexte. Ces travaux introduisent également la question de la dynamique de l'environnement. Le focus n'est plus l'utilisateur (en tout cas, pas directement), mais l'environnement devenu dynamique et les nœuds qui le composent. Ces travaux utilisent ainsi l'information de contexte pour un meilleur usage des ressources présents dans l'environnement. On voit apparaître, lors de ces travaux, la communauté *Fog Computing* et la possibilité d'utilisation des ressources de proximité pour l'exécution de tâches diverses.

Toutes ces contributions se rejoignent dans l'évolution des Systèmes d'Information vers des Systèmes d'Information Pervasifs (SIP). Cette nouvelle génération de systèmes fait l'objet de mes dernières recherches, réalisées à partir de 2016 et présentés dans le chapitre 4. Ces recherches, ancrées dans la communauté SI, ont pour focus non plus l'utilisateur ou l'environnement isolés, mais les deux, ensemble, à travers notamment la définition d'un cadre conceptuel pour les SIP (nommé Espace de Services) et l'étude de la gestion opportuniste de ressources dans ces systèmes.

La Figure 2 représente une cartographie illustrant ces contributions et leurs influences réciproques. Les contributions illustrées dans cette carte représentent les contributions les plus marquantes de ma carrière, celles où j'ai eu un apport plus significatif pour la communauté en question. Ces travaux menés dans les différentes équipes et communautés illustrent également ma vision de ce que doit être un travail de recherche en Informatique : un travail collaboratif, de construction collective, par l'apport des uns et des autres à la résolution de différents problèmes de recherche. Dans chaque communauté, j'ai été amené à m'y intégrer, à m'imprégner des concepts et des problématiques afin de pouvoir apporter une plus-value à la communauté. Ces travaux étant donc menés au sein de différentes équipes, le restant de ce document est volontairement écrit à la première personne du pluriel (« nous »), afin de souligner ce qui est peut-être la caractéristique majeure d'un Système d'Information Pervasif, sa pluridisciplinarité.

Ce document s'organise donc comme suit : on démarre par un état de l'art sur la notion de contexte (partie II), avant d'introduire les contributions. La partie III détaille les contributions. Le chapitre 1 introduit mes contributions dans les collecticiels (« *Groupware systems* ») ; le chapitre 2 présente les contributions réalisées autour des environnements pervasifs (« *Pervasive systems* ») ; alors que le chapitre 3 présente mes contributions dans l'orientation service (« *Service-oriented systems* »). Enfin, le chapitre 4 introduit mes recherches autour des Systèmes d'Information Pervasifs, avant de conclure (partie IV) avec mes perspectives et travaux à venir.

II État de l'art

La notion de contexte peut être définie comme tout élément d'information permettant de caractériser la situation d'une entité, qu'il s'agisse d'une personne, d'un endroit ou d'un autre objet (utilisateur, application, etc.), considérée pertinente pour l'interaction entre l'utilisateur et l'application [74]. Cette notion accompagne mes recherches depuis mes travaux de thèse. Elle a été déclinée sur différents problématiques de recherche, sur diverses communautés de l'Informatique. Véritable fil d'Ariane de mes travaux, il est important de bien comprendre cette notion et ses caractéristiques avant de pouvoir présenter les contributions qui l'appliquent. Ce chapitre présente ainsi un état de l'art sur la notion de contexte. Issu de [130], cet état de l'art synthétise mon expérience et ma perception de la notion de contexte et de son ingénierie. Cet état de l'art, destiné d'abord à la vulgarisation visant des étudiants en Informatique, se présente sous la forme d'une *roadmap*, présentant différentes dimensions nécessaires à la prise en compte logicielle de cette notion.

1 Avant-propos

La notion de contexte devient de plus en plus utilisée aujourd'hui au sein d'applications qu'on pourrait qualifier « d'intelligentes », car capables d'observer l'environnement et de réagir en conséquence. On observe déjà ce phénomène avec un nombre croissant d'applications capables d'observer des éléments de l'environnement dont par exemple la localisation de l'utilisateur, son activité physique, etc. L'observation de l'environnement est désormais possible grâce à différents types d'applications. Le développement de capteurs, d'actionneurs, de nano-ordinateurs et d'autres technologies à faible coût liées à l'Internet des Objets (*Internet of Things* - IoT) permet aux développeurs de proposer facilement des applications qui observent l'environnement physique et interagissent avec lui. Ce type d'application fait déjà partie de notre vie quotidienne mais, dans la plupart des cas, son développement est encore effectué de manière *ad hoc*, malgré toutes les recherches qui ont été faites autour de la notion de contexte et des applications sensibles au contexte (applications capables d'adapter leur comportement aux variations observées dans le contexte d'usage [13, 16]). Aujourd'hui, le principal défi ne réside plus dans les technologies elles-mêmes, mais principalement dans la compréhension des enjeux et dans l'exploration des possibilités offertes par ces nouvelles technologies. En effet, afin d'aller plus loin dans l'usage de la technologie, il est nécessaire de mieux comprendre la notion de contexte et ses enjeux, car cette notion est centrale pour la conception et la réalisation de nouvelles solutions.

Les systèmes sensibles au contexte peuvent être considérés comme des applications capables de répondre à ces changements. Ils sont définis en tant qu'applications capables d'observer les changements de contexte et d'adapter leur comportement en conséquence [13, 16]. Par rapport aux applications traditionnelles, les applications sensibles au contexte peuvent être considérées comme plus complexes car elles doivent faire face à des environnements hétérogènes et dynamiques. Elles doivent fonctionner, souvent en continu, dans des conditions changeantes. Elles doivent observer différents éléments de l'environnement et réagir à leurs changements, souvent en utilisant des plateformes informatiques très limitées (par exemple, des nano-ordinateurs ou des *smartphones* avec des limitations de batterie et de connectivité). Un tel environnement d'exécution dynamique et contraint a un impact important sur l'architecture et le développement de ces logiciels, notamment en termes de modularité, d'intégration, d'interopérabilité et de nombre croissant de contraintes non fonctionnelles (par exemple, la robustesse et l'évolutivité). Dans ces conditions, les qualités traditionnellement attendues d'un logiciel, telles que la flexibilité, la dynamique, la modularité et l'extensibilité, deviennent difficiles à satisfaire, notamment avec les processus de développement *ad hoc* qui sont souvent adoptés lors du développement d'applications sensibles au contexte, comme observé dans [16, 17].

Un aspect en particulier rend complexe le développement de ces applications : la notion de contexte elle-même. Celle-ci correspond à un concept large et ambigu qui a été étudié et défini de plusieurs manières différentes, tant en informatique que dans d'autres domaines [24,32,33,171]. Supporter cette notion dans une application informatique soulève plusieurs défis allant de l'identification des informations de contexte pertinentes, de leur acquisition et leur modélisation, jusqu'à leur interprétation et leur exploitation à différentes fins [13, 24, 132,133]. Il devient rapidement difficile pour des concepteurs non-experts de concevoir et de construire de nouvelles applications utilisant cette notion.

La compréhension de la notion de contexte et de son support est une tâche complexe mais nécessaire. Elle est complexe parce que la notion de contexte est elle-même une notion complexe et ambiguë, dont l'intégration au sein d'une application implique plusieurs questions techniques. Elle est nécessaire parce que ce n'est qu'en comprenant cette notion et les manières dont elle peut s'intégrer à une application qu'on peut réellement explorer tout son potentiel et toutes les opportunités qu'elle ouvre. Seule une meilleure compréhension de cette notion permettra l'établissement d'un véritable processus d'*ingénierie de contexte*, permettant le développement de nouvelles applications sensibles au contexte complexes et extensibles. Plus que jamais, il devient nécessaire de former une nouvelle génération de concepteurs et de développeurs capables de raisonner autour des éléments de contexte de la même manière qu'ils sont formés pour intégrer des concepts tels que les composants et la programmation orientée objets.

Il est ainsi important de fournir à ces jeunes concepteurs non-experts les connaissances nécessaires pour raisonner autour des questions et des défis liés au support et à la gestion du contexte. Nous avons pu identifier un ensemble de dimensions qui peuvent être considérées comme nécessaires pour un tel support [132], et nous avons analysé l'impact de la qualité des informations de contexte sur ces dimensions [133]. Ces dimensions ont été vues comme des lignes directrices dans un processus d'analyse des besoins, aidant les concepteurs non-experts à identifier la problématique autour du support de la notion de contexte dans de nouvelles applications sensibles au contexte. Une étude de littérature a ainsi été menée [130] afin de mettre en évidence les solutions existantes et les questions ouvertes liées au support et à la gestion de contexte. Cette étude est destinée à servir de base à la formation de nouveaux « ingénieurs de contexte », capables de comprendre et de construire de nouvelles applications sensibles au contexte, notamment pour les Systèmes d'Information de demain.

Cet état de l'art démarre ainsi par l'étude de trois scénarios d'applications qui illustrent les types d'usages qu'on peut faire de l'information de contexte et les difficultés pour sa mise en œuvre. Ces scénarios seront utilisés tout au long de ce chapitre en tant qu'exemple, afin d'illustrer différentes questions soulevées par la roadmap. Ces scénarios sont suivis par une étude menée auprès d'un groupe de 50 étudiants en master MIAGE¹. L'objectif de cette étude est de mieux comprendre la perception que ces jeunes développeurs ont de la notion de contexte. La roadmap et chacune de ses dimensions sont ainsi discutées, à la fois dans ses aspects fonctionnels, mais également dans la prise en compte des aspects qualitatifs, particulièrement pertinents lorsqu'on considère l'information de contexte. La roadmap soulève ainsi de nombreuses questions qui doivent être prises en considération lors de la conception d'applications sensibles au contexte. Des nombreux éléments de réponse mis en lumière par l'étude de la littérature sont alors indiqués pour chaque dimension.

2 Exemples d'applications

Avant d'examiner les défis liés au développement de nouvelles applications sensibles au contexte, il est important de considérer quelques scénarios pouvant explorer la notion de contexte. Plusieurs

¹ MIAGE : Méthodes Informatique Appliquées à la Gestion d'Entreprises.

domaines d'application peuvent bénéficier de cette notion, dont les villes intelligentes (« *smartcities* ») et l'agriculture intelligente (« *smart agriculture* »). Afin de démontrer l'intérêt de telles applications, considérons trois scénarios illustratifs.

Le premier scénario est celui d'un système d'alerte aux inondations, proposé par [111, 223]. Ce scénario, appelé *GridStix*, envisage un réseau de capteurs sans fil (*Wireless Sensor Network - WSN*) déployé sur les rivières *Ribble*, en Angleterre, et *Dee* au Pays de Galles. Chaque nœud du *GridStix* (illustré dans la Figure 3a) est constitué de capteurs pour la hauteur de l'eau et le débit, et d'une alimentation fournie par des batteries rechargées par des panneaux solaires. Les nœuds sont équipés des réseaux de communications Wi-Fi (802.11b) et Bluetooth, utilisés pour la transmission de données entre les nœuds, et d'un nœud de liaison, équipée d'une connexion GSM, capable de remonter les informations à un serveur central. En plus des différents modes de transmission et de collecte de données, chaque nœud peut être activé ou désactivé en fonction du niveau de charge de sa batterie et de l'état de la rivière. Sur la base des informations collectées par les nœuds *GridStix*, une application d'alerte aux inondations considère un modèle stochastique pour la prévision des situations d'inondation. Elle peut également effectuer des actions d'adaptation, telles que l'activation ou la désactivation des nœuds pour économiser la batterie en fonction de l'état de celle-ci et de l'état des nœuds voisins (afin d'éviter un manque d'observations sur certains segments de la rivière). Pour assurer une telle reconfiguration des nœuds, le système doit observer les changements de contexte survenus sur les nœuds afin d'y répondre de manière réactive et proactive. Cela implique non seulement l'acquisition de données collectés par les capteurs *GridStix*, mais aussi le transfert de ces données et leur mise à disposition pour le traitement de modèles stochastiques. Cela implique également la spécification de politiques d'adaptation qui énoncent les actions requises pour conduire le système à une configuration qui correspondrait au mieux à son contexte actuel (par exemple, éteindre un nœud dont la batterie est faible par un nœud voisin dont la batterie est pleine, pour que le premier puisse se recharger).



Figure 3. (a) Une installation *GridStix* [223] et (b) un prototype de système d'observation de stress hydrique [130].

Un autre exemple de scénario est celui envisagé par le projet CC-Sem², dont l'objectif était celui de développer une plateforme intégrée pour la surveillance, le contrôle et la planification intelligents de la consommation et de la production d'énergie dans des scénarios urbains. Ce projet assume que les capacités de surveillance, de contrôle et de gestion de la consommation et de la production d'énergie

² <https://www.fing.edu.uy/inco/grupos/cecal/hpc/cc-sem/>

sont très importantes pour la mise en œuvre du paradigme de la ville intelligente. Pour cela, le projet envisage l'utilisation de compteurs d'électricité intelligents pour collecter des informations sur la consommation des foyers, ainsi que l'utilisation d'autres capteurs pour collecter des informations telles que la température, l'humidité et les conditions météorologiques. Les données collectées pourraient ensuite être analysées afin d'identifier de possibles modèles de consommation d'énergie.

Ces modèles peuvent servir de base à des recommandations suggérant des mesures d'économie pour les utilisateurs finaux ou des maisons intelligentes (par exemple, réduire l'intensité de la climatisation ou du chauffage en fonction des informations météorologiques), mais également à des mesures préventives telles que la coupure des chauffe-eaux et des systèmes de climatisation en cas de surcharge du réseau électrique. Ces mêmes modèles pourraient également servir aux fournisseurs d'énergie afin de mieux prévoir la consommation et d'anticiper les actions préventives afin d'éviter les problèmes dus à la surconsommation.

La réalisation d'un tel scénario exige non seulement le déploiement de compteurs d'énergie intelligents et de capteurs de température/humidité, mais également le déploiement d'une infrastructure appropriée. Une telle infrastructure est nécessaire pour la collecte et le transfert de données brutes, ainsi que pour leur analyse à l'aide de techniques *Big Data*. Elle implique également le traitement des questions sur la confidentialité et la sécurité du réseau, nécessaires pour assurer la sécurité des données de consommation personnelle mais aussi pour empêcher des actions malicieuses.

Enfin, un troisième scénario possible serait celui d'une application IoT dans le cas de l'agriculture intelligente. En effet, l'utilisation de capteurs et d'actuateurs ouvre de nouvelles perspectives pour l'agriculture. En utilisant différents capteurs, tels que des capteurs d'humidité, de stress hydrique, de luminosité ou encore de température, il est possible de mieux surveiller l'état de santé général des cultures et d'optimiser leur production. Des petites cultures cultivées sous des serres, comme les fleurs, les tomates, les fraises ou les épices, peuvent ainsi bénéficier de la surveillance constante, notamment de la température et des conditions hydriques. Les données observées par des capteurs situés directement sur le terrain peuvent être utilisées principalement pour la prise de décision : les producteurs peuvent recevoir des rapports quotidiens sur leurs cultures et décider des actions préventives pour améliorer la production. Ces données peuvent également être utilisées pour prendre des mesures automatiques, en contrôlant, par exemple, l'approvisionnement en eau d'une parcelle en fonction du stress hydrique des cultures.

La Figure 3b montre un prototype de système de surveillance du stress hydrique dans lequel un microcontrôleur Arduino est utilisé d'une part pour surveiller le stress hydrique, grâce à un capteur d'humidité du sol, et d'autre part pour commander une pompe à eau. Des systèmes commerciaux similaires existent déjà pour un usage résidentiel, comme par exemple le système Daisy³. Néanmoins, le déploiement de ce type d'équipement sur une culture professionnelle exige de prendre en compte non seulement son infrastructure et son approvisionnement en énergie, mais également de relever des défis tels que le choix des capteurs et des données à surveiller par rapport à la culture, ou encore le choix de la fréquence nécessaire pour la collecte de ses données en fonction des besoins de la culture et de l'environnement. On peut aussi parler du choix des seuils appropriés pour déclencher des actions, ou encore le choix de la manière de représenter ou de stocker les données collectées pour une meilleure analyse de ces données. La qualité de la prise de décision (automatique ou non) dépend de ces questions, depuis la sélection des données à observer jusqu'à leur analyse.

Ces scénarios illustrent quelques-unes des questions qui doivent être soulevées lors de la conception de ce type d'application : quel type d'information observer ? ; quel capteur utiliser ? ; comment exploiter ces données obtenues ? ; quel modèle utiliser pour permettre une bonne interprétation des données ? ; etc. Un concepteur non-expert peut facilement se perdre au milieu ces différentes options.

³ <http://daisy.si/>

Le manque de connaissances sur le sujet peut également l'amener à sous-estimer certaines questions, comme par exemple, l'impact de la qualité des données sur l'application. La localisation est un bon exemple de ce type de question. La localisation peut être représentée de différentes manières (coordonnées GPS, adresse postale, etc.) et est assujettie à des possibles erreurs lors de son acquisition, selon les conditions d'observation. La Figure 4 ci-dessous illustre l'observation d'un même trajet réalisée utilisant deux terminaux distincts (un iPhone 5 pour la Figure 4a et un LG Nexus 5 pour la Figure 4b). Même si les observations ont été réalisées de manière simultanée, la Figure 4 montre d'importantes différences entre les données acquises, illustrant assez bien la question de qualité de la donnée de contexte et de son imprécision. La promotion des connaissances nécessaire à l'ingénierie de contexte auprès des concepteurs non-experts est nécessaire si on souhaite stimuler le développement des nouvelles applications sensibles au contexte.



Figure 4. Observation simultanée d'un même tracé par deux terminaux distincts.

3 Public cible

Le nombre d'applications explorant la notion de contexte est en constante augmentation grâce au développement des différentes technologies, dont les capteurs intégrés aux *smartphones* et aux tablettes, mais également associés à des dispositifs IoT. Avant d'être considérées comme sensibles au contexte, des applications telles que celles présentées précédemment peuvent être considérées par les consommateurs et par des concepteurs non-experts comme des applications « intelligentes », car elles proposent ce qui peut être perçu comme un comportement intelligent. En effet, être capable d'observer l'environnement physique et d'adapter dynamiquement son comportement sans aucune intervention humaine sont des comportements souvent perçus comme « intelligents » (ou « *smart* ») par les consommateurs et, par conséquent, de plus en plus de développeurs envisagent de les intégrer dans leurs nouvelles applications.

Même si ces applications sont de plus en plus populaires, leur conception et leur mise en œuvre sont souvent peu ou mal maîtrisées par des concepteurs non-experts. Afin d'illustrer cette situation, nous avons invité deux groupes d'étudiants en 2^{ème} année de master MIAGE à participer à une enquête. Dans cette enquête, les étudiants ont été invités à répondre à une série de 50 questions sur leurs pratiques en matière de conception/ingénierie d'applications dites « intelligentes ». Ces questions portent sur différents aspects concernant le projet, la conception et le développement de ces applications. Les étudiants, organisés en deux groupes comprenant chacun une vingtaine de

personnes, étaient composés, dans leur majorité, par des étudiants en apprentissage, avec 1 à 3 ans d'expérience, ainsi que des étudiants comptant environ 4 à 6 mois de stage, autant sur le développement de logiciels (principalement Web) que sur les Systèmes d'Information.

Avant de soumettre les élèves à l'enquête, ceux-ci ont participé à petite expérience à l'aide d'un nano-ordinateur RaspberryPi ZeroW (voir Figure 5). Les étudiants devaient construire une petite application qui observait la température dans la pièce et qui réagissait à la température en modifiant la couleur d'une LED (couleur rouge si la température est supérieure à un seuil, jaune si elle est inférieure, et verte sinon). Les étudiants ont été organisés en petits groupes. Chaque groupe a reçu un kit contenant un RaspberryPi ZeroW, un capteur de température BMP 280 et une triple LED (voir Figure 5), ainsi qu'une carte SD contenant le système d'exploitation *Raspbian Jessie* et deux exemples de code pour manipuler le capteur et la LED. Ce n'est qu'après avoir connecté les capteurs et construit l'application que les étudiants ont reçu le questionnaire à remplir.

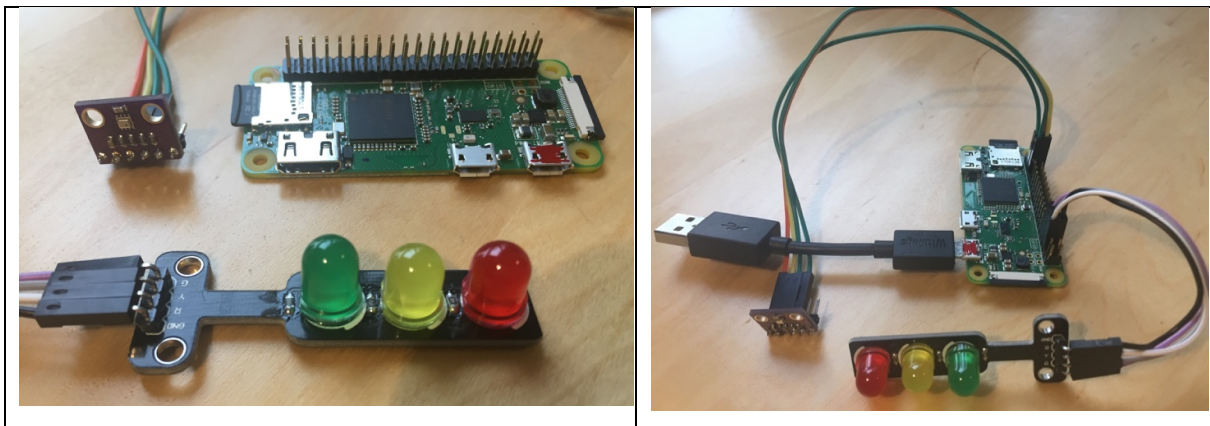


Figure 5. Kit distribué aux étudiants comportant un RaspberryPi ZeroW, un capteur de température I2C BMP 280 et un triple LED [130].

Bien que tous les étudiants concernés aient une formation en Informatique et soient habitués au développement logiciel, seule la moitié parmi eux (48,83%) ont déclaré avoir déjà participé au développement d'une application intelligente. Parmi ceux qui n'y avaient jamais participé, 63,6% prévoyait de (ou aurait souhaité) participer à ce type de projet. Parmi ceux qui avaient déjà une expérience, les applications mobiles et Web semblaient être les plus concernées, comptant respectivement pour 66,67% et 42,85% des réponses, suivies par les applications de géolocalisation et les applications de maison intelligente (concernant chacune 28,57%). Indépendamment de leur expérience antérieure, presque tous les étudiants ont déclaré connaître couramment utilisées pour les applications sensibles au contexte : 100% ont déclaré connaître RaspberryPi (utilisé dans l'expérience précédant le questionnaire) et les téléphones Android, 86% pour l'iPhone, 83% et 81% pour l'iPad et l'iPad Pro, 81% pour les tablettes Android et 48,83% pour les Arduino. Une tendance similaire est apparue pour les plateformes qu'ils déclaraient avoir déjà utilisées ou souhaiter utiliser : 93% ont déclaré utiliser/vouloir utiliser le téléphone Android comme plateforme cible, 72% pour RaspberryPi, 67,44% pour les tablettes Android et l'iPhone, 30,23% pour Arduino et même 11,62% pour les plateformes SmartTV. Ces chiffres illustrent bien l'intérêt croissant que ces jeunes développeurs portent à la création d'applications pour les plateformes mobiles et IoT, et à l'exploration des possibilités de ces plateformes.

Il faut remarquer que tous ces étudiants avaient déjà été brièvement initiés à la notion de contexte et à l'Informatique Ubiquitaire pendant leur scolarité. Bien que la notion de contexte ne soit pas totalement inconnue à ces étudiants, ils ne peuvent toujours pas être considérés comme des « experts », étant pour la plupart des novices en la matière. Interrogés sur les plateformes de

programmation (bibliothèques, API, *framework* ou *middleware*), presque aucune technologie propre à la prise en compte de la notion de contexte et au développement d'applications IoT n'a été citée. Seuls 4 étudiants ont cité Pi4J, la bibliothèque proposée dans l'expérience qu'ils ont réalisée avec le RaspberryPi. Parmi ceux ayant une expérience de développement antérieure, 36,36 % ont déclaré utiliser un accès direct (*ad hoc*) aux appareils et 31,81 % ont déclaré utiliser des appels directs au Système d'Exploitation ou aux langages de programmation utilisés. Les multiples plateformes et technologies proposées dans la littérature (*e.g.* [66,51, 87, 217]) leur semblent inconnues.

De même il a été demandé aux étudiants ayant une expérience préalable dans le développement d'applications dites « intelligentes » à quel moment du (ou des) projet(s) les technologies utilisées ont été choisies : 22,72% n'ont pas pu répondre (« je ne sais pas »), autant que « au tout début, c'était prédéfini dans le cahier des charges du projet ». Interrogés sur la difficulté d'utilisation de ces technologies, 31,81% des élèves ont évalué la difficulté comme étant « moyenne », en soulignant que certains éléments étaient inconnus et qu'ils devaient apprendre à manipuler ces technologies. Il est intéressant de noter qu'aucun étudiant n'a évalué la difficulté comme étant « facile » et que seulement 9% environ l'ont déclarée comme étant « difficile », en soulignant qu'ils n'avaient jamais utilisé de telles technologies auparavant. En tout cas, environ 90% des étudiants ayant une expérience de développement dans ce type d'application ont déclaré qu'il était nécessaire d'avoir davantage des connaissances sur ces technologies. D'ailleurs, 50% ont considéré que « cela aide beaucoup » et 40% ont choisi « cela pourrait aider, mais ce n'est pas obligatoire ». Une meilleure diffusion des connaissances sur ces technologies semble ainsi être un levier intéressant pour faire progresser les applications sensibles au contexte.

Enfin, à la question « avez-vous déjà entendu parler de l'informatique sensible au contexte », 52,27% des étudiants ont répondu « oui, j'ai quelques notions sur le sujet » et 22,72% ont répondu « oui, très légèrement », ce qui correspond aux valeurs attendues puisque les deux groupes d'étudiants ont déjà une certaine expérience universitaire dans ce domaine. Néanmoins, lorsqu'on leur a demandé si l'application qu'ils avaient construite ou souhaitaient construire était « sensible au contexte », 36,36% ont répondu « je ne sais pas ». De même, lorsqu'on leur a demandé s'ils connaissaient la notion de « contexte », 43,18% ont répondu « oui, légèrement », 25% ont déclaré avoir quelques connaissances sur le concept et 25% ont indiqué avoir quelques difficultés avec celui-ci. Ces difficultés sont confirmées par la question sur comment cette notion pouvait correspondre à leurs besoins passés et futurs, seulement 34,09% des étudiants ont pu répondre à cette question. A travers ces éléments, il devient clair que la compréhension de la notion de contexte et son utilisation reste un défi pour ces étudiants, même pour ceux qui ont déjà une certaine expérience avec ces applications.

Cette enquête illustre la perception d'un public clé pour le développement de la notion de contexte : les concepteurs de logiciels qui, malgré une certaine expérience, ne sont pas nécessairement des experts en matière de conception et de développement d'applications sensibles au contexte, mais qui pourraient être amenés à participer à ce type de projet dans un avenir proche.

4 Roadmap pour l'ingénierie de contexte

Comme on peut observer dans les scénarios présentés précédemment, l'ingénierie des applications utilisant la notion de contexte peut devenir une tâche difficile en raison de la nature complexe de cette notion. L'ingénierie de telles applications implique la prise en compte de différents aspects concernant la notion de contexte et son support, incluant la collecte, le transfert et l'analyse de données de contexte. Les concepteurs non-experts, tels que les étudiants mentionnés ci-dessus, sont laissés seuls pour comprendre et identifier les concepts et les composants nécessaires à la construction de telles applications. L'acquisition des connaissances nécessaires au développement d'applications explorant la notion de contexte représente donc un défi pour les concepteurs non-experts.

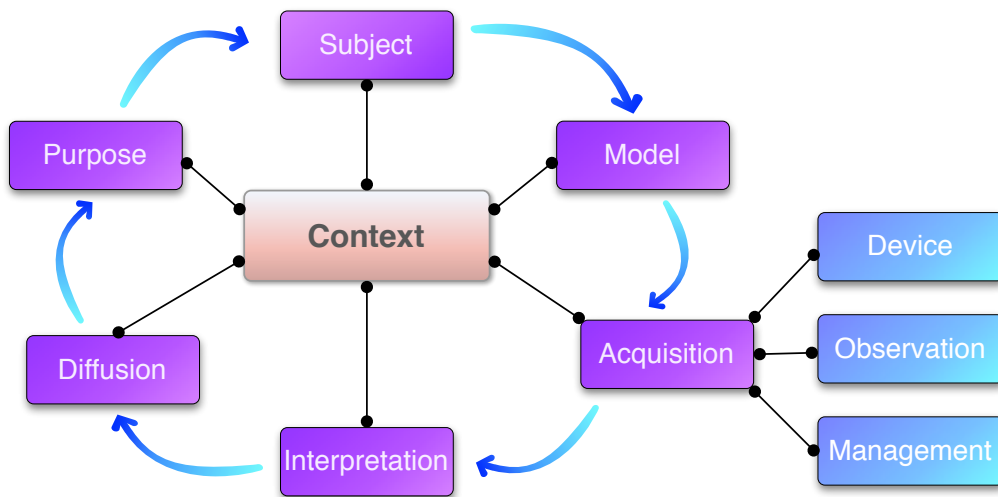


Figure 6. Dimensions de la roadmap pour l'ingénierie de contexte [132].

Les enjeux pour la prise en compte de l'information de contexte sont ainsi multiples. Différents défis liés au support logiciel de la notion de contexte ont pu être mis en avant dans [132] et organisés en 6 dimensions, considérées comme nécessaires à la prise en charge de cette notion (Figure 6) : *finalité* (« *purpose* »), *sujet* (« *subject* »), *modèle* (« *model* »), *acquisition* (« *acquisition* »), *interprétation* (« *interpretation* ») et *distribution* (« *diffusion* »). Ces dimensions forment une *roadmap* pour l'ingénierie de contexte. Chaque dimension se concentre sur un aspect précis et aborde différentes questions nécessaires à la gestion de contexte, conduisant à l'identification d'exigences fonctionnelles et non fonctionnelles qui devraient être pris en compte et satisfaits (au moins partiellement) par ces applications. Ces dimensions ne suivent pas nécessairement un ordre particulier. Comme le démontre [16], les projets développant des applications intelligentes utilisant la notion de contexte ne suivent pas un processus unique ; le processus adopté peut changer en fonction de l'équipe et du projet lui-même. Néanmoins, afin de simplifier la présentation de la roadmap, nous allons discuter des dimensions proposées en respectant l'ordre représenté à la Figure 6.

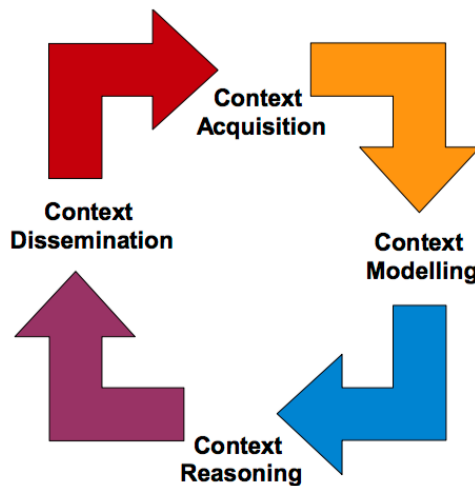


Figure 7. Cycle de vie pour la gestion de contexte proposé par [195].

Cette roadmap représente un premier pas vers une approche globale, permettant de mieux appréhender les différents aspects nécessaires à la gestion de contexte. L'objectif principal est donc d'aider les concepteurs non-experts à mieux comprendre les défis et les problèmes liés à la prise en

charge de la notion de contexte dans les applications. Certains des défis suggérés ici ont été mis en évidence par des travaux dans la littérature. Un bon exemple est le cycle de vie proposé par [195] (Figure 7), qui considère les différentes étapes (modélisation, acquisition, raisonnement et distribution) pour le traitement de données de contexte. Toutes ces étapes sont considérées dans la roadmap comme des dimensions, auxquelles s'ajoutent deux nouvelles dimensions (*finalité* et *sujet*). La roadmap se présente ainsi comme un résumé des principaux défis mis en évidence par la littérature, tels que [13, 24, 32, 171, 96, 52, 21, 158, 195]. L'objectif principal est de discuter des solutions et des questions ouvertes proposées dans la littérature pour chaque dimension à travers des exemples concrets issus de cette même littérature.

Cette roadmap proposée dans [132] a été étendue dans [133] afin de prendre en compte la notion de Qualité de Contexte (QoC – *Quality of Context*). Les informations de contexte sont naturellement dynamiques et incertaines ; elles peuvent naturellement contenir des erreurs, être dépassées ou même incomplètes [256]. La qualité des informations recueillies par un capteur peut varier en fonction de plusieurs facteurs souvent imprévisibles (*e.g.* conditions météorologiques, défaillances dans l'approvisionnement en énergie, interférences de communication, etc.). L'incertitude étant indissociable de l'information de contexte, la qualité de contexte devient une préoccupation centrale pour atteindre la fiabilité nécessaire au développement d'applications utilisant cette notion. La gestion de la qualité de contexte exige ainsi une réflexion approfondie sur les conséquences de la qualité sur le comportement de l'application, et son influence sur chaque aspect de la gestion du contexte. En effet, avec le développement croissant d'applications sensibles au contexte dans différents domaines d'application (e-santé, maisons intelligentes, transports, etc.), l'importance de la gestion de la qualité de contexte augmente considérablement, car les conséquences d'une observation de mauvaise qualité peuvent être dramatiques sur le comportement de l'application et affecter sérieusement la fiabilité de celle-ci. Ces conséquences peuvent être illustrées par le scénario de l'agriculture intelligente : des observations de faible qualité provenant d'un capteur d'humidité défectueux peuvent conduire à des décisions erronées concernant l'irrigation, exposant les cultures à un arrosage excessif ou insuffisant.

Selon Bellavista *et al.* [21], la qualité de contexte (QoC) est généralement définie comme l'ensemble des paramètres qui expriment les exigences de qualité et les propriétés des données de contexte (par exemple, la précision, la fraîcheur, la fiabilité...). Pour pouvoir observer et traiter ces propriétés, il faut tenir compte de leur influence sur chaque aspect de la gestion de contexte. La notion de qualité de contexte va ainsi affecter chacune des dimensions représentées par la roadmap. Dans chaque dimension, des défis particuliers sont pris en compte et, d'une manière ou d'une autre, influencés par la notion de qualité. La qualité de contexte est ainsi perçue comme un axe transversal, comme le montre la Figure 8, affectant tous les aspects de la gestion de contexte représentés par les autres dimensions de la roadmap. Lors de l'examen de chaque dimension, l'influence de la prise en compte de la qualité du contexte doit ainsi être considérée. Cette influence est matérialisée à la Figure 8, par les verbes attachés à chaque dimension, représentant une ligne directrice lors de la prise en compte de la notion de qualité de contexte sur la dimension. Comme pour le restant de la roadmap, le but principal n'est pas nécessairement celui de donner des solutions à ces questions, mais surtout de susciter une réflexion sur l'impact de la qualité sur la gestion de contexte.

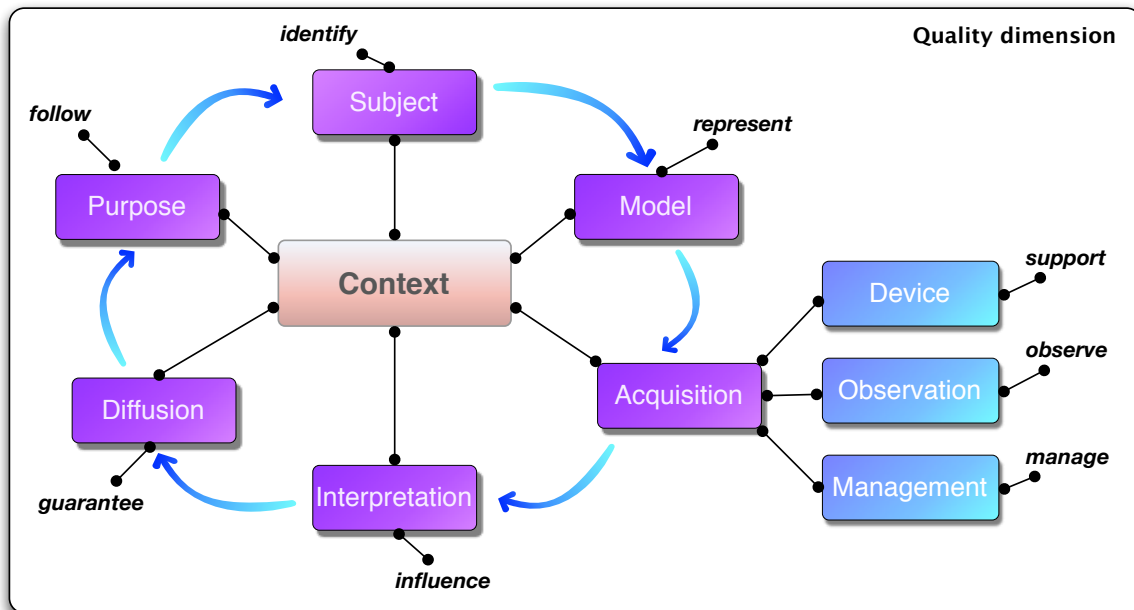


Figure 8. La Qualité de Contexte vue comme une dimension transversale, affectant chaque dimension de la roadmap [133].

4.1 Dimension Finalité

La première dimension est celle de la *finalité*, laquelle se concentre justement sur la finalité de l'utilisation des informations de contexte dans une application donnée, ainsi qu'aux moyens et aux mécanismes nécessaires pour y parvenir. Cette dimension examine ainsi les raisons pour lesquelles une application donnée a besoin de l'information de contexte. Une telle finalité a un impact significatif sur la manière dont cette information est exploitée et, par conséquent, sur les informations qui seront prises en compte, la manière dont elles seront acquises, représentées et analysées. Par exemple, en considérant les scénarios présentés précédemment, deux finalités principales peuvent être mises en évidence : l'adaptation, comme dans le scénario de *smart* agriculture, dans lequel l'approvisionnement en eau est automatiquement adapté en fonction des conditions d'humidité ; mais aussi pour la prise de décision, comme dans le scénario CC-Sem, dans lequel les informations collectées sur la consommation d'énergie sont à destination des fournisseurs d'énergie pour la prise de décision.

Lors de l'analyse de la littérature, une première classe d'applications préconisant l'usage de la notion de contexte est représentée, bien naturellement, par les applications dites sensibles au contexte, dont l'objectif est l'adaptation. Cette adaptation peut affecter différents aspects du comportement d'une application : on peut adapter le contenu fourni à l'utilisateur [225, 59], les services offerts par l'application [256, 50], ou encore la composition de l'application elle-même [200,66,87] en fonction du contexte d'exécution ou de celui de l'utilisateur. Cependant, comme pour les scénarios présentés précédemment, l'adaptation n'est pas la seule finalité pour les informations de contexte. Dans [52], les auteurs soulignent six usages courants pour ce type d'information : (i) l'affichage du contexte (c.a.d. la présentation des informations de contexte à l'utilisateur) ; (ii) l'augmentation du contexte (c.a.d. l'annotation des données avec les informations de contexte) ; (iii) la configuration en fonction du contexte (la configuration d'un service en utilisant les informations de contexte) ; (iv) les actions déclenchées par le contexte (c.a.d. le déclenchement d'actions en fonction des informations de contexte) ; (v) la médiation contextuelle (c.a.d. la modification des services ou d'un contenu afin de mieux correspondre au contexte d'utilisation) ; et (vi) la présentation en fonction du contexte (c.a.d. l'adaptation de l'interface utilisateur ou de la présentation d'un contenu).

Comme on peut le constater, la plupart de ces utilisations consiste à adapter le comportement de l'application (contenu, actions, services ou interface) en fonction du contexte d'utilisation, comme on

a pu voir sur [52], les informations de contexte peuvent également être utilisées pour l'annotation ou simplement être affichées. L'annotation de données ou d'objets à l'aide d'informations de contexte permet aux applications, aux services ou même aux utilisateurs de mieux caractériser ces informations ou ces données, tandis que l'affichage de l'information de contexte peut contribuer aux processus de prise de décision. Par exemple, Kornyshova *et al.* [145] utilisent les informations de contexte pour caractériser des fragments de méthodes sur l'ingénierie des méthodes, tandis que Ploesser *et al.* [197] ou encore Saidani *et al.* [219] utilisent les informations de contexte pour caractériser les tâches sur un modèle de workflow. Plus récemment, des applications IoT sont utilisées pour acquérir massivement les informations de l'environnement, ouvrant ainsi de nouvelles perspectives pour l'utilisation de ces informations de contexte sur de nombreuses applications. Les travaux sur les *smartcities* [195, 163, 204, 272], ainsi que de projet tels que le CC-Sem, illustrent assez bien cette tendance. Ils démontrent l'intérêt d'utiliser les informations de contexte comme support à la prise de décision ainsi qu'à l'adaptation.

Comme l'illustrent ces travaux, plusieurs types d'applications peuvent utiliser la notion de contexte, de différentes manières, avec différentes implications sur la conception et le comportement de l'application. Quel que soit l'usage de l'information de contexte, le traitement de ces informations implique les différents aspects liés à leur gestion, depuis leur observation jusqu'à leur utilisation sur le logiciel. Par exemple, décider des modalités et des moyens de collecte et de transfert des données sur les scénarios GridStix et CC-Sem, définir un seuil pour l'irrigation dans un scénario de *smart* agriculture, choisir des méthodes d'analyse des données pour CC-Sem ou GridStix, etc., sont autant de décisions à prendre lors de l'élaboration de ces scénarios. L'identification de ces questions en amont une nécessité afin de garantir une meilleure prise en charge de l'information de contexte, et éviter ainsi des erreurs de conception qui pourraient coûter chères aux applications.

La dimension *finalité* considère ainsi les objectifs pour lesquels la notion de contexte est utilisée dans une application. Selon ces objectifs (par exemple, adaptation ou prise de décision), une application peut être plus ou moins sensible aux erreurs ou aux informations de contexte de mauvaise qualité, qui peuvent entraîner des décisions erronées, dont les conséquences peuvent être plus ou moins importantes selon l'application. Par exemple, si on considère une application de e-santé proposant d'adapter automatiquement les niveaux d'insuline en fonction du taux de glycémie d'un patient ou d'appeler les urgences en cas de chute, la fiabilité d'une telle application dépend directement de la qualité de l'information de contexte observée. Une information erronée ou de trop mauvaise qualité peut conduire à une mauvaise décision de la part de l'application, avec des conséquences importantes sur la santé du patient. La même chose peut être affirmée dans le scénario de *smart* agriculture, dans lequel des erreurs sur la lecture des données ou encore un problème affectant l'alimentation des capteurs (ce qui peut conduire à des informations de contexte manquantes) peuvent affecter l'approvisionnement en eau et par conséquent la production.

La question soulevée, d'un point de vue qualité, par la dimension *finalité* est essentiellement de savoir s'il convient ou non de surveiller (suivre, ou « *follow* » de la Figure 8) la qualité de contexte dans une application et comment le faire. La gestion de la qualité de contexte doit tenir compte des conséquences d'une information de contexte de mauvaise qualité et des conséquences de l'absence d'information à ce sujet. Ces conséquences doivent être prises en compte, tout comme les coûts associés à la gestion de la qualité de ces informations. Pour atteindre ces objectifs, une application va utiliser plusieurs mécanismes qui peuvent potentiellement être affectés par la qualité du contexte. L'inclusion de la QoC dans ces mécanismes représente un coût qui doit être pris en compte. Le coût de l'observation de la QoC sera-t-il plus ou moins important que le risque de ne pas l'observer ? Par exemple, dans le cas d'une application d'e-santé pour la détection de chutes, la prise en compte de la QoC dans le processus d'adaptation implique l'utilisation de différents algorithmes pour détecter et éliminer les mesures suspectes. Ces algorithmes demandent une certaine capacité de calcul et représentent un coût énergétique supplémentaire pour les dispositifs qui supporteront l'application, sans compter les coûts de conception et de développement eux-mêmes. Même si ces coûts peuvent

paraître importants par rapport aux coûts globaux de l'application, les risques et les conséquences possibles de la non prise en compte de la QoC dans ce cas justifient ces coûts. Le concepteur d'une telle application doit ainsi prendre en compte ces risques et les coûts associés à l'observation de la qualité de contexte par rapport à l'objectif de l'application. Toutefois, une question découle de cette analyse : comment peut-on mesurer ces risques et ces coûts ? Ce point reste une question ouverte.

4.2 Dimension Sujet

La deuxième dimension de la roadmap visible à la Figure 6 est la dimension *Sujet*. Celle-ci se concentre sur les informations qui pourraient être considérées comme étant des éléments de contexte et sur la manière d'identifier les éléments les plus pertinents. Il ne s'agit pas d'une question triviale, car la notion de contexte correspond à un concept large et souvent ambigu [171, 32, 96], se référant potentiellement à des éléments très différents, dont la pertinence dépend de l'usage que l'on en fera. La dimension *sujet* est directement liée à la première dimension, puisque l'objectif guide l'usage de ces informations, et par conséquent, la pertinence (ou non) d'une information donnée pour cette application.

La dimension *sujet* est directement liée à la définition même de la notion de contexte. Différentes définitions ont été proposées pour cette notion [32, 171, 96]. La plus acceptée reste celle proposée par Dey [74], définissant la notion de contexte comme toute information considérée comme pertinente pouvant être utilisée pour caractériser la situation d'une entité (une personne, un lieu ou un objet), lors d'une interaction entre un utilisateur et un système. Cette définition met en évidence à la fois une entité observée (par exemple, l'utilisateur) et une information qui est observée à propos de cette entité. L'entité délimite l'observation : le but est d'observer une entité donnée, mais en regardant cette entité, différents éléments peuvent être observés. Par exemple, lorsqu'on considère un utilisateur, il est possible d'observer son emplacement, son humeur, son niveau d'expertise, etc. Lorsqu'on considère un appareil, il est possible d'observer sa mémoire disponible, sa connexion réseau, etc.

Ainsi, l'entité correspond à l'objet de l'observation. Elle joue un rôle central dans la modélisation du contexte, comme le souligne [63, 35], puisque c'est précisément le contexte de cette entité qui est actuellement observé. Tout ce qui est observé est lié à cette entité [136]. On appelle donc un élément de contexte l'information observée (localisation, mémoire, etc.) sur une entité (sujet de la Figure 9). En observant de tels éléments de contexte, on obtient des valeurs correspondant à leur état actuel qui vont probablement évoluer dans le temps. Par exemple, en observant l'*élément de contexte* « localisation » d'une entité « utilisateur », il est possible d'obtenir des *valeurs* de la latitude et de la longitude correspondant à l'emplacement actuel de cet utilisateur. De même, dans le scénario de l'agriculture intelligente, l'*élément de contexte* « température » d'une parcelle de terrain, représentant l'*entité* observée, peut être estimé grâce à des *valeurs* multiples obtenues à partir d'un capteur de température. La Figure 9 représente un méta-modèle que nous avons proposé dans [136] résumant cette idée à travers laquelle une information de contexte est vue comme un ensemble de valeurs observées dynamiquement à propos d'un certain élément de contexte par rapport à un sujet (entité) donné.

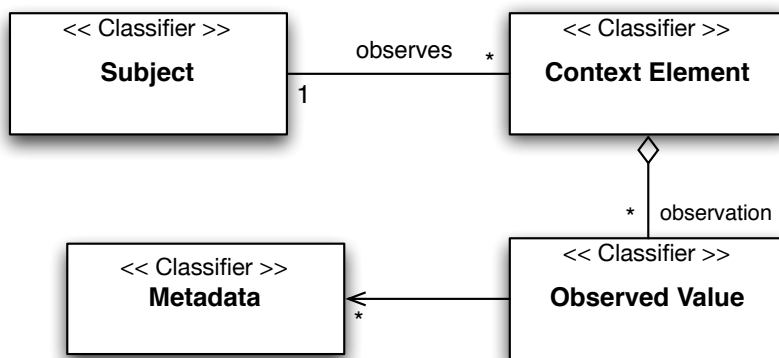


Figure 9. Meta-modèle pour l'information de contexte [136].

Identifier quelles informations doivent être considérées en tant qu'élément de contexte signifie identifier quelles entités et quels éléments de contexte doivent être considérés comme pertinents pour une certaine application. Cette pertinence dépend de la finalité identifiée, puisque les informations de contexte sont observées afin de satisfaire cette finalité. Leur pertinence dépend de leur contribution à cet objectif. Par exemple, dans le scénario GridStix, étant donné que l'objectif de ce système est la prévision des inondations, la principale entité à prendre en compte est la rivière elle-même et plus précisément son état. Ce dernier est déterminé en observant la hauteur du niveau de l'eau et le débit de la rivière à plusieurs endroits. Les nœuds GridStix peuvent aussi être considérés comme des entités puisqu'il est nécessaire d'observer leur niveau de batterie et leurs capacités de communication (connexions Bluetooth et Wi-Fi) pour permettre l'adaptation de l'infrastructure afin de réduire la consommation d'énergie.

Ce processus d'identification reste un défi important, comme l'indiquent [74, 96, 16], notamment parce que toute information pouvant être utilisée pour caractériser quelque chose (une entité ou le sujet selon le méta-modèle de la Figure 9) peut potentiellement être considérée comme un élément de contexte observable. Le principal aspect à prendre en compte reste sa pertinence, si cette information peut être pertinente pour la caractérisation d'une entité, compte tenu de l'objectif du système. Par exemple, l'information relative au niveau de la batterie peut être considérée comme pertinente dans le scénario GridStix car ce scénario suppose l'adaptation du comportement des nœuds en fonction de l'état de la batterie. Ces mêmes informations peuvent être ignorées dans le scénario CC-Sem, car dans ce scénario on peut supposer une alimentation électrique continue. Selon Bauer & Dey [16], puisque le contexte est l'élément clé qui définit la fonctionnalité d'un système sensible au contexte et façonne son comportement, le choix des informations de contexte à inclure dans ces systèmes devient une tâche importante dans leur conception.

Lorsqu'on considère la dimension *sujet*, la littérature illustre assez bien la variabilité de la notion de contexte. Par exemple, les premiers travaux en Informatique Ubiquitaire considéraient notamment la localisation et le dispositif de l'utilisateur en tant qu'éléments de contexte [225, 36], alors que des travaux plus récents considèrent même l'expression des situations complexes sur le comportement de l'utilisateur [256,66]. Des multiples visions sur ce qui peut être considéré comme un élément de contexte sont ainsi proposées, en fonction aussi des multiples définitions qu'on peut trouver pour la notion de contexte. Schilit & Theimer [225] définit le contexte comme la localisation, l'identité des personnes et des objets à proximité, ainsi que les changements survenus sur ces objets. À travers cette définition, ces auteurs se concentrent sur trois questions : où, qui est aux alentours et quelles sont les ressources environnantes. Pour Moran & Dourish [167], l'information de contexte fait référence à la situation physique et sociale dans laquelle les dispositifs informatiques sont intégrés. Cette définition se concentre sur les dispositifs en délimitant cette notion à leur perception logicielle, sans détailler les éléments possibles, ce qui lui garantit une certaine généralité. Cette vision « logicielle » est partagée par Dey [74], dont la définition se concentre sur l'interaction entre un utilisateur et un système donné.

Il en découle de ces différentes définitions (et perceptions de ce qui peut être la notion de contexte) une multitude d'informations pouvant être considérées comme de l'information de contexte. Par exemple, dans [225, 59, 36], la localisation est le principal élément pris en considération, tandis que dans [87,92], les informations sur l'environnement d'exécution, telles que la mémoire disponible ou l'état de la connexion réseau, sont considérées. Dans [200, 66], ce sont les informations sur le dispositif d'exécution et sur les dispositifs environnants qui sont considérées et utilisées afin de mieux identifier la situation de l'utilisateur, et d'ainsi déployer l'application sur ces dispositifs. Dans [206], les données obtenues à partir d'un accéléromètre d'un *smartphone* sont utilisées pour la détection de l'activité de l'utilisateur, tandis que dans [199], la localisation est estimée à l'aide des données GPS et de la GSM. Ces informations, ainsi que celles sur la connectivité du dispositif, sont utilisées pour adapter la livraison de contenu sur une application de dossier médical.

Lors de l'identification des informations de contexte pertinentes, il convient également d'identifier de possibles indicateurs de qualité qui peuvent y être associés. Souvent, la qualité de contexte est constituée de plusieurs éléments tels que la précision, l'actualité, la fraîcheur ou la probabilité d'exactitude [158,21]. L'identification des informations de contexte qui seront observées permet aux développeurs d'identifier également les critères de qualité qu'il sera pertinent d'y associer pour atteindre la finalité de l'application. Par exemple, si on considère l'information de localisation, différents indicateurs de qualité peuvent être pris en compte, tels que le taux d'erreur ou la précision estimée, la fraîcheur (qui peut être obtenue à partir d'un horodatage) ou même le nombre de satellites disponibles, lorsqu'il s'agit de données GPS. Plusieurs critères de qualité sont ainsi possibles, comme l'illustre [158]. Ces auteurs ont identifié et comparé différents indicateurs de QoC proposés dans la littérature, en mettant en évidence la variation des termes et de la signification de ces critères. Tout comme l'information de contexte elle-même, la pertinence d'un indicateur donné dépend souvent de l'usage qui en sera fait. Là encore, la finalité d'un système impacte fortement la pertinence d'une information de contexte et d'un critère de QoC. Les deux sont observés en tenant compte de cet objectif, et leur contribution à sa satisfaction détermine leur pertinence. Par exemple, si l'on considère les scénarios CC-Sem et de stress hydrique présentés précédemment, les deux peuvent considérer la précision comme un indicateur de QoC, mais leurs besoins ne seront pas les mêmes, car les erreurs seront sûrement mieux tolérées sur le premier scénario que sur le deuxième.

Il existe un certain consensus dans la littérature (*e.g.* [16, 74, 96, 158]) sur la difficulté de définir quels éléments de contexte ou quels indicateurs de QoC peuvent être considérés comme pertinents pour une application donnée. Cette question du choix des éléments à prendre en considération demeure une question ouverte. Même si les travaux cités n'abordent pas directement le problème du choix des informations de contexte à observer et des indicateurs de qualité à prendre en compte, certains parmi eux [92, 87, 108, 51, 158] proposent des solutions intéressantes pour intégrer ces éléments pendant les phases de conception. Par exemple, Geihls *et al.* [92] et Chabridon *et al.* [51] proposent une approche MDE (*Model Driven Engineering*) pour le développement d'applications sensibles au contexte. Les propriétés correspondant aux éléments de contexte à observer (pour la première) ou aux indicateurs de QoC (pour la seconde) sont modélisées à l'aide d'UML et d'autres représentations haut niveau au tout début du processus de conception. En se concentrant sur cette modélisation, ces approches aident les concepteurs à mieux identifier les éléments nécessaires à leurs applications. D'une part, en libérant les concepteurs des détails de mise en œuvre au début du projet, ceux-ci peuvent se concentrer sur les concepts et ensuite considérer plus facilement les informations nécessaires à leurs applications. D'autre part, en produisant du code source à partir de ces spécifications de modèle, de telles approches aident les concepteurs dans la tâche de développement elle-même.

La littérature révèle également d'autres questions qui découlent de cette identification des informations de contexte pertinentes. La première concerne la relation entre les éléments observés. Les éléments de contexte ne sont pas nécessairement indépendants, et leur relation peut également être pertinente. C'est notamment le cas des applications coopératives, dans lesquelles les membres

d'un groupe, les tâches et les objets peuvent être liés de différentes manières. Par exemple, le fait de savoir qu'une activité est liée à un groupe précis peut être aussi important que de connaître le groupe lui-même [137]. Le modèle présenté dans le chapitre III.1 explore justement ces relations afin de mieux caractériser la pertinence d'une information pour un utilisateur. Dans [90], ces relations sont explorées à travers des règles, permettant de raisonner sur les droits d'accès aux ressources partagées en fonction du contexte de l'utilisateur. Une autre question est la granularité de l'information observée. Certains éléments de contexte peuvent être décomposés en éléments de niveau inférieur ou regroupés pour former des éléments de niveau supérieur. La gestion de différents niveaux d'abstraction peut être requise par des applications complexes. Par exemple, dans [66], les auteurs proposent de regrouper les informations de bas niveau afin de décrire les situations complexes d'un utilisateur. De même, dans [194], les auteurs proposent une architecture modulable qui permet de composer de nouvelles informations de contexte à partir des données de granularité inférieure. Dans les deux cas, cette composition peut être assimilée à une interprétation des données brutes pour produire de nouvelles informations de contexte.

4.3 Dimension Modèle

Il ne suffit pas d'identifier les informations de contexte pertinentes, il faut également réfléchir à la manière de représenter ces informations. La dimension *Modèle* se concentre précisément sur la question de la modélisation de l'information de contexte. L'objectif est de déterminer la représentation la plus appropriée pour ces informations sur une application donnée, en fonction de sa finalité. Les informations de contexte doivent en effet être représentées au sein d'une application de telle sorte qu'il devienne pratique et possible pour cette application de les explorer et d'atteindre ainsi son objectif. Un modèle de contexte permet la définition des processus traitant l'information de manière indépendante des techniques d'acquisition utilisées. Un modèle inapproprié peut compromettre, ou du moins rendre plus complexe, la mise en œuvre d'une application donnée. Par exemple, dans le scénario GridStix, l'adoption d'un modèle de contexte approprié (par exemple, un modèle orienté objets) permettrait, d'une part, de définir des règles d'adaptation (notamment pour activer ou désactiver un nœud) indépendamment des capteurs ou des API utilisés pour acquérir ces données. D'autre part, un modèle inapproprié (par exemple un modèle trop complexe) peut avoir un impact négatif sur les performances du système, car l'analyse des données disponible par des modèles stochastiques peut exiger un traitement supplémentaire sur ces données.

La représentation des informations de contexte est une question difficile en raison de la nature de ces informations. Tout d'abord, les informations de contexte peuvent être *hétérogènes*. Étant donné la multitude d'éléments de contexte qui peuvent être observés, les informations obtenues peuvent varier d'informations numériques, comme les coordonnées GPS ou un pourcentage (par exemple, la charge du processeur), à des valeurs symboliques (par exemple, le rôle d'un utilisateur dans un groupe). Une telle hétérogénéité peut être observée aussi bien sur le contenu que et la structure de ces informations. Par exemple, dans le scénario GridStix, de multiples éléments peuvent être pris en compte, tels que le débit et le niveau des eaux de la rivière (l'entité observée), le niveau de la batterie et la connectivité Bluetooth des nœuds. La plupart de ces éléments prennent en compte des valeurs numériques, suivies d'un horodatage (métadonnées décrivant l'observation), sauf l'état Bluetooth, qui peut être représenté par une valeur booléenne (activé ou désactivé). Cependant, dans d'autres scénarios, des éléments de contexte ayant une structure plus complexe peuvent être observés. Par exemple, lorsque l'on considère les informations de localisation, plusieurs représentations sont possibles, telles que les coordonnées GPS ou l'adresse postale. Les deux représentations sont composées de valeurs multiples (au moins la latitude et la longitude pour la première, le nom et le numéro de la rue, la localité, le code postal, le pays, etc. pour la seconde). C'est le modèle de contexte qui organise et structure les données obtenues à partir de capteurs, les transformant en informations de contexte pouvant être explorées par l'application.

Par ailleurs, les informations de contexte sont naturellement *dynamiques*, variant selon les observations. Par exemple, dans le scénario de *smart* agriculture, les valeurs observées pour les niveaux d'humidité peuvent varier entre les observations en fonction de la consommation des plantes et des conditions météorologiques. Cette dynamique doit être soutenue par le modèle de contexte, qui doit conserver les valeurs associées aux différents éléments de contexte et supporter leur évolution dans le temps. En effet, on peut considérer que, par définition, le contexte consiste à caractériser la situation d'une entité qui est (ou peut l'être) en constante évolution.

Enfin, il ne faut jamais oublier que les informations de contexte sont également *incertaines* et souvent incomplètes ou présentant des erreurs, des imprécisions et des données manquantes [256,52,51,158], principalement en raison de problèmes lors de l'acquisition des données (problèmes de connexion, interférences, etc.). Par exemple, dans le scénario CC-Sem, les conditions météorologiques et notamment l'exposition des capteurs de température au soleil peuvent altérer sensiblement la qualité des mesures. De même, dans le scénario GridStix, les conditions de la rivière (par exemple, un débit important en période de crue) peuvent endommager certains nœuds GridStix, entraînant un manque de données à propos de certains segments de la rivière. Cette incertitude représente un problème important, puisque ces données influencent le comportement d'une application sensible au contexte, ce qui fait de la qualité de contexte une question particulièrement sensible.

Il est à souligner que ces trois aspects (hétérogénéité, dynamisme et incertitude) caractérisent profondément l'information de contexte. La gestion de ces aspects, notamment au niveau du modèle de contexte, est un facteur clé pour exploiter avec succès cette information dans une application. Plus que dans les applications traditionnelles, la gestion des données contextuelles implique de traiter ces aspects en priorité. Ces aspects ne peuvent pas être dissociés de la gestion de contexte.

Prendre en compte la QoC dans la dimension *Modèle* implique de se demander comment représenter les informations de qualité et comment elles seront associées au contexte observé. Plusieurs travaux de recherche ont été menés sur les modèles de contexte [24,32,33,171], et de nombreuses propositions ont déjà examiné la question de la qualité de contexte [52,51,158,108], souvent par le biais de métadonnées représentant des indicateurs de QoC. Comme le résume [24], une bonne approche de modélisation du contexte doit inclure la modélisation de la qualité des informations de contexte pour soutenir le raisonnement sur le contexte.

De manière générale, la modélisation de l'information de contexte a été largement traitée dans la littérature, où de nombreux modèles de contexte ont été proposés [171,24, 32]. Tous ces travaux mettent en évidence les défis liés à cette modélisation, et notamment la gestion de l'hétérogénéité, du dynamisme et de l'incertitude qui caractérisent la notion de contexte. Les préoccupations en matière de QoC affectent fortement la modélisation de cette notion, puisque ces modèles doivent inclure des informations de qualité nécessaire au traitement de la QoC dans l'application. Comme le suggèrent Chalmers *et al.* [52], ou Hoyos *et al.* [108], les informations de qualité doivent faire partie du modèle de contexte et ne peuvent pas en être dissociées dans une vision holistique de la modélisation du contexte. En outre, la modélisation du contexte joue un rôle important dans l'extensibilité et l'évolution de l'application. Les modèles de contexte contribuent à isoler le comportement de l'application des technologies utilisées pour l'acquisition de ces informations. Ce faisant, ces modèles contribuent à la possibilité d'évolution, permettant à de nouvelles informations de contexte d'être facilement prises en compte dans l'application, sans exiger de gros efforts de réécriture du code de la part des développeurs.

Plusieurs approches de modélisation du contexte existent, allant de simples ensembles « clé-valeur » jusqu'aux ontologies complexes [171,24,50, 219,92], en passant par des modèles structurés (basés sur XML ou RDF, par exemple [150]) et des modèles orientés objets (comme celui présenté dans le chapitre III.1). Ces multiples paradigmes de modélisation impliquent différents degrés de complexité, tant au niveau de la mise en œuvre que de l'exécution. Ils offrent également différentes capacités de raisonnement, qui peuvent inclure des traitements *ad hoc*, des techniques statistiques et d'analyse de

données ou un raisonnement complexe basé sur des règles. Les modèles les plus simples, tels que les modèles « clé-valeur », seront faciles à mettre en œuvre mais n’offriront pas de mécanisme de raisonnement particulier. Au contraire, les modèles basés sur des ontologies seront plus complexes à mettre en œuvre, mais ils permettront des mécanismes de raisonnement tout aussi complexes. Par exemple, dans [219], un modèle basé sur les ontologies est utilisé pour représenter les informations de contexte (temps, lieu, disponibilité, etc.) des entités liées à un processus métier (acteurs, ressources, etc.) sont pris en considération. Dans [262], différents paradigmes de représentation sont combinés. Tout d’abord, une ontologie de contexte, illustrée à la Figure 10a, permet de décrire des concepts significatifs liés aux informations de contexte, y compris les entités (l’utilisateur, un objet, etc.), les éléments de contexte associés (e.g., la localisation) et leur représentation. Ensuite, un méta-modèle orienté objets, reprenant les mêmes concepts, est utilisé principalement à des fins de mise en œuvre (voir Figure 10b). Enfin, un schéma XML représentant ces concepts est utilisé pour l’échange de ces informations entre différents nœuds de calcul.

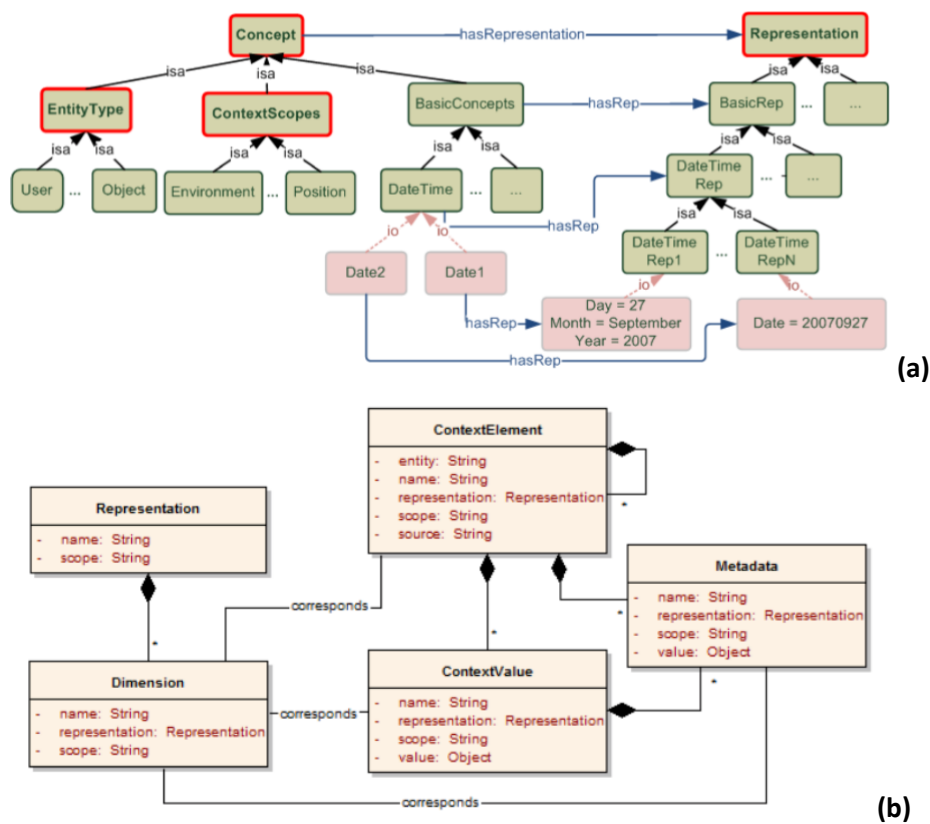


Figure 10. Ontologie (a) et méta-modèle orienté objets utilisés pour la modélisation de l'information de contexte [262].

La modélisation est également une préoccupation importante lorsqu’il s’agit de la QoC. Dans [52], un modèle de contexte est proposé en tenant compte de l’incertitude des informations de contexte. Cela se fait notamment par la définition de relations permettant de comparer les informations de contexte dans l’incertitude. Hoyos *et al.* [108], quant à eux, analysent les effets de la qualité sur un modèle de contexte. Ils envisagent une approche MDE pour la modélisation du contexte, en proposant un DSL (*Domain Specific Language*) pour la création des modèles de contexte. Grâce à ce langage, il est possible de décrire différents éléments de contexte et ses sources de données, ainsi que des situations plus complexes combinant les différents éléments de contexte. De même, Chabridon *et al.* [51] et Marie *et al.* [158] ont également envisagé une approche MDE, en se concentrant particulièrement sur la modélisation et la prise en charge des indicateurs de QoC. Dans [158], ces auteurs ont proposé un méta-modèle de QoC dans lequel les indicateurs de qualité sont associés à des informations de

contexte. Chaque indicateur a un ensemble de valeurs associées et est défini par un critère de QoC contenant un ensemble de métriques définies, permettant ainsi une définition complète de chaque indicateur, allant de la définition du concept à ses métriques.

4.4 Dimension Acquisition

Comme mentionné précédemment, les modèles de contexte jouent un rôle important en isolant le comportement de l'application de la technologie utilisée pour acquérir des informations de contexte. Ce faisant, les modèles de contexte contribuent à masquer l'hétérogénéité des dispositifs d'acquisition. En effet, il est impossible de considérer l'information de contexte sans envisager son acquisition, ou celle des informations sur la qualité du contexte. Les informations de contexte doivent être acquises en observant l'environnement autour d'une entité donnée. La dimension *Acquisition* met en évidence les défis liés à l'acquisition d'informations de contexte à partir de l'environnement, ce qui implique de prendre en compte les dispositifs de capture, le processus d'observation et la gestion de ces sources d'information. La dimension *Acquisition* considère donc cette question à travers trois points de vue (ou sous-dimensions) différents : *dispositif*, *observation* et *gestion* (cf. Figure 6).

Afin de capturer correctement les informations de contexte de l'environnement, il faut donc observer cet environnement à l'aide d'un dispositif d'acquisition approprié. De nombreux dispositifs peuvent être utilisés pour acquérir un seul élément de contexte. L'exemple le plus courant est peut-être celui des informations de localisation, qui peuvent être observées à l'aide de différents dispositifs et méthodes (GPS, estimation basée sur le GSM, triangulation Wi-Fi, etc.), mais on peut en dire autant d'autres éléments de contexte. Par exemple, la température peut être observée à l'aide de modèles de capteurs différents. Tous ces capteurs n'offrent pas le même support selon les langages de programmation, les bibliothèques et les plateformes utilisées. À titre d'illustration, le site *HomeAutomation.org*⁴ compare 5 modèles distincts de capteurs de température pour Arduino, avec différents codes de programmation en C pour accéder à chacun d'entre eux. Le code et le processus nécessaires à la saisie des données issues de ces capteurs ne sont pas nécessairement les mêmes selon le modèle choisi. Cette hétérogénéité rend le processus d'acquisition plus complexe et peut sérieusement compromettre l'interopérabilité entre les dispositifs d'acquisition.

L'hétérogénéité et l'interopérabilité de ces dispositifs se présentent donc comme d'importants défis pour l'acquisition de l'information de contexte, car la nature des dispositifs utilisés peut être très variée. Cette hétérogénéité rend plus complexe la prise en charge de différents éléments de contexte dans une application donnée, ainsi que l'évolution des applications existantes (par exemple, l'observation de nouveaux éléments de contexte ou la modification du dispositif d'acquisition), car le changement du dispositif d'observation peut affecter le code source ou la conception de l'application. Cela est particulièrement vrai dans les scénarios de déploiement à grande échelle, tels le projet CC-Sem ou le scénario de *smart* agriculture. Dans ces cas, où un grand nombre de capteurs doivent être déployés, il est particulièrement important de prendre en charge plusieurs modèles de capteurs (par exemple, les capteurs de température sur le scénario CC-Sem) et pouvoir remplacer facilement un capteur par un autre similaire (par exemple, un capteur d'humidité du sol par un autre capteur d'humidité). On comprend mieux l'importance d'isoler l'application et son comportement de la technologie exacte utilisée pour l'acquisition d'informations de contexte, comme le soulignent [24, 74], notamment à travers une modélisation appropriée de cette information.

Plusieurs travaux de recherche se sont penchés sur cette question, proposant des mécanismes permettant d'isoler les applications de l'hétérogénéité de l'environnement et par conséquent de permettre une meilleure interopérabilité entre les différents dispositifs d'acquisition. Parmi ceux-ci, on peut citer le *Context Toolkit* proposé par [74]. Cette boîte à outils isole l'application elle-même de

⁴ <http://www.homautomation.org/2014/02/18/arduino-temperature-sensor-comparison/>

la technologie d'acquisition à travers différentes abstractions, et notamment la notion de *context widget*, qui encapsule l'accès au dispositif physique. L'application n'a à traiter que ces abstractions, aucune connaissance directe sur le dispositif physique n'est nécessaire. Cette connaissance est concentrée à l'intérieur de chaque *widget*, ce qui offre une interface standard pour l'application, améliorant au passage l'interopérabilité du point de vue de l'application. Une approche similaire est adoptée dans [194]. Ces auteurs proposent une architecture modulable dans laquelle un ensemble de *plugins* de contexte est chargé dynamiquement en fonction des besoins de l'application. Là encore, les applications ne sont pas confrontées directement à l'environnement physique, ne gardant le contact qu'avec leurs *plugins* de contexte, ce qui offre, à nouveau, une interface standard pour l'accès aux informations de contexte. Toute l'interaction avec l'environnement réel est centrée sur le plugin, ce qui réduit la complexité de l'application et augmente les possibilités de réutilisation du code.

Moins discutée dans la littérature mais toujours importante, l'acquisition d'une information de qualité de contexte est assujettie aux mêmes considérations. En effet, tous les capteurs n'offrent pas les mêmes informations ou les mêmes métriques nécessaires à la construction de certains indicateurs de qualité, et les concepteurs de logiciels doivent souvent envisager d'autres moyens pour calculer ou estimer ces indicateurs. Par exemple, si l'on considère les capteurs de température comme ceux mentionnés par *HomeAutomation.org*, ces capteurs peuvent facilement être influencés par des facteurs externes (e.g. l'exposition à la lumière du soleil ou à une source de froid ou de chaleur), ce qui peut entraîner des erreurs importantes dans les données acquises. Dans l'expérience que nous avons menée auprès des étudiants MIAGE, cinq capteurs de température du modèle BMP 280 ont été utilisés dans une même pièce d'environ 35 m². Une différence jusqu'à 2~3°C a pu être observée sur les données de température entre les capteurs. Malheureusement, les capteurs comme le BMP 280 n'offrent pas nativement de métadonnées ou d'informations de qualité permettant à une application de calculer automatiquement les indicateurs de QoC. Le calcul d'indicateurs de QoC exige une certaine précision, dans ce cas, un équipement supplémentaire ou des processus d'étalonnage. Cette question de savoir comment obtenir ou estimer les indicateurs de QoC demeure une question ouverte laissée à la charge du concepteur de l'application.

Or cette question va largement influencer le choix des dispositifs d'acquisition et de manière dont cette acquisition sera réalisée. La notion de QoC influence en effet l'analyse de chacune de sous-dimensions de l'acquisition (*dispositif*, *observation* et *gestion*, comme l'illustre la Figure 8). Tout d'abord, lorsque l'on considère les dispositifs nécessaires à l'acquisition d'informations de contexte, il est également important de se demander si ces dispositifs sont capables de prendre en charge l'acquisition d'indicateurs de QoC. Tout comme les informations de contexte, ces informations sont calculées à partir de celles collectées par les dispositifs d'acquisition. La possibilité d'obtenir un indicateur de QoC donné dépend ainsi de la capacité de ces dispositifs à fournir les informations nécessaires à son calcul. Par exemple, lorsque l'on considère les données GPS, le nombre de satellites peut être utilisé pour estimer la précision des données. Si cette information ne peut être obtenue pour une raison quelconque, ce critère ne sera pas disponible. Dans le scénario de *smart* agriculture, on peut envisager d'utiliser la précision comme un indicateur de QoC. Cependant, il est nécessaire d'examiner d'abord comment obtenir un tel indicateur, car la plupart des capteurs d'humidité du sol ne sont pas en mesure de le calculer par eux-mêmes. Une phase d'étalonnage peut alors être réalisée, fournissant ainsi une estimation approximative de la précision du capteur concerné. La fiabilité des données peut également être considérée comme un indicateur de qualité possible. Si l'on considère, par exemple, une application pour le travail en équipe qui informe en permanence les utilisateurs de l'avancement d'un projet, les informations sur l'avancement des tâches pourraient dépendre fortement des informations fournies par les utilisateurs eux-mêmes. Dans ce cas, il est difficile d'estimer la fiabilité des données et donc la fiabilité de ces informations.

L'utilisation d'un certain dispositif d'acquisition pour l'observation de l'environnement signifie aussi alimenter le modèle de contexte avec les valeurs observées. Le processus d'*observation* doit prendre en compte non seulement le dispositif utilisé à cet effet, mais aussi la fréquence d'observation, en

fonction de la dynamique attendue des informations observées. Par exemple, les informations de localisation exigeront une observation active afin de garantir une certaine précision, tandis que le rôle d'un utilisateur peut être acquis à la demande. Une fois observées, ces informations doivent également être tenues à jour afin de représenter le contexte actuel des entités observées. Par exemple, dans le scénario de *smart* agriculture, les informations obtenues à partir d'un capteur d'humidité du sol doivent être régulièrement mises à jour afin de suivre les changements du niveau de stress hydrique des cultures. Ces mises à jour doivent être suffisamment fréquentes pour satisfaire les besoins hydriques des plantes, mais des observations trop fréquentes seront probablement inefficaces, voire inutiles, car la situation hydrique d'une culture ne changera pas de manière significative sur une très courte période (par exemple, à une échelle de quelques secondes).

Les besoins concernant la qualité de contexte peuvent guider le choix d'un dispositif d'acquisition plutôt qu'un autre. Les coûts associés à ces dispositifs peuvent également avoir une influence non négligeable sur ce choix. Par exemple, dans une application de maison intelligente, on peut envisager d'utiliser des capteurs au sol pour détecter la chute d'un résident, car ces dispositifs peuvent offrir une meilleure précision pour la détection des chutes que des simples accéléromètres. Or les coûts associés à ces dispositifs ne sont pas les mêmes, même s'ils peuvent être justifiés en fonction des objectifs de l'application (par exemple, si l'application est conçue pour une maison médicalisée). Les développeurs de logiciels doivent être conscients de ces problèmes lorsqu'ils examinent les indicateurs de QoC envisagés et les dispositifs capables de les obtenir.

De manière générale, la prise en compte de la QoC va avoir un impact sur le processus d'observation et sur les politiques utilisées pour guider ce processus, et vice-versa. Par exemple, le fait de considérer si une information de contexte nécessite une observation très fréquente implique probablement que la fraîcheur est un indicateur de qualité pertinent pour celle-ci, et vice versa : maintenir des niveaux élevés de fraîcheur exige des observations très fréquentes. C'est souvent le cas des informations de localisation sur les applications de transport : lorsque l'on envisage un scénario dans lequel des véhicules se déplacent, la fraîcheur de l'information de localisation indiquera si ces informations peuvent encore être utilisées ou si de nouvelles mesures sont nécessaires. Les développeurs de ces applications doivent alors réfléchir à la manière la plus appropriée d'observer les indicateurs de qualité concernés pendant le processus d'observation et à la fréquence à laquelle celui-ci doit être effectué.

Malheureusement, le processus d'observation, et les considérations de QoC lui concernant, n'ont pas reçu la même attention dans la littérature que les autres dimensions considérées par la roadmap. L'une des raisons qui motivent cette situation pourrait être la dépendance entre ce processus et l'application elle-même. En effet, le processus d'observation dépend fortement des besoins de l'application, compte tenu des informations de contexte utilisées et des critères de QoC pris en considération. Néanmoins, la plupart des solutions de *middleware* pour la gestion de contexte proposées dans la littérature, telles que [87,194,92], offrent des mécanismes de type *push* (suivant le modèle *publish-subscribe*) et *pull* (correspondant à un modèle à la demande) pour la capture des données de contexte à partir de l'environnement. Ces mécanismes constituent la base de politiques d'observation réussies. Néanmoins, la définition de ces politiques est laissée sous la responsabilité des concepteurs de l'application, en fonction des besoins de celle-ci.

La dimension *acquisition* implique également la *gestion* de l'environnement. De par leur nature, les données de contexte sont en constante évolution. Selon Greenberg [96], la notion de contexte est une construction dynamique vue sur une période de temps, des épisodes d'utilisation, l'interaction sociale, les objectifs internes et les influences locales. Il ne s'agit pas de l'état d'un environnement prédéfini avec un ensemble de ressources fixe. Cette notion fait partie d'un processus d'interaction avec un environnement qui est en constante évolution et qui peut être composé d'un ensemble de ressources reconfigurables, migratoires, réparties et ceci à plusieurs échelles [63]. La dynamique est intrinsèque à la notion de contexte et doit être prise en compte correctement lorsqu'on l'envisage dans une application. Elle doit être prise en compte à travers le processus d'acquisition, mais aussi à travers la gestion de cet environnement. Celui-ci ne peut pas être supposé statique. Par définition, les

applications sensibles au contexte doivent considérer un environnement dynamique, dans lequel les conditions d'exécution peuvent varier, les utilisateurs et les dispositifs peuvent se déplacer, les ressources peuvent entrer ou disparaître à tout moment. À mesure que les ressources se déplacent ou changent leur état actuel, la composition de l'environnement et des ressources disponibles changent également, rendant nécessaire la capacité de découvrir et de gérer ces ressources environnantes.

L'environnement lui-même étant dynamique, la disponibilité d'un certain dispositif pour l'acquisition de contexte n'est pas garantie. Certains dispositifs peuvent disparaître (par exemple en étant éteints) et d'autres peuvent rejoindre l'environnement, permettant la capture du contexte d'une nouvelle entité ou des nouvelles informations sur une entité déjà en observation. La *gestion* de cet environnement dynamique représente ainsi un défi, compte tenu de l'évolution de l'environnement et de la disponibilité des dispositifs d'acquisition qu'il contient. Le scénario GridStix est un bon exemple de cette gestion, puisque les nœuds GridStix peuvent être activés ou désactivés, entrant et sortant du système, en fonction du niveau de charge de leurs batteries. Il est donc nécessaire de gérer les dispositifs d'acquisition sur cet environnement. Cette gestion de l'infrastructure d'acquisition est également influencée par les indicateurs de QoC. On peut considérer par exemple la désactivation d'un dispositif si la précision offerte par celui-ci est trop faible ou inversement, le réactiver afin d'augmenter la précision globale du système. Cet exemple est visible dans le scénario GridStix, dans lequel les nœuds GridStix contenant des capteurs de débit et le niveau de l'eau peuvent être désactivés afin d'économiser leur batterie et réactivés afin de garantir que chaque segment de la rivière dispose d'un nombre suffisant de capteurs. En somme, il est alors important de considérer non seulement la manière dont les informations de QoC sont gérées, mais également l'influence de cette information sur la gestion de l'environnement utilisé pour l'acquisition.

4.5 Dimension Interprétation

Les données collectées au cours du processus d'acquisition correspondent à des données brutes qui doivent être souvent agrégées ou interprétées afin d'être mieux exploitées par des applications sensibles au contexte. La dimension *Interprétation* se concentre sur cette question, en considérant les défis liés à l'interprétation de l'information de contexte sous ses différentes formes (règles d'interprétation, exploration du contexte, etc.). Elle examine comment transformer les données brutes en connaissances utiles pour une application donnée. Par exemple, dans le cadre du scénario de *smart* agriculture, un capteur d'humidité, comme celui utilisé dans le prototype illustré par la Figure 5b, peut être utilisé pour évaluer le niveau d'humidité. Les données brutes offertes par ce capteur correspondent en fait aux valeurs d'impédance observées sur la parcelle de sol autour du capteur. Ces données brutes sont comparées à des seuils prédéfinis afin de déduire le niveau d'humidité de la parcelle. L'objectif de cette dimension est ainsi d'étudier les mécanismes d'interprétation qui pourraient être utilisés par une certaine application. Différents mécanismes de raisonnement et d'agrégation peuvent être envisagés, du mécanisme de raisonnement *ad hoc* jusqu'aux systèmes complexes basés sur des règles [219,90]. Les *plugins* proposés par [194], mentionnés précédemment, représentent un mécanisme intéressant pour agréger des données de contexte de manière transparente : l'application peut traiter des informations de contexte agrégées de la même manière qu'elle traite des données brutes par le biais de ces *plugins*.

Il est néanmoins à souligner que les possibilités d'interprétation de l'information de contexte dépendent aussi des capacités offertes par le modèle de contexte. Ces mécanismes ne peuvent pas être dissociés de ce modèle. D'une part, le modèle de contexte délimite les possibilités d'interprétation, et d'autre part, les informations déduites par ces mécanismes d'interprétation pourront aussi alimenter le modèle de contexte, comme un mécanisme d'acquisition, en construisant ainsi des informations de haut niveau à partir de données brutes. Par exemple, les règles proposées dans [90] ne sont applicables qu'à l'aide d'un modèle de contexte basé sur une ontologie. D'autres mécanismes complexes, pas nécessairement basés sur des ontologies, peuvent également être cités.

Dans [66], un mécanisme de *workflow* est utilisé pour déduire des situations complexes à partir des données de contexte, tandis que dans [223], la programmation par contraintes est utilisée afin de contrôler l'adaptation de l'application en fonction des changements d'environnement.

En fonction du modèle de contexte proposé, les possibilités offertes par les mécanismes d'interprétation peuvent être immenses, notamment dans la prise en compte du caractère dynamique et incertain des informations de contexte. La dimension *Interprétation* va ainsi permettre une meilleure exploitation des données de QoC. Inversement la qualité des informations disponibles peut affecter les mécanismes d'interprétation, et influencer ainsi la fiabilité de l'application cible. Une illustration de cette influence est donnée par Vanrompay [253]. Dans ce travail, un ensemble de métriques évaluant la QoC est analysé. Vanrompay [253] propose d'utiliser ces métriques sur la prédiction du contexte, afin d'éviter que des informations de mauvaise qualité n'affectent le mécanisme de prédiction. Ces travaux démontrent l'importance de considérer comment la QoC influence les mécanismes de raisonnement et donc l'interprétation qu'on pourrait avoir de l'information de contexte.

En outre, une nouvelle tendance à l'interprétation peut être observée : l'extraction ou le *mining* d'informations de contexte. L'idée est d'appliquer les techniques de *data mining* sur les informations de contexte à différentes fins : pour découvrir des informations manquantes [206, 230] ; pour anticiper l'évolution du contexte [161], etc. Par exemple, Ramakrishnan [206] considère différentes méthodes statistiques, et notamment les réseaux bayésiens, pour analyser les données issues d'accéléromètres et de gyroscopes afin d'identifier l'activité d'un utilisateur (marche, course, etc.). Mayrhofer *et al.* [161] proposent d'utiliser des chaînes de Markov afin de déduire les informations de contexte à venir et d'anticiper ainsi la prochaine situation éventuelle d'un utilisateur. Toutes ces techniques peuvent contribuer à des applications sensibles au contexte en leur permettant d'adopter un comportement plus proactif, voire même prédictif, anticipant l'évolution de l'environnement.

A travers l'application de techniques d'analyse de données, comme celles citées précédemment, il est possible pouvoir explorer davantage les informations sur la QoC pour de meilleurs résultats. On peut, par exemple, envisager l'analyse de données historiques, aussi bien sur les données de contexte elles-mêmes que sur leurs indicateurs de qualité. Par exemple, la recherche d'*outliners* ou encore des signaux faibles sur les valeurs de contexte et de leurs indicateurs de qualité (*e.g.* l'exactitude, la précision, etc.) peut contribuer à prévoir de nouvelles tendances et à anticiper des actions. Les signaux faibles sont généralement considérés comme des valeurs anormales ou des « informations sur le changement potentiel d'un système vers une direction inconnue » [162]. L'analyse de ces signaux peut être particulièrement intéressante dans certains scénarios, car elles peuvent révéler des événements ou des phénomènes encore méconnus. Dans le scénario CC-Sem, l'analyse des signaux faibles peut contribuer à la prévision de la consommation d'énergie, en révélant de nouvelles tendances sur la consommation d'énergie (par exemple la présence d'un nouveau dispositif) ou en prévoyant les variations dues aux changements météorologiques.

On peut alors observer une importante interdépendance entre les trois dimensions *Modèle*, *Acquisition* et *Interprétation*. Celles-ci sont alors intrinsèquement liées et ne peuvent pas être dissociées, comme l'illustre la Figure 11. Tout d'abord, elles sont connectées parce que ces dimensions échangent des données, représentées par les flots de données de la Figure 11 : les mécanismes d'acquisition alimentent le modèle de contexte avec des données brutes, qui sont consommées par le mécanisme d'interprétation, et dont les résultats alimenteront à nouveau l'extension du modèle (c'est-à-dire les instances). Deuxièmement, ces trois dimensions ont un impact dans les choix réalisées sur les autres dimensions (comme l'illustre la flèche d'influence de la Figure 11) : les mécanismes d'acquisition auront une influence sur les mécanismes d'interprétation qui pourront être appliqués (par exemple, les données de gyroscope et d'accéléromètre peuvent être utilisées pour générer des informations sur les activités d'un utilisateur, grâce à des méthodes statistiques comme dans [206]). Inversement, le mécanisme d'interprétation aura une influence sur le choix des méthodes d'acquisition (par exemple, les méthodes de triangulation peuvent être utilisées pour déduire une

localisation à partir de données GSM au lieu de GPS, comme dans [199]). De même, les décisions relatives à l'interprétation et à l'acquisition peuvent inspirer le modèle choisi, tant au niveau de son intention (c'est-à-dire de son structure) que de son extension (c'est-à-dire les instances). Réciproquement, le choix du modèle guidera et limitera les possibilités d'interprétation (par exemple, les mécanismes basés sur des règles, tels que [90], seront difficiles à appliquer sans un modèle basé sur une ontologie, et les méthodes statistiques, telles que les réseaux bayésiens appliqués sur [206], nécessitent souvent une représentation numérique des données).

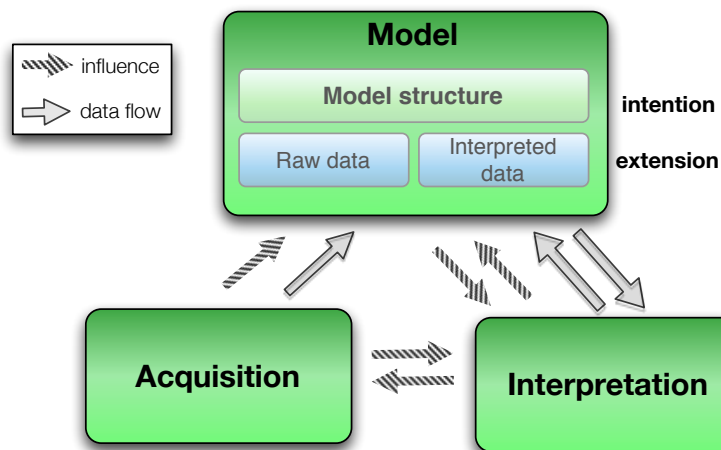


Figure 11. Influence et flot de données entre les dimensions modèle, acquisition et interprétation [130].

4.6 Dimension distribution

La dimension *Distribution* explore les questions liées à la transmission d'informations de contexte entre plusieurs nœuds dans un environnement réparti. En effet, avec la distribution des applications, la distribution d'informations de contexte sur plusieurs nœuds devient aussi une nécessité. Les applications sensibles au contexte se comportent bien souvent comme des applications réparties, dans lesquelles de multiples nœuds communiquent et échangent des informations sur leur état actuel. La distribution d'informations de contexte vers d'autres nœuds devient alors nécessaire, principalement lorsqu'il s'agit d'adapter le comportement des applications aux ressources disponibles dans l'environnement, comme c'est le cas dans Da *et al.* [66], lequel utilise les informations de contexte sur les nœuds voisins pour adapter le déploiement de l'application.

Comme on peut observer dans le cas de [66], dans certains cas, les informations de contexte doivent être transmises à partir du nœud dans lequel elles sont observées vers un autre nœud, où elles seront traitées, interprétées ou stockées. Par exemple, dans le scénario GridStix, les informations de contexte concernant l'état de la rivière et des nœuds sont transmises à un serveur distant pour traitement et la production d'alertes aux inondations. De même, dans le scénario CC-Sem, les informations de contexte sont transférées de leur lieu de production (*i.e.* dispositif d'acquisition) vers une infrastructure de calcul appropriée, qui permettra l'analyse des données sur la consommation d'énergie à large échelle. Comme le souligne [21], différentes approches pour la distribution des informations de contexte sont possibles, entre approches centralisées, pair-à-pair, ou encore hybrides ou hiérarchiques. Par exemple, Devlic & Klintskog [72] proposent une solution hiérarchique basée sur un protocole de communication SIP pour le partage des informations de contexte entre les membres d'une communauté.

Cependant, l'apparition de l'IoT a apporté avec elle une nouvelle préoccupation pour la distribution des informations de contexte : l'évolutivité. En effet, dans un environnement IoT, le nombre de dispositifs peut croître de manière exponentielle [193], ce qui exige des mécanismes de distribution

capables de s'y adapter. De nombreuses approches soulignées par Bellavista *et al.* [21] sont incapables de gérer potentiellement des centaines de nœuds, comme il est envisagé dans les scénarios type IoT. Les architectures hybrides, utilisant des approches hiérarchiques, telles que [217], ou celles combinant l'IoT aux infrastructures *cloud*, telles que [163,260], sont de plus en plus utilisées.

Plusieurs autres défis découlent de cette distribution, notamment la stabilité des informations de contexte (combien de temps une information donnée reste-t-elle valide et utile après avoir été transférée depuis un autre nœud ?) et la cohérence des données collectées, puisque des données contradictoires peuvent être rapportées depuis plusieurs nœuds observant une même entité (par exemple, plusieurs capteurs de température observant différentes valeurs pour une même pièce en fonction d'influences externes, telles que le soleil ou le chauffage). En effet, lorsque l'on considère la distribution des informations de contexte, il est important de se demander si cette transmission pourra affecter la qualité des informations transmises. Selon Bellavista *et al.* [21], il est nécessaire de prendre en compte le processus de distribution et la qualité des données de contexte échangées afin de pouvoir assurer la satisfaction des utilisateurs. Si la distribution des données de contexte ne tient pas compte de la qualité de ces données, certaines décisions prises à partir de ces données pourraient être affectées, induites en erreur par des données rendues de mauvaise qualité par leur transfert. Par exemple, lorsqu'on considère une application telle que celle proposée par Da *et al.* [66], qui déploie ses composants sur des nœuds voisins en fonction des ressources disponibles, si les informations sur ces ressources sont expirées (ou ne sont plus valables), en raison de la latence du réseau, la décision sur le déploiement d'un composant peut entraîner la perte de performances et, par conséquent, l'insatisfaction des utilisateurs. Les concepteurs de ces applications doivent ainsi considérer comment réaliser le transfert des informations de contexte (et de QoC), mais également comment garantir que cette diffusion n'affectera pas la qualité de ces données.

Ce besoin de distribution de l'information de contexte est d'autant plus important dans le cadre de scénarios type IoT. Dans ces scénarios, les informations de contexte sont souvent transférées vers des serveurs distants ou des plateformes de *Cloud Computing* pour l'analyse des données. Dans ces cas, des indicateurs de qualité tels que la latence ou la perte de paquets peuvent affecter de manière significative la fiabilité des informations.

L'utilisation d'infrastructures de *Cloud Computing* amène aussi à considérer la question de la persistance de l'information de contexte et ses politiques de contrôle d'accès. Malheureusement, ces questions restent marginales dans la littérature par rapport à leur importance pour la vie privée de l'utilisateur. Une raison pouvant expliquer ce phénomène est le fait que les informations de contexte sont souvent supposées être consommées en « temps réel » : il s'agit du contexte d'une entité donnée (utilisateur, objet, service, etc.) à ce moment précis. Dans ce cas, le stockage et l'analyse historique ne sont pas nécessairement pris en compte, puisque considérés peu pertinents selon l'application. Or ceci est sur le point de changer avec l'adoption massive de solutions de stockage dans le *cloud* et l'application de techniques de *data mining* pour l'analyse des données de contexte, comme dans [206]. Les questions de sécurité et de contrôle d'accès à ces données risquent de devenir de plus en plus pertinentes dans les prochaines années. Aujourd'hui, le nombre de travaux traitant de ces questions n'est pas proportionnel à leur pertinence. Seuls quelques travaux se sont intéressés à ces questions. Par exemple, Devlic *et al.* [73] proposent des politiques d'accès basées sur la norme XACML, en utilisant une approche FOAF (*Friend Of A Friend*).

5 Synthèse des questions

La roadmap présentée dans ce chapitre analyse les différents défis de la gestion de contexte, organisés en six dimensions. Chaque dimension prend en compte un aspect précis de la gestion de contexte et ses préoccupations en matière de qualité.

Tant la roadmap pour l'ingénierie de contexte présentée à la Figure 6, que ses préoccupations en matière de qualité de contexte, illustrées à la Figure 8, offrent une vue multidimensionnelle de la gestion de contexte et des multiples aspects impliquant cette gestion. Le Tableau 1 résume l'ensemble de questions soulevées par cette roadmap et quelques exemples de travaux les traitant retrouvés dans la littérature.

Tableau 1. Les dimensions de gestion de contexte avec les principales questions et quelques exemples [130].

Dimension	Concepts Clés	Questions	Exemples
Finalité	Adaptation Qualité de Contexte (QoC)	Pourquoi utiliser l'information de contexte ? Quelle est la finalité pour l'usage de cette information dans l'application ? Comment cette information sera utilisée ? Pourquoi l'application devrait surveiller les indicateurs QoC ? La QoC est suffisamment pertinente pour l'application ?	49 50 59 66 87 200 225 256 137 145 197 219
Sujet	Information de Contexte Indicateur QoC	Quelle information est considérée comme contexte ? Comment l'identifier ? Quelle information est nécessaire à l'application ? Quels indicateurs de qualité peuvent être utilisés ?	16 17 36 199 206 225 158
Modèle	Modèle de Contexte Métriques QoC & modèles	Comment représenter l'information de contexte et de QoC ? Comment structurer ces données ? Comment prendre en charge l'hétérogénéité, la dynamique et l'incertitude dans ces représentations ?	24 92 137 150 223 262 51 52 108 158
Acquisition	Dispositifs d'observation Capteurs Plateforme d'acquisition Mesure de QoC	Comment acquérir les données de contexte et la QoC ? Quel dispositif d'acquisition peut être utilisé ? Comment et avec quelle fréquence ces données doivent être collectées ? Comment gérer l'environnement et ses dispositifs ? Comment tolérer l'hétérogénéité du matériel ? Les indicateurs QoC sont-ils supportés par ces dispositifs ?	74 66 87 194
Interprétation	Raisonnement <i>Context mining</i>	Comment interpréter les données de contexte et de QoC ? Comment produire des nouvelles données de contexte et de QoC à partir de données de bas niveau ? Comment appliquer des méthodes de raisonnement ou d'analyse de données aux données disponibles ? Comment prendre en compte la QoC lors de l'interprétation ?	52 66 90 161 206 223 230
Distribution	Distribution de l'information Passage à l'échelle	Comment transférer les données de contexte et de QoC entre les nœuds ? Est-ce que ce transfert peut affecter la QoC ? Comment garantir la fiabilité des données pendant le transfert ? Comment garantir la validité et la cohérence des données ? Comment gérer le passage à l'échelle lors que le nombre de nœuds augmente ? Comment mettre en place des politiques d'accès et de protection de la confidentialité ?	21 72 193 217

Il convient de noter que les dimensions proposées dans cette roadmap n'auront pas toutes la même influence sur tous les types d'applications. La pertinence de chaque dimension dépend de l'objectif de l'application et des informations de contexte considérées par celle-ci. Il est alors essentiel de considérer et de discuter chaque dimension, en considérant sa pertinence et l'impact de la qualité de contexte sur celle-ci.

L'objectif principal de cette roadmap reste celui d'aider des concepteurs non-experts à mieux comprendre la notion de contexte, ainsi que les questions et les défis nécessaires à sa prise en charge. Ces concepteurs sont confrontés à un concept très complexe, dont la compréhension et la gestion sont loin d'être simples, comme le montre les nombreux travaux de la littérature cités dans cet état de l'art. A travers ces questions (Tableau 1), cette roadmap essaie d'accompagner ces concepteurs dans leur montée en compétences autour de la notion de contexte et de son support.

6 Considérations finales

La notion de contexte a été largement explorée de diverses manières à travers différentes applications. Cette utilisation est susceptible de se développer dans un avenir proche, notamment grâce à la démocratisation de l'IoT et de ses technologies, lesquelles permettent une observation aisée de l'environnement physique à l'aide de dispositifs peu coûteux. Néanmoins, la notion de contexte demeure un concept obscur et ambigu, malgré les nombreuses études et définitions apportées par la littérature [24, 35, 49, 50, 63, 96, 97]. Quelle que soit la définition utilisée, la notion de contexte dénote un ensemble d'informations capables de caractériser un élément ou une action, apportant un complément nécessaire à la compréhension de ce dernier. La définition donnée par Dey [74], largement adoptée aujourd'hui, apparaît alors comme celle la plus pragmatique, puisqu'elle limite l'analyse au cadre de l'interaction entre un utilisateur et un système. C'est la prise en compte des éléments qui entourent cette interaction et qui lui donnent un certain sens qui se trouve au centre de l'analyse. Même si on se place clairement dans le cadre de cette définition ici, considérant la notion de contexte comme tout élément d'information permettant de caractériser un élément observé jugé pertinent pour le bon fonctionnement d'un système informatique, cette notion demeure inéluctablement large et difficile à maîtriser. La question de savoir quelles informations peuvent être considérées comme élément de contexte et quelles informations ne le sont pas, illustre bien cette question. Des informations telles que la mémoire disponible, le niveau de la batterie ou encore le rôle de l'utilisateur dans une organisation peuvent être considérées comme élément contexte par certains [194, 92, 219], ou comme simple paramètres par d'autres [111,223]. Certains auteurs, dont [49], ont tenté d'établir une distinction entre les données de contexte et celles de l'application. Pour ces auteurs, les données de contexte correspondent à un ensemble de paramètres, qui sont externes à l'application et qui influencent le comportement de l'application. Malgré les efforts pour clarifier cette distinction, les frontières restent souvent floues, tout comme la notion de contexte elle-même, qui demeure souvent méconnue par beaucoup de jeunes concepteurs de logiciels.

Cette même ambiguïté est aussi visible entre les applications sensibles au contexte et les applications dites « auto-adaptatives ». Les scénarios de *smart* intelligente et de GridStix illustrent bien ces ambiguïtés. Toutes les deux utilisent l'information de contexte pour adapter leur comportement, mais les auteurs de la seconde la considèrent comme une application auto-adaptative [223]. Selon Khan [123], les concepts de sensibilité au contexte et d'auto-adaptation sont souvent sources de confusion car les applications auto-adaptatives adaptent leur comportement souvent en fonction des stimuli issus des informations de contexte. Il est donc difficile de faire une distinction claire entre ces deux concepts. Les deux peuvent être considérés comme des systèmes adaptatifs qui, selon Colman *et al.* [61], visent à atteindre un certain objectif par la définition d'une forme de boucle dans laquelle l'environnement et/ou le système est lui-même surveillé, les informations recueillies sont analysées, une décision est prise par rapport aux changements nécessaires, et ces changements sont ensuite mis en œuvre par le système. Pour ces auteurs, la « conscience de soi » signifie que les changements

peuvent souvent être traités automatiquement par rapport aux systèmes conventionnels qui nécessitent une refonte, une mise en œuvre ou un redéploiement *offline*, ce qui est également vrai pour les systèmes sensibles au contexte, puisqu'ils adaptent, sans intervention humaine, leur comportement en fonction des changements observés dans l'information de contexte. Même si certains auteurs ont tenté d'établir une certaine distinction entre ces concepts [61,123], la question la plus importante n'est pas vraiment ces différences potentielles (si elles existent réellement), mais le support à l'information de contexte dans ces systèmes. Ces deux concepts particulièrement complexes se basent sur l'information de contexte, un type d'information très dynamique, hétérogène et, de surcroît, incertaine. La gestion de contexte soulève ainsi plusieurs défis qui doivent être pris en considération lors du développement d'un système observant cette notion.

La question principale est donc comment la gérer et exploiter l'information de contexte dans un système ? Comme le souligne [63], il est communément admis que le contexte concerne l'évolution d'un espace d'information structuré et partagé, et que cet espace est conçu pour servir un objectif particulier. Quelle que soit l'information considérée comme un contexte, elle dépend profondément du système en question et de ses objectifs. Quelle que soit cette information, il est nécessaire de la gérer de manière appropriée, afin d'en tirer toute sa potentialité. Ce support nécessite de comprendre les défis qu'implique l'usage de cette notion et les principales caractéristiques de l'information de contexte, telles que justement son hétérogénéité, son dynamisme et son incertitude.

Le principal objectif de la roadmap présentée dans ce chapitre est précisément de contribuer à cette compréhension. Celle-ci est particulièrement nécessaire dans ce document, puisque tout au long de ma carrière, j'ai pu décliner la notion de contexte sur différents systèmes, ce qui impliquait bien souvent de considérer certaines questions soulignées par cette roadmap. Sans cette compréhension sur la notion de contexte et son support, difficile de comprendre dans leur globalité les problématiques de recherche soulevées par chacune des contributions présentées dans ce document.

III Contributions

1 Contexte dans les collecticiels

1.1 Rappel de la problématique

Ce chapitre présente les contributions directement issues du travail de thèse (2002-2006), portant sur l'adaptation de l'information de conscience de groupe (« *group awareness* ») dans les collecticiels (« *groupware systems* »), notamment ceux supportant un usage mobile. Pour rappel, la notion de conscience de groupe désigne certaines informations à travers lesquelles les membres d'un groupe, alors qu'ils sont engagés dans leurs activités individuelles, capturent ce que les autres participants font (ou ne font pas) et ajustent leurs propres activités en conséquence [99, 226]. L'information de conscience de groupe peut être définie comme la connaissance qu'un utilisateur a à propos de son groupe, de ses collègues et de leurs activités, laquelle constitue un contexte pour ses propres activités individuelles. Ce contexte est utilisé pour garantir que les contributions individuelles soient pertinentes pour le groupe dans son ensemble, et pour évaluer les actions individuelles par rapport aux objectifs et à la progression du groupe [77]. Cette notion est indispensable pour les collecticiels, où elle contribue à la coordination dans ces logiciels dédiés au travail en équipe.

L'information de conscience de groupe joue un rôle très important dans la coordination du travail au sein des équipes travaillant en mode asynchrone ou géographiquement éloignées. La triste réalité de l'année 2020 a remis sous les feux des projecteurs ces logiciels et ce besoin de coordination pour les équipes travaillant de manière répartie. Beaucoup d'employés mis en télétravail à cause de la situation sanitaire (pendant le confinement et même après) ont eu l'impression de perdre le contact avec leurs collègues et leurs activités au cours de cette période. Le manque d'un support adéquat à l'information de conscience de groupe s'est fait alors cruellement sentir.

Avant même la crise sanitaire de cette année, ce besoin d'une information de conscience de groupe adaptée se faisait sentir, notamment auprès des équipes nomades. En effet, les nouvelles technologies ayant libérées les équipes, le travail en situation de mobilité est devenu une réalité (l'adage « *anytime, anywhere* » s'applique désormais au quotidien dans nos vies professionnelles). Par contre, le risque de perte de contact entre les membres d'une équipe s'est aussi accru avec cette nouvelle mobilité, augmentant ainsi l'importance d'un support à la conscience de groupe adéquat.

Lorsqu'elle est correctement observée, l'information de conscience de groupe peut être abondante et le risque de surcharge cognitive devient réel : si on présente à l'utilisateur l'ensemble d'informations disponibles, celui-ci risque d'être « noyé » par le surplus d'information, ce qui l'empêcherait d'assimiler les informations pertinentes. Selon Bouthier [30], d'une part les informations échangées et présentées peuvent être très nombreuses, particulièrement lorsque le groupe est formé d'un grand nombre de membres ou bien de nombreux artefacts sont manipulés. D'autre part les ressources mentales de l'utilisateur, comme sa mémoire ou son attention, sont limitées. Il doit pourtant interpréter et intégrer ces informations pour construire sa conscience de groupe. La surcharge survient lorsque l'utilisateur se retrouve en face d'un trop grand nombre d'informations à traiter. Il subit alors une situation de stress qui peut l'amener à rejeter en bloc l'ensemble des informations. Ce stress peut poser des difficultés à un membre. Ces difficultés, bien que subies par un membre de groupe, peuvent être à l'origine de rupture dans la réalisation d'activités et dans la circulation d'informations qui pénalisent le travail des autres membres.

L'utilisateur en situation de mobilité, avec potentiellement un environnement contraint (e.g. terminaux aux capacités limitées, connectivité aussi limitée, environnement inadéquat, bruyant, etc.), le risque de surcharge cognitive est encore plus élevé. Il faut donc impérativement que cette information de conscience de groupe soit la plus adaptée possible afin de réduire le risque de

surcharge cognitive. En d'autres termes, l'utilisateur ne doit pas passer plus de temps à prendre conscience de ce qui se passe dans son équipe qu'à réaliser les tâches qui lui incombent.

Il faut donc réduire le volume total d'information proposée à l'utilisateur, afin de ne lui proposer que celles les plus pertinentes, lui offrir donc la « bonne information » au « bon moment ». Cet utilisateur étant nomade et potentiellement équipé d'un terminal d'accès aux capacités restreintes, il devient important de prendre en compte le contexte de cet utilisateur au moment de l'accès aux informations de conscience de groupe. Les informations de conscience de groupe devraient alors être adaptées au contexte d'accès, mais également aux intérêts de l'utilisateur vis-à-vis de ce type d'information. Or la pertinence de l'information de conscience de groupe peut varier, pour un même utilisateur, en fonction du contexte dans lequel il accède à ces informations. En effet, les utilisateurs dans un collectif ont tendance à développer des habitudes de travail (*e.g.* consulter ses messages à certains endroits ou à partir de certains dispositifs, utiliser certains terminaux en particulier pour la réalisation de certaines tâches, etc.) qui peuvent avoir une influence sur leurs préférences par rapport aux informations de conscience de groupe.

Deux sous-problèmes découlent alors de la problématique qui était l'adaptation de l'information de conscience de groupe au contexte de l'utilisateur : tout d'abord, la question de la pertinence de l'information de conscience de groupe, puis la question de la représentation de la notion de contexte au sein d'un collectif. D'une part, il se pose la question de l'expression des préférences de l'utilisateur et du processus d'adaptation lui-même, qui devrait tenir compte de ces préférences variables. D'autre part, les collectifs étant des applications collaboratives, il était important de considérer l'utilisateur non seulement en tant qu'individu, mais également en tant que membre d'une équipe. La notion de contexte physique, communément considérée à l'époque (2002-2006) par les applications sensibles au contexte, se montrait alors trop étroite pour les collectifs. Il fallait élargir la prise en compte de ce concept à travers une modélisation appropriée à ce type d'outil collaboratif.

En somme, deux questions en particulier ont été traitées par les travaux décrits ci-dessous : l'adaptation de l'information de conscience de groupe, mais également la modélisation de la notion de contexte afin de tenir compte des particularités d'un collectif.

1.2 Contributions & Impact

Deux contributions se dégagent de ces travaux. Tout d'abord, un *modèle de contexte* tenant compte des *aspects collaboratifs* liés à l'utilisateur. Celui-ci n'est plus vu comme un individu isolé, mais comme quelqu'un participant à une ou plusieurs équipes de travail, ce qui a conduit à la prise en compte d'informations comme le(s) rôle(s) joué(s) dans l'équipe ou encore les objets partagés par celle-ci en tant qu'éléments de contexte. Puis, un *mécanisme de filtrage* pour l'information de conscience de groupe (*group awareness*) tenant compte à la fois des préférences utilisateur, exprimées à travers un ensemble de profils, et de son contexte d'utilisation.

Le modèle de contexte proposé en 2004 [137] se présentait comme un modèle à deux niveaux (Figure 12 et Figure 13) : d'abord, nous avons la notion de « description de contexte » qui décrit le contexte observé d'une entité comme un ensemble d'éléments de contexte observables (Figure 12) ; puis, nous avons la définition de ces éléments et des relations pouvant les relier (Figure 13). Une particularité de ce modèle repose sur la notion d'entité elle-même (indiquée par « cible » à la Figure 12), qui est vue dans le modèle comme un élément de contexte. Par exemple, lorsqu'on observe le contexte d'un utilisateur, on peut observer le terminal (« dispositif ») qu'il utilise. L'entité observée dans ce cas serait l'utilisateur (représenté par la classe « membre » dans le modèle) et un élément de contexte observé serait alors son dispositif, mais on peut également observer le contexte d'exécution de ce même dispositif, avec par exemple les applications qui s'y exécutent. Il devient ainsi l'entité observée avec d'autres éléments de contexte associés.

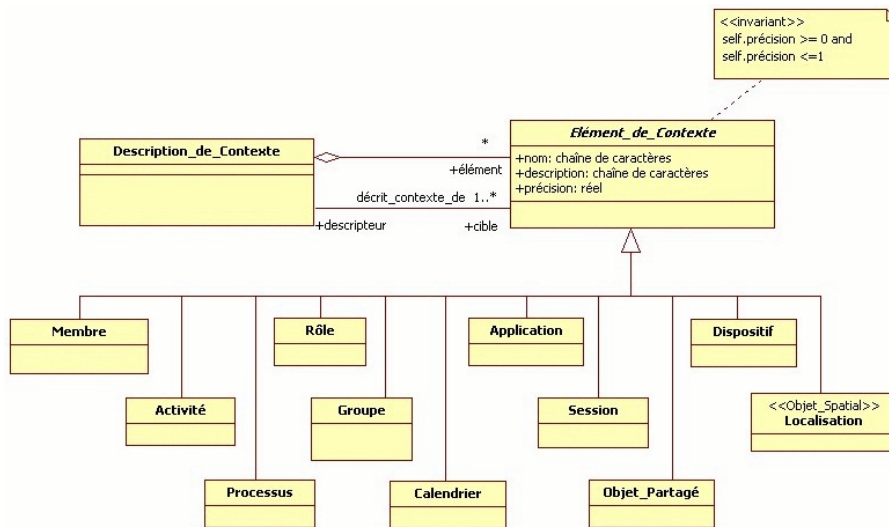


Figure 12. Premier niveau du modèle de contexte proposé dans [125].

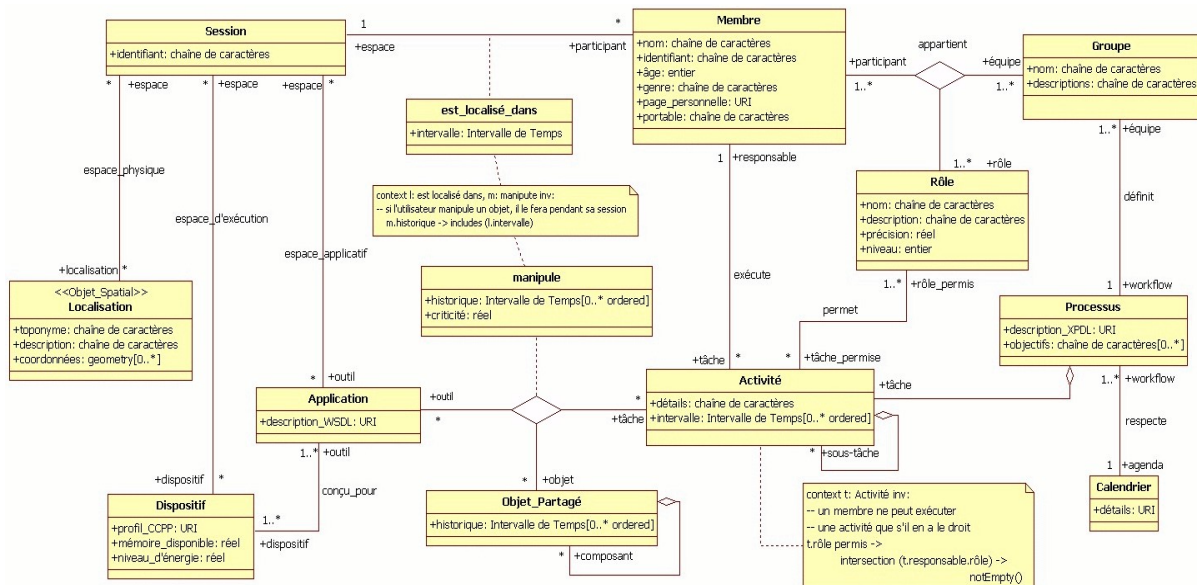


Figure 13. Détails du modèle de contexte proposé dans [125].

Ce modèle était parmi les premiers à étendre la notion de contexte à ses aspects collaboratifs, avec la prise en compte de concepts comme les notions d'activité, de rôle, ou encore d'objet partagé. On peut également citer Alarcon & Fuller [3], Alarcon *et al.* [4] et Borges *et al.* [29]. Ces aspects sont aujourd'hui reconnus par plusieurs auteurs comme étant particulièrement pertinents, surtout en ce qui concerne les collecticiels [67, 143, 185, 216]. Il convient également d'observer que, même si originellement il n'a pas été conçu ainsi, le premier niveau de ce modèle (Figure 12) peut être aisément assimilé à un méta-modèle définissant la notion de contexte comme un ensemble d'éléments de contexte observables à partir d'un autre élément (l'entité cible). On peut lui critiquer son manque de récursivité (un élément de contexte n'étant pas composé d'autres éléments de contexte), qui aurait peut-être permis de simplifier la description de certaines situations, comme par exemple, une machine virtuelle s'exécutant sur un certain dispositif.

Par ailleurs, l'approche utilisée par ce modèle reste particulière parmi les autres modèles de contexte. Il utilise en fait une approche de représentation de connaissances par objets, soutenu par un système de représentation de connaissances utilisant ce paradigme. A l'époque de sa publication (2004),

l'usage de l'orientation à objets comme paradigme de représentation n'était pas majoritaire (on peut citer, par exemple, [15]), d'autres paradigmes étant également utilisés (p.ex. les modèles XML/RDF [150]). Déjà à l'époque, l'usage d'ontologies pour la représentation de contexte s'intensifiait (comme l'illustrent [3,4,38,201]), devenant majoritaire dans l'Informatique Pervasive, notamment à partir de 2006-2008, sous l'impulsion des travaux autour des l'orientation service (e.g. [209,244,245, 248]). L'usage d'un système de représentation de connaissances par objets (et non d'un simple langage de programmation orienté objets) demeure à ce jour marginal. Cet usage permet de garder la simplicité de l'approche objets, tout en permettant une analyse à deux niveaux : intentionnelle et extensionnelle (c.a.d. de la structure et de ses instances), laquelle a permis la définition de plusieurs opérateurs et mesures de similarité utilisés en tant que mécanisme de raisonnement. Par exemple, on peut citer la mesure *SimO*, dont l'objectif était de comparer deux objets (*o* et *o'*) représentant deux éléments de contexte, en tenant compte des valeurs observées pour leurs attributs et de leur indicateur de précision (voir Figure 14). La prise en compte de la précision s'avère nécessaire en raison de la nature imprécise et dynamique de l'information de contexte : si deux objets se caractérisent par une faible précision, les informations qu'ils contiennent peuvent être erronées, ce dont la mesure de similarité doit rendre compte. Ainsi, l'attribut précision, défini dans le modèle de contexte est utilisé pour modérer la relation de similarité entre deux objets, puisque nous ne pouvons pas affirmer qu'il existe une forte similarité entre eux si leur contenu a un faible degré de précision (les informations sont trop imprécises pour que la mesure soit elle-même précise) [125].

- Soit $fd(o)$ la cardinalité de l'ensemble formé par les attributs de l'objet o dont la valeur est connue
 $fd(o) = \{ a_i \mid o.a_i \neq null \}$
- Soit $fi(v, v') \in [0, 1]$ une fonction d'identité capable de comparer deux valeurs v, v' en fonction de la nature de ces valeurs
- Soit o et o' deux objets appartenant à l'extension (ϵ) de la classe *Élément de Contexte* (EC)
 $o, o' \in EC_\epsilon, o = \{ a_i = v_i \}_{1 \leq i \leq fd(o)}, o' = \{ a_j = v_j \}_{1 \leq j \leq fd(o')}$
- la mesure de similarité *SimO* est alors définie comme

$$o.SimO(o') = \left(\sum_i^{fd(o,o')} fi(o.a_i, o'.a_i) \times w_i \right) \times fp(o, o'), \text{ où } fp(p, p') = \frac{o.précision + o'.précision}{2}$$

Figure 14. Définition de la mesure de similarité *SimO* d'après [125].

Ces mesures s'ajoutent aux opérations de *subsumes* et *plugin*, offertes par les systèmes de représentation de connaissances par objets. Elles tiennent profit également des associations existantes entre les concepts du modèle, tel des règles utilisées sur une ontologie. La richesse du modèle s'exprime ainsi non seulement à travers les concepts représentés (c.a.d. les classes), mais également à travers les associations et liens reliant les classes et objets.

Le mécanisme de filtrage, quant à lui, se distingue aussi bien par l'adaptation de l'information de conscience de groupe au contexte courant de l'utilisateur (semblable à [3,4]), que par l'expression des préférences de l'utilisateur en fonction elles-aussi du contexte. Ces préférences concernant les informations de conscience de groupe s'expriment au travers des profils associés à l'utilisateur. Chaque profil est associé à une ou plusieurs descriptions de contexte, décrivant les situations où il s'appliquerait, et identifie un ensemble d'informations de conscience de groupe considérées comme pertinentes dans ces situations. Le mécanisme agit ainsi en deux étapes : (i) d'abord, on sélectionne les profils pouvant s'appliquer en fonction du contexte courant de l'utilisateur ; (ii) puis, on filtre les

informations de conscience de groupe disponibles en fonction des profils sélectionnés. La sélection des profils applicables utilise les mesures de similarité mentionnées précédemment, comparant ainsi le contexte associé à chaque profil au contexte courant de l'utilisateur. Les profils sélectionnés sont ensuite appliqués, filtrant les informations disponibles selon les règles définies par ceux-ci.

Même s'il est à noter une certaine proximité avec des travaux de filtrage collaborative dans les systèmes de recommandation sensibles au contexte [2,259], la définition de plusieurs profils contextualisés n'était pas habituelle dans l'Informatique Pervasive au moment de la publication de ces travaux (2005-2006). Par ailleurs, l'hypothèse sous-jacente à ces travaux est également à souligner. Celle-ci serait que les besoins d'un utilisateur en matière de conscience de groupe peuvent varier en fonction de son contexte, ce qui a été mis en avant aussi par d'autres travaux postérieurs tels que [27, 34]. On suppose ainsi le développement par les utilisateurs de certaines habitudes (et donc la possibilité d'exprimer des préférences) pour de « situations types », qui agissent comme des « patrons » d'usage (par exemple, un administrateur réseau ayant l'habitude de consulter les logs d'activités de la nuit précédente en arrivant au bureau le matin). Même si cette hypothèse est reconnue par d'autres travaux (*e.g.* [64]), la définition de plusieurs profils, exprimant ces préférences, selon le contexte demeure minoritaire dans la littérature et a inspiré localement d'autres travaux comme [41]. Ces travaux ont également donné lieu à quelques collaborations postérieures [42, 124], à travers notamment l'usage du modèle de contexte proposé dans un système multi-agents.

Bien plus tardivement, ces travaux ont inspiré un travail de réflexion sur la prise en compte des notions de contexte et de conscience de groupe dans les collecticiels et dans les systèmes sensibles au contexte. Ce travail, mené en 2018-2019 [127, 128], s'interroge sur la proximité entre ces deux concepts et les différences dans leur usage et dans leur support logiciel. En effet, ces deux notions peuvent être vues comme une seule, comme une information capable de caractériser les interactions d'un utilisateur, soit un utilisateur isolé ou un membre d'une communauté. Cependant, leur traitement diffère selon le type d'application : dans les collecticiels, l'information de conscience de groupe est utilisée notamment pour la prise de décision ; l'objectif est donc de proposer cette information à l'utilisateur final, alors que, dans les systèmes sensibles au contexte, l'information de contexte est utilisée à des fins d'adaptation, de manière transparente à l'utilisateur. Ces différents focus conduisent naturellement à un traitement distinct de ces deux notions, même dans des travaux qui les combinent (*e.g.* [143, 160, 263]). Ainsi, nous avons mené une étude détaillée de la littérature [127, 128] qui nous a permis d'identifier et d'analyser les points communs et les différences de traitement entre ces deux notions. Ce travail se positionne comme un complément à ceux menés précédemment, dans lesquels le modèle de contexte discuté précédemment permet de tenir compte des aspects collaboratifs propres à la notion de conscience de groupe. La réflexion dans ce dernier travail a conduit à la proposition d'une vision unifiée pour le traitement de ces deux notions, perçues désormais comme une information de contexte aux usages multiples à travers une architecture conceptuelle unique pour son support. Cette architecture (voir Figure 15), inspirée des systèmes sensibles au contexte, encourage une meilleure séparation entre l'application, ses fonctionnalités d'adaptation et de collaboration, vues comme autant de services, et les mécanismes de gestion de contexte (acquisition, modèle, raisonnement et distribution) tenant compte des concepts issus de ces deux notions. L'adoption d'une telle architecture permettrait une simplification dans le développement de collecticiels, logiciels traditionnellement réputés par leur complexité, désormais sensibles au contexte.

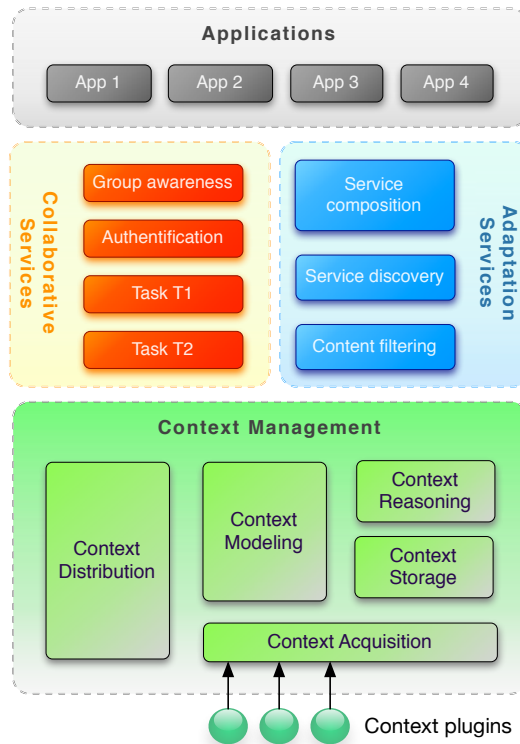


Figure 15. Architecture conceptuelle pour un traitement commun de la notion de contexte et de conscience de groupe [128].

Enfin, même datant de plusieurs années (essentiellement 2004-2006), ces travaux restent d'actualité, comme l'attestent les citations postérieures à 2013 (voir cf. section 1.3), car les problèmes traités sont toujours présents dans l'usage des collecticiels. Au-delà de ces systèmes, l'aspect collaboratif inhérent aux Systèmes d'Information rend ces propositions aussi applicables dans le cas de ces systèmes.

1.3 Bibliométrie

Ces travaux de recherche ont donné lieu à plusieurs publications essentiellement entre 2003 et 2008, dont une thèse en 2006. Sur un total de 17 publications, dont 11 publiées pendant la thèse (2002-2006) auxquelles s'ajoutent 6 autres articles publiés après 2006, neuf listées ci-dessous peuvent être soulignées par le nombre de citations ou par leur contenu. Ces publications ont été analysées, à l'aide des sites scholar.google.com et www.researchgate.net par rapport à leur nombre de citations, en tant qu'indicateur de leur impact. Ces citations ont été rangées en fonction de leur date de publication selon trois périodes : avant 2008, entre 2008 et 2013, postérieur à 2013. Les autocitations ont également été comptabilisées. Le Tableau 2 détaille les données obtenues, illustrées par la Figure 16.

- **COMIND 2003** [126] : M. Kirsch-Pinheiro, José Valdeni de Lima, Marcos R. S. Borges, "A framework for awareness support in groupware systems". **Computers in Industry**, 52(1): 47-57 (2003)
- **CRIWG 2004** [137] : M. Kirsch-Pinheiro, Jérôme Gensel, Hervé Martin, "Representing Context for an Adaptive Awareness Mechanism". In: Gert-Jan de Vreede, Luis A. Guerrero, Gabriela Marín Raventós (eds.), 10th International Workshop Groupware: Design, Implementation and Use, **CRIWG 2004**, LNCS 3198, 339-348 (2004)
- **MATA 2004** [138] : M. Kirsch-Pinheiro, Jérôme Gensel, Hervé Martin, "Awareness on Mobile Groupware Systems". In: Ahmed Karmouch, Larry Korba, Edmundo Roberto Mauro Madeira (eds.), First International Workshop on Mobility Aware Technologies and Applications, **MATA 2004**, LNCS 3284, 78-87 (2004)

- **SAC 2005** [139] : M. Kirsch-Pinheiro, Marlène Villanova-Oliver, Jérôme Gensel, Hervé Martin, "Context-aware filtering for collaborative web systems: adapting the awareness information to the user's context". In: Hisham Haddad, Lorie M. Liebrock, Andrea Omicini, Roger L. Wainwright (eds.), Proceedings of the 2005 ACM Symposium on Applied Computing (**SAC 2005**), 1668-1673 (**2005**)
- **CSCWD 2005** [140] : M. Kirsch-Pinheiro, Marlène Villanova-Oliver, Jérôme Gensel, Hervé Martin, "BW-M: a framework for awareness support in Web-based groupware systems". In: Weiming Shen, Anne E. James, Kuo-Ming Chao, Muhammad Younas, Zongkai Lin, Jean-Paul A. Barthès (eds.), Proceedings of the Ninth International Conference on Computer Supported Cooperative Work in Design, **CSCWD 2005**, Volume 1, 240-246 (**2005**)
- **Thèse** [125] : M. Kirsch-Pinheiro, « Adaptation Contextuelle et Personnalisée de l'Information de Conscience de Groupe au sein des Systèmes d'Information Coopératifs », **PhD Thesis**, Université Joseph Fourier - Grenoble I, Grenoble, France (**2006**)
- **CAISE 06 Workshop** [141] : M. Kirsch-Pinheiro, Marlène Villanova-Oliver, Jérôme Gensel, Hervé Martin, "A Personalized and Context-Aware Adaptation Process for Web-Based Groupware Systems". Proceedings of the **CAISE 06, Workshop** on Ubiquitous Mobile Information and Collaboration Systems, **UMICS 2006** (**2006**)
- **Ubicomm 2008** [142] : M. Kirsch Pinheiro, Marlène Villanova-Oliver, Jérôme Gensel, Yolande Berbers, Hervé Martin, "Personalizing Web-Based Information Systems through Context-Aware User Profiles", *International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*, **Ubicomm 2008**, (**2008**)
- **EGC 2008** [93] : Jérôme Gensel, Marlène Villanova-Oliver, M. Kirsch-Pinheiro, « Modèles de contexte pour l'adaptation à l'utilisateur dans des Systèmes d'Information Web collaboratifs », *8èmes Journées Francophones d'Extraction et Gestion des Connaissances (EGC'08)*, *Atelier sur la Modélisation Utilisateur et Personnalisation d'Interfaces Web*, 5-15 (**2008**)

Tableau 2. Analyse bibliométrique de publications choisies.

Référence	Année	Total	≤ 2008	> 2008 et ≤ 2013	> 2013	Self (autocitation)
COMIND 2003	2003	90	49	20	10	11
CRIWG 2004	2004	71	13	21	13	24
MATA 2004	2004	11	4	4	1	2
ACM SAC 2005	2005	45	16	22	4	2
CSCWD 2005	2005	8	2	3	2	1
Thèse	2006	8	4	2	1	1
CAISE 06 Workshop	2006	13	0	7	4	2
Ubicomm 2008	2008	7	0	4	3	0
EGC 2008	2008	18	2	6	7	3
Total / %		271	33,95 %	32,84 %	16,61 %	16,61 %

Plusieurs éléments peuvent ressortir de l'analyse de ces informations (Figure 16 et Tableau 2). L'article le plus cité est celui publié dans la revue COMIND, portant sur les mécanismes de support à la conscience de groupe. Même si celui-ci ne fait techniquement pas partie de ces contributions, il a donné les bases sur lesquelles ces contributions ont été bâties. On peut également observer le nombre important d'autocitations pour l'article CRIWG 2004. Celui-ci s'explique par l'aspect fondateur de cet article par rapport aux travaux postérieurs. Il introduit le modèle de contexte proposé, constituant ainsi le point de référence pour la suite de ces travaux.

Il convient également d'observer un changement de communauté opéré au cours de ces travaux. Les contributions pouvant se placer aussi bien dans le domaine de l'Informatique Pervasive que dans la communauté TCAO (Travail Coopérative Assisté par Ordinateur), ces différentes publications ont ainsi ciblé ces deux communautés distinctes. Les premiers travaux (COMIND 2003, CRIWG 2004, CSCWD 2004) visaient notamment la communauté TCAO, alors que les publications qui ont suivi (MATA 2004, SAC 2005, CAISE 06 Workshop, Ubicomm 2008) visaient principalement l'Informatique Pervasive, qui sera abordé dans le prochain chapitre.

Il est aussi à observer la période où ces travaux ont eu plus d'influence (quantifiée par le nombre de citations) est la période allant jusqu'à 2013, correspondant à une période où les travaux sur l'adaptation au contexte étaient également nombreux.

Enfin, il est important de souligner l'importance de ces travaux sur les contributions postérieures : les opérations utilisées dans le mécanisme de filtrage proposé ont influencé les travaux sur la sélection de services, ainsi que le modèle de contexte qui a influencé les travaux sur les Systèmes d'Information, sans mentionner les collaborations réalisées depuis 2008 [42, 124] rendues possibles par ces travaux.

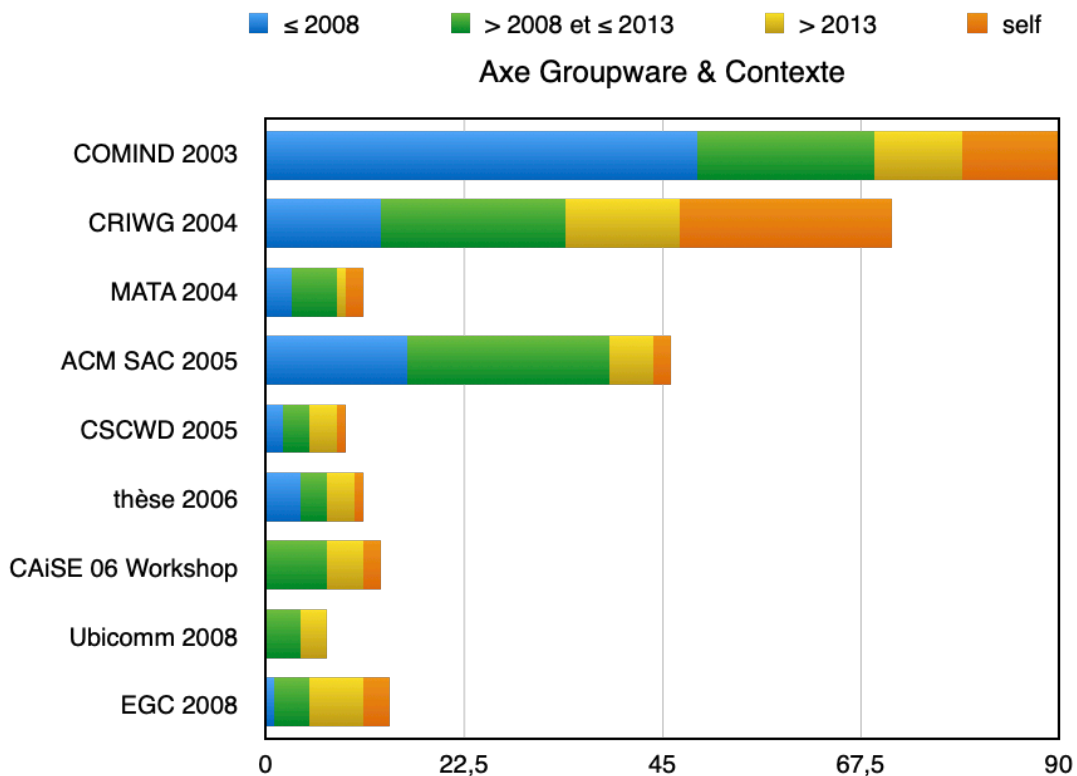


Figure 16. Évolution des citations dans le temps.

2 Contexte dans les environnements pervasifs

Dans ce chapitre, on se place dans le domaine de l'Informatique Pervasive. Celle-ci peut être définie comme l'intégration transparente de l'informatique et de ses dispositifs dans notre quotidien [13]. Aussi connu sous la dénomination d'Informatique Ubiquitaire, ce terme représente, selon Moran & Dourish [167], une tendance vers des environnements enrichis par des nombreux dispositifs informatiques, souvent mobiles ou embarqués dans l'environnement, connectés par des réseaux fixes ou sans fil. Originellement proposée par Weiser [264], cette vision d'une informatique devenue invisible à nos yeux devient petit à petit une réalité, puisque, comme le souligne Bell & Dourish [20], nous utilisons continuellement des ressources informatiques sans forcément les percevoir comme des ordinateurs.

Les challenges soulevés dans ce domaine sont multiples, liés notamment à la dynamique et à l'hétérogénéité de ces environnements dits pervasifs. En effet, ces environnements se caractérisent par leur hétérogénéité, comportant des dispositifs aussi variés que des équipements réseaux (routeur, *switch*...), des ordinateurs (fixes ou portables), des *smartphones* et tablettes, ou encore des dispositifs utilisés pour l'IoT (*e.g.* RaspberryPI, Arduino). Ils sont également souvent très dynamiques, intégrant des dispositifs pouvant entrer et sortir du réseau, se déconnecter ou s'étendre facilement. Des nouveaux dispositifs peuvent facilement apparaître dans le périmètre, alors que d'autres peuvent disparaître en fonction de leur usage, de leur mobilité (ou celle de leur propriétaire), de l'état de leur batterie ou niveau d'alimentation, etc. La notion de contexte devient ainsi un élément clé dans ce domaine, promettant une meilleure prise en compte de ces environnements. L'observation du contexte d'exécution de ces ressources représente une information particulièrement intéressante pour les applications et plateformes voulant tirer profit de ces environnements et des capacités d'interaction, de stockage ou même de calcul s'y trouvant.

Dans ce chapitre, nous nous intéressons particulièrement à ces environnements, en se focalisant sur deux problèmes distincts : tout d'abord, comment rendre disponible l'information de contexte dans ces environnements (Section 2.1) ; puis comment, à partir de l'information de contexte, rationaliser l'usage des ressources de calcul dans ces environnements, notamment pour l'exécution d'applications de type « *Big Data* » (Section 2.2).

2.1 Context grouping

2.1.1 Rappel de la problématique

Les applications sensibles au contexte se présentent de plus en plus comme des applications réparties. Elles peuvent impliquer différents nœuds qui peuvent coopérer pour atteindre les objectifs de l'application ou simplement pour offrir à l'utilisateur une meilleure expérience. Différents scénarios peuvent illustrer cette tendance, comme, par exemple, des scénarios impliquant des systèmes multi-échelles [217], ou l'adaptation d'une application aux ressources matérielles et logicielles disponibles dans l'environnement [65], ou encore la possibilité de proposer une composition opportuniste de services à l'utilisateur [68]. Quoiqu'il en soit le caractère réparti de ces applications et services peut impliquer également la distribution des informations de contexte à travers les différents éléments présents dans l'environnement.

En effet, afin de bénéficier pleinement de l'environnement pervasif qui les entoure, certaines applications doivent compter sur l'échange d'informations concernant les ressources environnantes et leur contexte d'exécution. Par exemple, une application de communication peut tirer profit de la présence d'un nœud possédant une meilleure capacité d'affichage ou de connexion à proximité et vouloir utiliser ces ressources à sa disposition en déployant certaines tâches sur les nœuds voisins. Ce scénario, envisagé par exemple sur [64], demande le partage, entre nœuds voisins, de leur contexte d'exécution. Du même, des scénarios impliquant le partage de ressources dans un bâtiment intelligent

(*smart building*), tel que considéré par [89], demanderaient également le partage d'informations de contexte sur les ressources disponibles afin que tout le potentiel offert par ces environnements puisse être exploité.

Cependant, les environnements pervasifs étant caractérisés par leur dynamisme, le partage d'informations concernant le contexte d'exécution des ressources présentes dans l'environnement soulève différents challenges. Non seulement le contexte de ces ressources va évoluer dans le temps, mais la composition de cet environnement va également évoluer, avec des nouvelles ressources y entrant ou des ressources le quittant ou devenant simplement indisponibles. Par ailleurs, toutes les applications ne sont pas forcément intéressées par toutes les informations de contexte disponibles, sans parler des questions évidentes de sécurité et de respect de la vie privée impliquant un meilleur contrôle des informations partagées entre les nœuds.

Dans cette contribution, nous nous intéressons à la question de comment partager des informations de contexte dans un environnement pervasif. Sans entrer dans les questions liées à la sécurité et à la vie privée, les présents travaux traitent particulièrement du dynamisme de l'information de contexte elle-même et de l'environnement. Ces informations doivent être mises à jour de manière régulière, en fonction des changements opérés sur le contexte observé. Même chose pour l'environnement dont la composition est fluctuante. Ainsi, nous focalisons dans ces travaux la distribution de l'information de contexte dans un environnement dynamique dont la composition et la nature varie dans le temps, à travers un mécanisme de distribution de contexte pair-à-pair.

2.1.2 Contributions & Impact

Dans les présents travaux, réalisés essentiellement entre 2008 et 2009, en partie dans le cadre du projet IST-MUSIC, nous avons proposé un mécanisme de distribution de contexte dans lequel les nœuds présents dans l'environnement sont dynamiquement organisés en groupes, en fonction d'un contexte commun partagé par les membres de chaque groupe. L'usage de l'information de contexte est ainsi double : elle est utilisée pour organiser les groupes et est aussi partagée entre les membres de ces groupes. Les nœuds, qui représentent les ressources disponibles dans l'environnement, s'organisent en groupes, en fonction d'un contexte commun, selon des critères prédéfinis au niveau applicatif. Par exemple, on peut définir un groupe pour des ressources co-localisées (*e.g.* localisées dans une même salle), ou disposant d'un même type de connexion réseau, ou encore appartenant à des utilisateurs partageant le même rôle dans l'organisation. A l'intérieur de ces groupes, c'est aussi l'information sur le contexte de leurs membres qui est distribuée parmi les autres membres, lesquels peuvent ainsi prendre conscience de la situation courante des autres ressources appartenant au groupe. Par exemple, dans le groupe défini sur une localisation commune (*i.e.* ressources co-localisées), il est possible de partager parmi les membres les informations concernant la mémoire disponible et les capacités d'affichage. Un tel partage serait utile, par exemple, dans une application de communication comme celle envisagée dans [64].

Les groupes sont définis par les applications dans un profil composé de 2 parties : les critères de formation (*criteria set* C_D), indiquant les critères d'appartenance d'un nœud au groupe ; et l'ensemble de dissémination (*dissemination set* I_D), indiquant les informations qui seront propagées dans le groupe ; tous les deux définis à partir d'informations de contexte :

$$G_D = \langle C_D, I_D \rangle$$

Il s'agit en effet des requêtes exprimées en CQL (*Context Query Language* [210]), langage proposé dans le cadre du projet IST-MUSIC, dont un exemple est illustré par la Figure 17. Le profil de chaque groupe agit donc comme un modèle (un « *template* »), instancié à un instant t en fonction du contexte courant de chaque nœud. Les groupes étant définis à partir d'informations contextuelles, leur composition est mise à jour dynamiquement : au fur et à mesure que le contexte d'exécution d'un nœud évolue, son appartenance à un groupe peut être remise en question, et il peut ainsi quitter une instance de ce

groupe pour rejoindre (ou fonder) une nouvelle instance, conformément aux nouvelles valeurs associées à son contexte courant. Par exemple, dans le cas d'un groupe dont le critère de formation repose sur la localisation, le déplacement d'un nœud d'une localisation A vers une localisation B va le conduire à quitter l'instance du groupe associée à cette première pour rejoindre l'instance associée à la seconde. La composition de chaque instance du groupe évolue ainsi au gré des changements de contexte de leurs membres. De même pour les informations répandues à l'intérieur de chaque instance : l'ensemble de dissémination reposant sur des informations de contexte, à chaque changement dans celles-ci, le nœud concerné se chargera de transmettre les nouvelles valeurs aux autres membres du groupe, permettant ainsi de les garder à jour par rapport à l'évolution de son état courant.

```

<contextQL>
  <ctxQuery>
    <action type="SELECT"/>
    <entity ontConcept="prefix:music:username">#any</entity>
    <scope ontConcept="prefix:music:Position"
      ontRep="prefix:music:DefaultPositionRep">#position</scope>
  </ctxQuery>
</contextQL>

```

Figure 17. Requête CQL définissant un groupe basé sur la localisation de son utilisateur [134].

Pour cela, le mécanisme repose sur un réseau pair-à-pair et sur les mécanismes de regroupement (« *grouping* ») existants dans ces réseaux. Il en résulte une architecture à deux niveaux (illustrée par la Figure 18), dans laquelle un premier niveau, dit niveau de « distribution » (« *distribution layer* ») s'occupe de la distribution de l'information reposant sur un réseau pair-à-pair (en occurrence ici, une mise en œuvre reposant sur la bibliothèque JXTA [95]), indépendamment du contexte courant ; et un niveau dit de « raisonnement » (« *reasoning layer* »), lequel s'occupe de la gestion de groupes en fonction du contexte, pilotant le niveau inférieur en fonction des changements observés dans l'information de contexte.

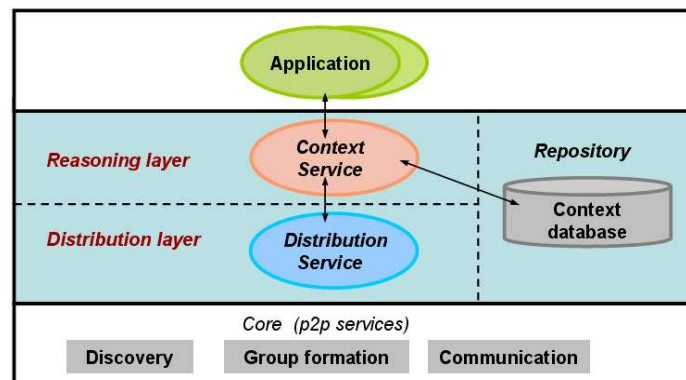


Figure 18. Architecture mise en place de pour le mécanisme de distribution de contexte proposé [134].

Par ailleurs, il convient d'observer qu'un même nœud peut appartenir à différents groupes. Chaque application pouvant définir ses propres profils, un même nœud peut appartenir à plusieurs groupes, en fonction des besoins des applications exécutées sur ces nœuds. Par exemple, la Figure 19 illustre deux groupes distincts qui se partagent un nœud commun. Celui-ci exécute donc deux applications distinctes, chacune proposant un groupe de distribution.

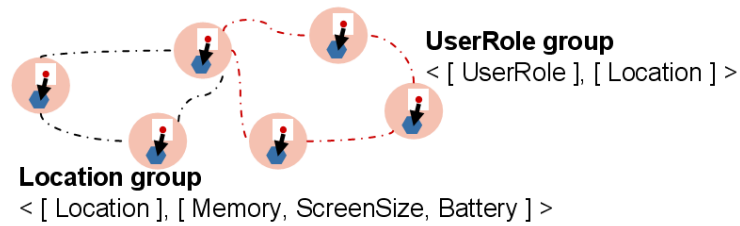


Figure 19. Illustration de différents nœuds appartenant à deux groupes distincts [134].

La définition d'un profil de groupe restant prédéfinie et relativement statique, elle constituait un point faible de la proposition. Nous avons ainsi étudié *a posteriori* (en 2012-2013) [254] la possibilité de modifier les profils des groupes, voire d'en définir des nouveaux, de manière plus dynamique, à partir des informations de contexte courant observées au sein des nœuds disponibles dans l'environnement. Nous avons ainsi considéré l'usage de l'Analyse de Concepts Formels (*Formal Concept Analysis* – FCA) pour identifier les possibles critères de formation des groupes, à partir des données observées dans les réseaux. Cette proposition repose ainsi sur une architecture hybride, dans laquelle des « super-nœuds » concentrent des informations sur des nœuds voisins. Ces super-nœuds peuvent de cette manière analyser les différentes informations de contexte partagées entre les nœuds afin d'identifier des éléments communs partagés entre différents nœuds, susceptibles de former des nouveaux groupes. Il s'agit dans ce cas d'identifier, à partir des données, des groupes possibles au lieu de reposer uniquement sur des groupes préétablis par les applications en exécution. Il s'agit aussi pour nous d'une première expérience avec cette méthode d'analyse de données (le FCA) qui sera détaillée plus tard (cf. section 3.1), où elle a été utilisée de manière plus intensive.

Les mécanismes de distribution de l'information de contexte comme le nôtre sont plutôt rares dans la littérature, comparativement à d'autres problématiques liées à la gestion de contexte, telle que la modélisation de cette information. Bellavista *et al.* [21] soulignent d'ailleurs différents travaux traitant de ce sujet. On peut alors observer que, malgré les limitations liées à une possible passage à l'échelle, les approches considérant un élément central (*e.g.* un serveur, un proxy, etc.), qui centralise la gestion des groupes ou la distribution de l'information, demeurent nombreuses. Même des travaux comme [71, 70], qui considèrent également des échanges point-à-point, supposent à un moment donné de l'échange (notamment au début), la présence d'un élément centralisateur (un serveur SIP dans le cas de [71, 70]). On retrouve des travaux prônant une architecture totalement décentralisée comme le nôtre dans le cadre des réseaux MANET (*Mobile Ad-hoc Networks*) ou VANET (*Vehicular Ad-hoc Networks*), comme dans le cas de Paridel *et al.* [193]. Cependant, dans ces cas, les groupes sont souvent formés uniquement par proximité géographique, se limitant à une notion de voisinage physique. Des travaux comme le nôtre, permettant à la fois l'usage d'une architecture décentralisée et une définition particulièrement flexible de la notion de groupe, demeurent rares.

L'adoption d'une architecture totalement répartie, sans la présence d'un élément central, constitue un point fort pour le support aux environnements pervasifs, notamment dans le support au caractère dynamique de ces environnements. Par contre, la gestion de la communication sans un tel élément peut s'avérer plus complexe, en raison de ce même caractère dynamique, puisque la composition de l'environnement peut varier fortement dans un court intervalle de temps. Dans cette contribution, nous nous sommes basés sur les services déjà offerts par les plateformes de communication P2P, au lieu de proposer des nouveaux protocoles de communications qui nous semblaient alors redondants par rapport à l'existant. Ainsi, le caractère totalement réparti de notre solution repose essentiellement sur les services offerts par la plateforme P2P JXTA [95], qui, à l'époque de la proposition (2008), était encore une plateforme P2P très populaire en Java. Le choix d'une architecture à deux niveaux, avec une séparation claire entre la gestion des groupes (*c.a.d.* le mécanisme de groupes lui-même) et la communication entre les pairs, s'est avéré un point particulièrement important de notre proposition, puisque la gestion de l'information de contexte se retrouve ici isolée des canaux de communication utilisés. Il serait alors aisé de faire évoluer le mécanisme proposé vers de nouvelles plateformes de

communication P2P, comme par exemple FreePastry⁵ ou TomP2P⁶, voir même vers de paradigmes de communication autres que les réseaux pair-à-pair.

Enfin, la définition de groupes à partir des informations de contexte offre une souplesse additionnelle à ce mécanisme. En effet, la notion de voisinage exprimée par les groupes se retrouve ici élargie. Celle-ci ne limite pas à un voisinage réseau (nœuds connectés sur un même sous-réseau) ou à un voisinage purement géographique (nœuds se trouvant à proximité géographique). N'importe quelle information de contexte peut être utilisée pour créer cette notion de voisinage : des utilisateurs disposant d'un même rôle, des dispositifs disposant des mêmes capacités de stockage, etc. On s'approche alors de la notion de communauté soutenue par différents auteurs de la communauté P2P, tel que Khambatti *et al.* [122], où une communauté de pairs se définit en fonction des attributs communs. Mais, contrairement à ces travaux, dans le nôtre, la valeur de ces attributs communs peut varier fortement dans le temps, puisqu'il s'agit d'information de contexte. Il en résulte des groupes au caractère très dynamique, dont la composition change en fonction de l'environnement. Le mécanisme offre ainsi plus de flexibilité aux applications, car celles-ci peuvent définir des communautés en fonction de leurs besoins, gérant ainsi la distribution de l'information de contexte en fonction de ces besoins.

L'application contrôle ainsi non seulement la formation des groupes (et donc la notion de voisinage), mais également le contenu retransmis à l'intérieur du groupe. Par la définition des ensembles de dissémination, les applications peuvent contrôler les informations de contexte qui seront partagées avec les autres nœuds, évitant ainsi le transfert d'information inutile et une exposition superflue de ces informations sur le réseau. En effet, chaque application n'est pas forcément intéressée par toutes les informations de contexte pouvant être observées. Le fait de limiter les informations qui seront transmises à uniquement celles pouvant intéresser les applications en exécution et uniquement aux nœuds partageant un certain contexte permet d'éviter que toutes les informations observées soient systématiquement retransmises à tous les nœuds voisins.

2.1.3 Bibliométrie

Les travaux décrits dans ce chapitre ont été réalisés entre 2008 et 2009, avec notamment la publication d'un article à la conférence DOA 2008 (*Distributed Objects, middleware and Applications*) [134]. Ces travaux ont été repris en 2012, avec une nouvelle publication en 2013 [254]. Par ailleurs, la réalisation de ces travaux a permis le développement d'autres collaborations sur le sujet, notamment dans le cadre du projet IST-MUSIC. Nous avons d'abord eu la proposition d'une architecture P2P permettant la distribution de contexte dans le cadre de ce projet [109], puis un travail sur la question du respect de la vie privée et la distribution de contexte guidée par des politiques propres à l'utilisateur (en opposition aux politiques propres aux applications, comme dans notre cas) [73]. Chacune de ces collaborations a donné lieu à une publication. Ainsi l'ensemble de publications liées à la distribution de contexte se retrouve ainsi résumée dans la liste ci-dessous.

- **DOA 2008 [134]** : Kirsch-Pinheiro, M.; Vanrompay, Y.; Victor, K.; Berbers, Y.; Valla, M.; Frà, C.; Mamelli, A.; Barone, P.; Hu, X.; Devlic, A.; Panagiotou, G., "Context Grouping Mechanism for Context Distribution in Ubiquitous Environments", In: Robert Meersman, Zahir Tari et al. (eds.), *10th International Symposium on Distributed Objects, Middleware, and Applications (DOA'08), OTM 2008 Conferences, Lecture Notes in Computer Science*, 5331, **2008**, 571-588.
- **ChapCtxGrp 2013 [254]** : Vanrompay, Y.; Kirsch Pinheiro, M.; Ben Mustapha, N.; Aufaure, M.-A., "Context-Based Grouping and Recommendation in MANETs", In : Kolomvatsos, K., Anagnostopoulos, C., Hadjiefthymiades, S. (Eds.), *Intelligent Technologies and Techniques for Pervasive Computing*, IGI Global, **2013**, 157-178.

⁵ <http://www.freepastry.org/FreePastry/>

⁶ <https://tomp2p.net>

- **ISD 2008 [109]** : Hu, X.; Ding, Y.; Paspallis, N.; Bratskas, P.; Papadopoulos, G.A.; Vanrompay, Y.; Kirsch Pinheiro, M.; Berbers, Y., "A Hybrid Peer-to-Peer Solution for Context Distribution in Mobile and Ubiquitous Environments", In: Papadopoulos G., Wojtkowski W., Wojtkowski G., Wrycza S., Zupancic J. (eds), *17th International Conference on Information Systems Development (ISD2008), Information Systems Development: Towards a Service Provision Society*, **2008**, Springer, 501-510. DOI : 10.1007/b137171_52
- **CoMoRea 2009 [73]** : Devlic, A.; Reichle, R.; Wagner, M.; Kirsch Pinheiro, M.; Vanrompay, Y.; Berbers, Y.; Valla, M., "Context inference of users' social relationships and distributed policy management", *6th IEEE Workshop on Context Modeling and Reasoning (CoMoRea), 7th IEEE International Conference on Pervasive Computing and Communication (PerCom'09)*, Galveston, Texas, 13 March **2009**. DOI : 10.1109/PERCOM.2009.4912890

Comme pour le chapitre précédent, nous avons analysé ces publications par rapport à leur nombre de citations. Nous avons utilisé pour cela différentes sources, dont notamment les sites scholar.google.com, www.researchgate.net et hal.archives-ouvertes.fr, ainsi que le site de l'éditeur, lorsque celui-ci était disponible. Ces citations ont été organisées, par la suite, en trois catégories : avant 2013 (ce qui correspond, approximativement, aux 5 premières années suivant l'apparition de la publication), entre 2013 et 2016, et après 2016. Les autocitations ont également été comptabilisées et distinguées des autres citations. Le Tableau 3 détaille les données obtenues, illustrées par la Figure 20.

Tableau 3. Nombre de citations observées pour les publications relatives à la thématique distribution de contexte.

Référence	Année	Total	≤ 2013	> 2013 et ≤ 2016	> 2016	self
DOA 2008	2008	13	7	2		4
ChapCtxGrp 2013	2013	6	1			3
ISD 2008	2008	6	5			1
CoMoRea 2009	2009	22	12	6		3
Total / %		47	53,19 %	17,02 %	6,38 %	23,4 %

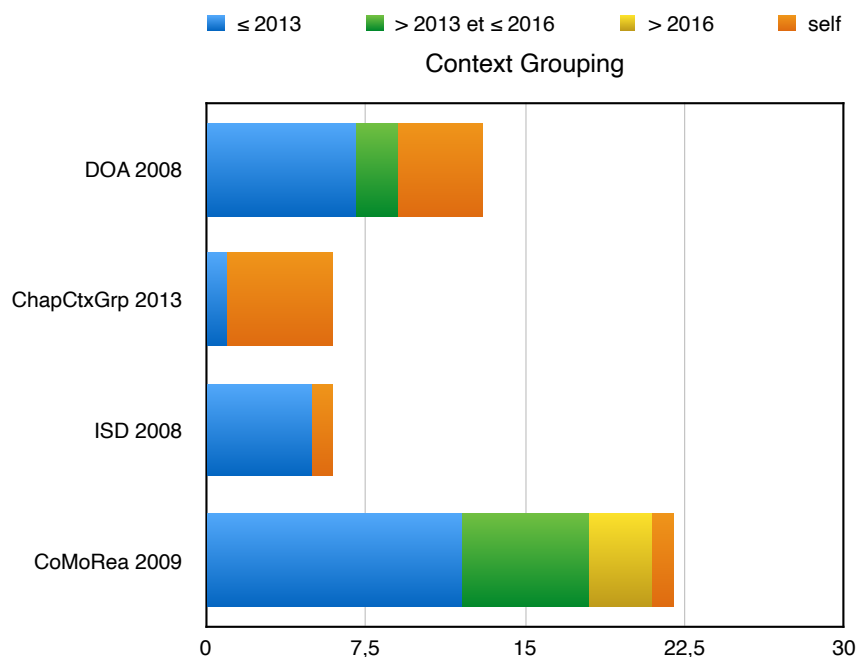


Figure 20. Graphique illustrant l'évolution des citations au cours des années.

Comme on pouvait s'attendre, les citations concernant ces articles se concentrent essentiellement sur les 5 premières années suivant leur publication (entre 2008 et 2013 donc). Ceci correspond à une période très intense pour les recherches sur l'informatique sensible au contexte. Ces travaux ont donc été repris par la communauté pendant cette période, mais n'ayant pas eu de suite, ces travaux ont petit à petit cédé leur place à des approches plus récentes.

2.2 Projet PER-MARE

2.2.1 Rappel de la problématique

Contrairement à ce qu'on peut le croire, les environnements pervasifs peuvent offrir d'importantes capacités de calcul réparties entre de nombreuses unités disponibles sur celui-ci. Ces différentes unités peuvent, en théorie, collaborer pour la collecte et le traitement des données issues notamment des capteurs afin de réaliser certaines tâches de manière autonome et adaptative. Ce scénario envisagé, entre autres, par Ausiello [11], illustre bien l'intérêt de l'usage de ces environnements pour l'exécution de certaines tâches de calcul, notamment pour les applications de type *Big Data*.

Cependant, l'hétérogénéité propre aux environnements pervasifs représente un challenge intéressant en matière de calcul. Avec des ressources aux caractéristiques très variées, voire diamétralement opposées, allant des serveurs haute performance (HPC) aux nano-ordinateurs (Raspberry PI, Arduino, etc.), l'usage à des fins de calcul des ressources intégrés à ces environnements représente un défi, notamment en raison des difficultés pour le placement des tâches de calcul sur ces ressources très hétérogènes. Comme l'a souligné Breitbach *et al.* [31], le placement de tâches de calcul dans un environnement hétérogène est plus complexe que celui réalisé en environnement Cloud ou sur les grilles de calcul. En effet, il est particulièrement difficile de distribuer des tâches de calcul et de garantir leur exécution sur des ressources qui ne se comportent pas de la même manière et dont les performances peuvent être très variées et variables. Cette hétérogénéité peut avoir un impact important sur l'exécution de ces tâches et sur leur performance. Le dynamisme de ces environnements, avec des nœuds particulièrement volatiles, qui peuvent disparaître ou rejoindre le réseau en cours d'exécution, va également avoir un impact sur les performances. Dans ces conditions, on peut difficilement anticiper les performances d'exécution d'un ensemble de tâches dans un tel environnement. Ceci est d'autant plus vrai pour les applications de type *Big Data*, qui doivent allier ce besoin de calcul à la gestion et au transfert d'importantes masses de données.

Le projet PER-MARE est né de ce constat. Allant de 2013 à 2014, ce projet est une coopération internationale CAPES/MAEE/ANII STIC-AmSud (project number 13STIC07) impliquant l'Université de Reims Champagne-Ardennes et l'Université Paris 1 Panthéon Sorbonne en France, ainsi que l'Universidade Federal de Santa Maria (UFSM) au Brésil et Universidad de la República en Uruguay. L'objectif principal du projet PER-MARE était d'offrir un support pour l'exécution d'applications de type MapReduce dans un environnement pervasif. MapReduce est un modèle de programmation pour les applications « data-intensives » dans lequel le calcul s'organise en deux phases : le *map*, où les données sont divisées en plusieurs blocs et traitées pour former un ensemble de pairs « <clé, valeur> » ; et le *reduce*, où les résultats de la première phase sont agrégés afin d'aboutir à un résultat final [265]. Les données divisées en plusieurs blocs sont ainsi réparties entre les unités de calcul et traitées dans la phase de *map*, résultant en un ensemble des pairs « < clé, valeur > », qui sont par la suite regroupés à nouveau en plusieurs blocs en fonction des valeurs de clé retrouvées, pour être enfin consolidés dans la phase *reduce* (voir Figure 21). Par son caractère facilement réparti, le principe de MapReduce a été largement utilisé par les plateformes de traitement *Big Data*, dont Apache Hadoop⁷, qui est était la principale plateforme *Big Data* à l'époque du démarrage du projet PER-MARE.

⁷ <https://hadoop.apache.org>

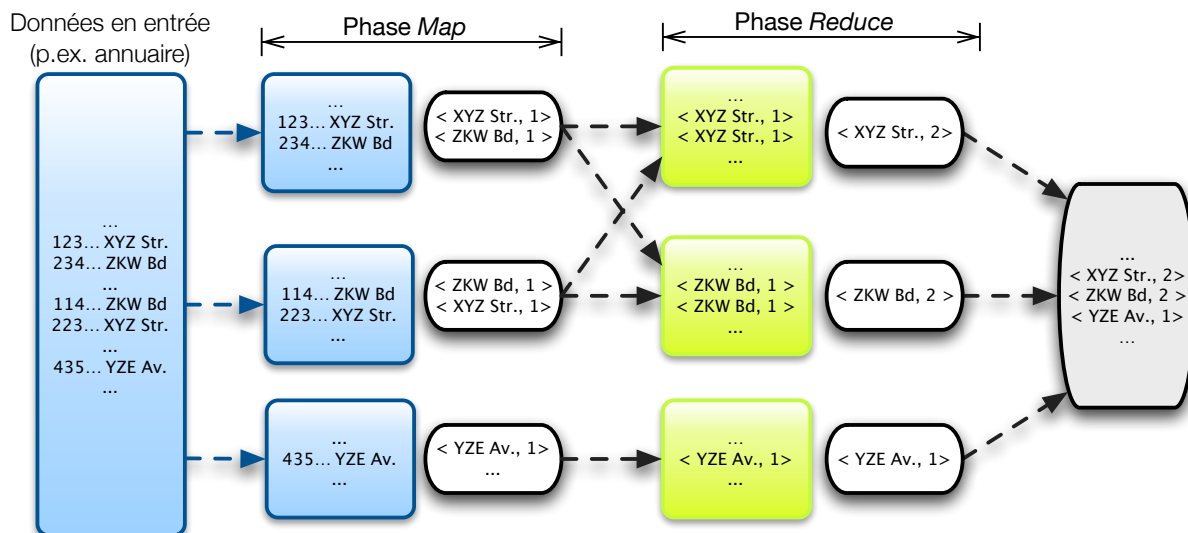


Figure 21. Illustration du principe MapReduce sur un exemple d'annuaire, où on comptabilise le nombre de téléphones fixes (« 123... ») par adresse postal (« XYZ Str... »).

L'exécution d'applications de type MapReduce dans un environnement pervasif demande un usage rationnalisé des ressources présentes dans ces environnements, afin de permettre une bonne distribution des tâches de calcul et des données selon les capacités des dispositifs disponibles. Le projet PER-MARE se place ainsi dans le cadre des grilles pervasives (« *pervasive grids* ») [192], dont l'objectif serait de constituer dynamiquement, de manière opportuniste, des grilles de calcul à partir des ressources disponibles à proximité. Selon Parshar & Pierson [192], les grilles pervasives représentent la généralisation extrême de la notion de grille de calcul, dans laquelle les ressources sont pervasives. Ces grilles peuvent ainsi intégrer aussi bien des dispositifs de type capteur ou actuateur que des terminaux classiques de haute performance. Les *desktop grids*, grilles composées de terminaux de type desktop PC mis à disposition par des utilisateurs volontaires, seraient ainsi un cas particulier de grille pervasive, par définition très hétérogène [235].

Les grilles pervasives peuvent être assimilées aux plateformes de *Fog* ou *Edge computing* [58, 187]. *Fog / Edge Computing* correspondent à une nouvelle tendance complémentaire au *Cloud Computing*. Il s'agit d'une alternative au modèle « tout cloud » dans lequel tous les traitements se font exclusivement sur le *cloud*. L'objectif serait alors d'utiliser les dispositifs à proximité pour la réalisation de certains traitements (agrégation de données, prétraitements, anonymisation, certains calculs simples, etc.), au lieu de toujours tout envoyer pour un traitement à distance sur des plateformes de *cloud* ou des centres de calcul (*datacenter*). Il s'agit d'un modèle particulièrement intéressant pour minimiser les problèmes liés à la latence du réseau ou au transfert des grandes quantités de données vers le *cloud*, ou encore à des problèmes liés au respect à la vie privée ou à la sécurité des données [105, 224], qui restent alors au plus près de leur environnement de production. En minimisant, voire en évitant, les transferts de données vers des plateformes distantes et en tirant profit des ressources à proximité, ce modèle permet non seulement de réduire ces problèmes mais également de mieux utiliser les ressources disponibles (et souvent sous-exploitées) en « bordure » du réseau ou à proximité de l'utilisateur.

Le projet PER-MARE explore ainsi l'usage des ressources hétérogènes pour des calculs de type *Big Data*. Deux approches complémentaires ont été explorées par le projet : (i) la prise en compte du contexte d'exécution dans la gestion des ressources au sein de la plateforme Hadoop ; et (ii) l'usage des ressources disponibles dans une plateforme de *Fog/Edge computing*. Dans les deux cas, la problématique demeure celle d'un usage opportuniste des ressources hétérogènes pour le *Big Data*. Chaque approche a donné lieu à ses propres contributions, détaillées dans ce qui suit.

2.2.2 Contributions

2.2.2.1 Prise en compte du contexte d'exécution sur la plateforme Hadoop

Apache Hadoop [265] est une plateforme particulièrement conçue pour la réalisation et le déploiement d'applications de type MapReduce sur de grilles de calcul. Grâce à Hadoop, un concepteur de ce type d'application peut se concentrer uniquement sur la conception des fonctions *map* et *reduce* qui seront exécutées par la plateforme sur l'ensemble de ressources disponibles dans une grille de calcul. La distribution des données ainsi que le déploiement du code sur les nœuds disponibles demeurent sur l'entière responsabilité de la plateforme (contrairement à ce qui se passe avec d'autres APIs de calcul réparti comme MPI⁸ et OpenMP⁹). Le concepteur n'a alors plus besoin de s'occuper de ces aspects, son code ainsi que ses données sont automatiquement répartis dans la grille de calcul par la plateforme elle-même. Cette facilité d'usage lui a valu un franc succès qui a fait de cette plateforme une de plus utilisées pour le développement d'applications *Big Data*. Aujourd'hui, même si l'usage de Hadoop s'est nettement réduit, cédant la place à des nouvelles plateformes comme Spark¹⁰ ou Mahout¹¹, entre autres, Hadoop reste la base de tout un écosystème *open source* dédié aux applications *Big Data*, comme l'illustre la Figure 22.

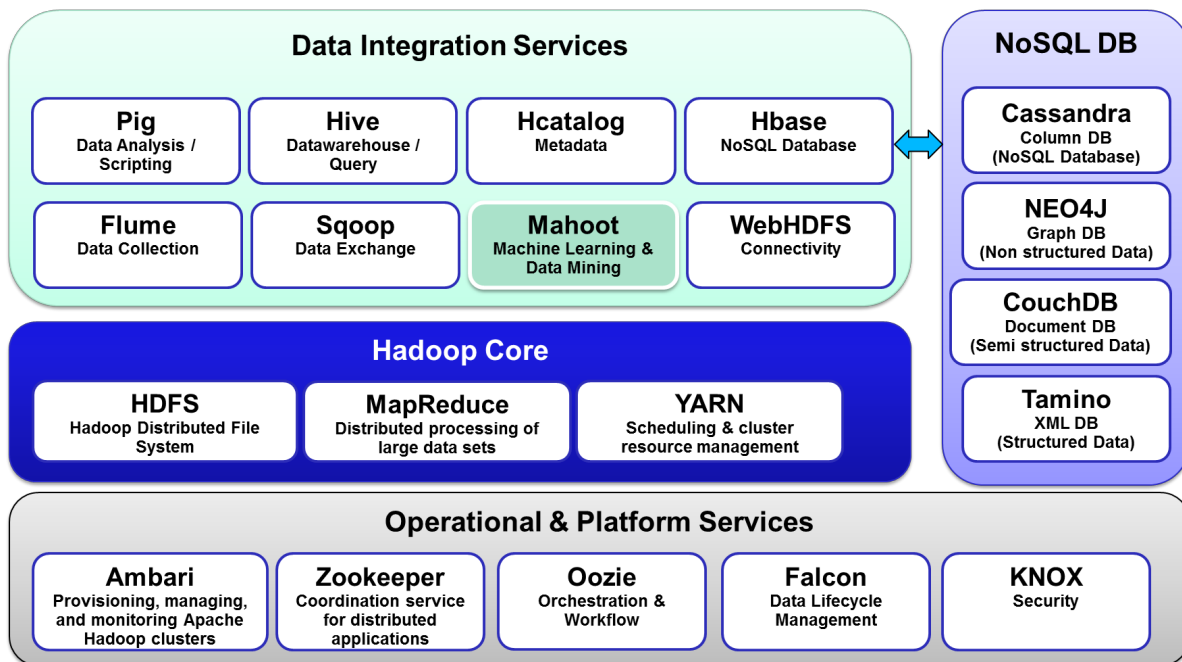


Figure 22. Ensemble d'outils pour le Big Data basés sur le framework Hadoop (illustration d'après [12]).

Apache Hadoop a été conçu pour s'exécuter sur des grilles de calcul « traditionnelles », autrement dit, des environnements de calcul homogènes, comptant un nombre stable de ressources disponibles. Cette plateforme n'a pas été conçue à la base pour s'exécuter sur des environnements hétérogènes et

⁸ <https://www.open-mpi.org/>

⁹ <https://www.openmp.org>

¹⁰ <https://spark.apache.org>

¹¹ <https://mahout.apache.org>

dynamiques (c'est-à-dire, des environnements dont la composition et l'état varient dans le temps), tels que les grilles pervasives. Selon Hagrais [100], la nature dynamique et ad-hoc des environnements pervasifs demande une certaine adaptation à des conditions d'opération changeantes et aux variations sur les préférences et le comportement des utilisateurs afin de favoriser un fonctionnement plus efficace et effectif, tout en prévenant des défaillances dans le système. Or la plateforme Hadoop, dans sa configuration originale, est incapable d'adapter dynamiquement son fonctionnement à un environnement variable, dans lequel les ressources sont de nature hétérogène et peuvent entrer ou sortir de l'environnement à tout moment. L'exécution de la plateforme Hadoop dans ce genre d'environnement va ainsi amener à un certain nombre d'inconvénients, liés notamment à l'allocation de ressources, entraînant le plus souvent une performance dégradée.

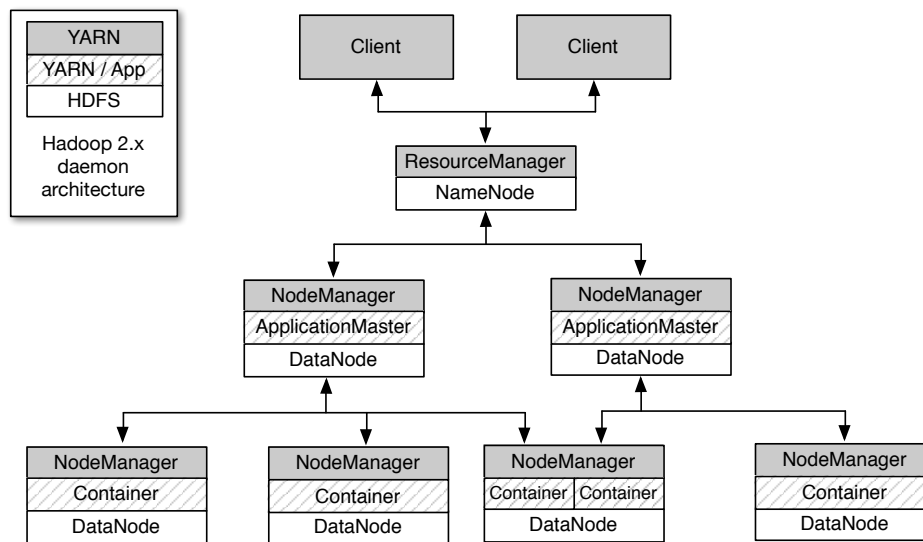


Figure 23. Overview de l'architecture Hadoop (d'après [45])

Une des raisons pour expliquer cette perte de performance repose sur la structure même de la plateforme. Celle-ci s'organise en deux parties, YARN¹², responsable de la gestion de ressources disponibles dans la grille de calcul, et HDFS¹³, responsable de la distribution des données dans la grille. Tous les deux respectent une architecture « maître-esclave », dans laquelle un nœud « maître » (le *ResourceManager* pour le premier, le *NameNode* pour le second) gère un ensemble de nœuds « esclaves » (les *NodeManager* pour YARN, et les *DataNode* pour HDFS, comme on peut voir sur la Figure 23). Le *NameNode* se charge de répliquer sur l'ensemble des nœuds les données soumises par une application à la plateforme, alors que le *ResourceManager* va s'occuper de l'exécution des tâches *map* et *reduce* soumises par l'application. Une même fonction *map* (ou *reduce*) sera ainsi lancée par le *ResourceManager* à chaque ressource (nœud) disponible sur un bloc distinct des données. Dans chaque nœud (cf. Figure 23), le *NodeManager* va assurer l'exécution de l'application (organisée par la plateforme en *ApplicationMaster* et *container*), alors que le *DataNode* va gérer les données présentes localement (aussi bien celles produites localement que les répliques des données distantes).

Le traitement des données est ainsi réparti entre l'ensemble de ressources. Chaque ressource est configurée de manière statique, à l'aide d'un fichier XML et le *ResourceManager* considère que les tâches (correspondant aux fonctions *map* et *reduce*) auront toutes à peu près le même temps

¹² <https://hadoop.apache.org/docs/stable/hadoop-yarn/hadoop-yarn-site/YARN.html>

¹³ <https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/HdfsDesign.html>

d'exécution quelque soit le nœud. Si une tâche prend plus de temps que les autres, le *ResourceManager* pourra déclencher une tâche spéculative, c.a.d. ordonner la réexécution d'une même tâche (*map* ou *reduce*) sur le même ensemble de données, mais sur un autre nœud. Or en présence des ressources hétérogènes, non seulement certaines tâches peuvent présenter des performances inférieures à d'autres (à cause de l'hétérogénéité des nœuds), mais également le nombre des tâches spéculatives peut augmenter de manière significative, dégradant ainsi la performance globale de l'application et consommant des ressources inutilement.

Cette possible perte de performance peut être observée à la Figure 24, issue de [46], dans laquelle on compare l'exécution d'une même application dans un environnement dédié (cas A) et dans un environnement hétérogène (cas B), où les capacités des nœuds impliqués évoluent pendant l'exécution des tâches. Une telle évolution peut s'expliquer, par exemple, par l'usage de ressources partagées avec d'autres applications et non par un usage exclusif de la grille de calcul. Dans la Figure 24, le cas A illustre une exécution dans un environnement homogène et dédié, correspondant donc à une grille de calcul traditionnelle. Par contre, dans le cas B, on simule une grille pervasive en réduisant la mémoire disponible de certains nœuds pendant l'exécution. La perte de performance est très significative, accompagnée d'une augmentation du nombre de tâches en exécution (représentée par l'intensité du dégradé dans le graphique), ce qui peut être expliqué par l'augmentation du nombre de tâches spéculatives.

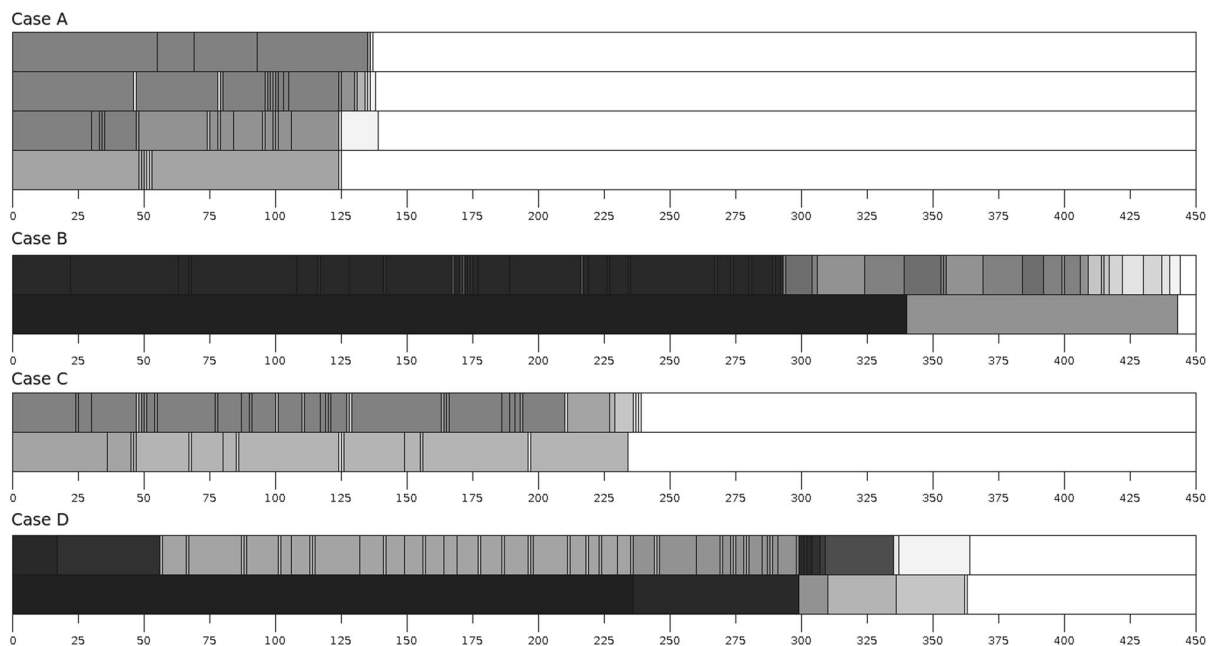


Figure 24. Scénarios d'exécution d'un benchmark Big Data (application TestDFSIO) dans un cluster homogène dédié (cas A) et dans un environnement pervasif, sans prise en compte du contexte (cas B) et avec prise en compte du contexte d'exécution (cas C et D) [46].

Afin de pallier ce problème, nous avons proposé, au sein du projet PER-MARE, d'adapter la plateforme Hadoop pour qu'elle puisse prendre en considération ces environnements hétérogènes. Nous avons ainsi modifié le *ResourceManager* et le *NodeManager* propres à Hadoop (cf. Figure 23) pour qu'ils puissent capturer des informations sur le contexte d'exécution des nœuds (nombre de cœurs disponibles, mémoire disponible, etc.). Pour cela, nous avons intégré le *ContextCollector* illustré par la Figure 25 au *NodeManager* pour que celui-ci puisse observer la situation courante de chaque nœud à la place de la configuration statique originellement proposée. Cet élément *ContextCollector* a été inspiré par nos travaux précédents sur la modélisation de contexte [137] (voir chapitre 1). Il considère

l'information de contexte comme étant observée à travers des éléments de contexte auxquels on associe un ensemble de valeurs dans un modèle orienté objets. En outre, chaque élément de contexte est associé à un concept précis d'une ontologie de contexte, telle que proposée par [170] (voir chapitre 3). L'intérêt de combiner, dans la modélisation de contexte, une approche orientée objets avec une approche par ontologie réside dans la possibilité d'avoir une description sémantique des éléments de contexte tout en gardant les avantages d'une implémentation orientée objet. Dans le cas du projet PER-MARE, le caractère léger de cette implémentation est particulièrement important puisque, s'agissant d'applications de type « haute performance » (*i.e.* cherchant atteindre des hauts niveaux de performance lors de leur exécution), la gestion de contexte ne devrait pas avoir un impact significatif sur la performance de l'application. L'approche mixte utilisée par le *ContextCollector*, inspirée également par les travaux du projet MUSIC [262], permet de garder une implémentation orientée objets légère, plus performante, tout en gardant la possibilité d'utiliser, à l'avenir, des mécanismes de raisonnement basés sur les ontologies.

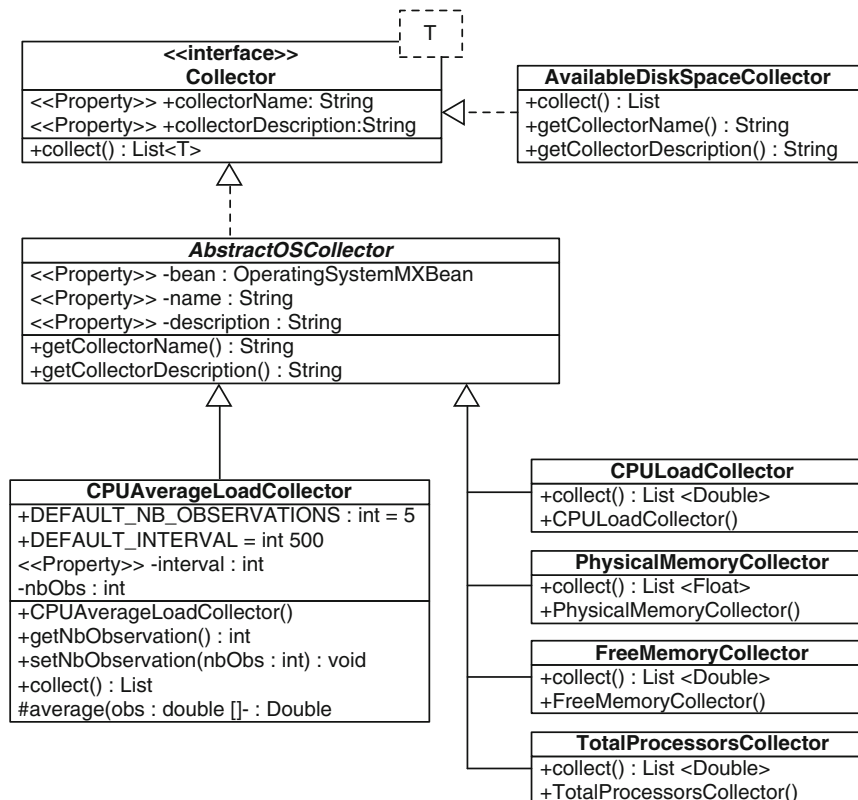


Figure 25. ContextCollector responsable par l'observation du contexte d'exécution d'un nœud [45].

Ainsi, dans un premier moment [45], nous utilisons l'information de contexte capturée par le *ContextCollector* pour remplacer le fichier de configuration statique utilisé par Hadoop, permettant une configuration automatique de chaque nœud à partir sur des informations réelles, collectées directement par le *ContextCollector*. Puis, nous avons modifié le *ResourceManager* [46] pour qu'il puisse utiliser ces informations de contexte dans son algorithme de *scheduling* (ordonnanceur). Celui-ci utilise alors des informations de contexte obtenues en temps réel pour ses décisions concernant la distribution des tâches de calcul. Les résultats obtenus [45, 46] avec différentes applications (des benchmarks exigeant différents types de ressources : CPU, mémoire et I/O) démontrent un gain de performance significatif par rapport à un environnement hétérogène où la plateforme n'a pas été adaptée. La Figure 24 illustre un de ces résultats pour l'application *TestDFSIO* (benchmark particulièrement gourmand en I/O). On peut observer une réduction significative du temps d'exécution entre le cas B et les cas C et D, où le *ContextCollector* a été utilisé dès le début de l'exécution (cas C) ou à mi-parcours (cas D).

Enfin, il est important de souligner que cette implémentation présente un impact minimal sur la structure interne de la plateforme Hadoop, notamment en ce qui concerne son *scheduler* (ordonnanceur). En effet, aussi bien l'architecture originelle (représentée à la Figure 23) que le *scheduler* originel utilisé par défaut sur la plateforme, le *Capacity Scheduler*, ont été maintenus inaltérés. Seules les données d'entrée ont été modifiées, remplacées par des données capturées dynamiquement par le *ContextCollector* sur chaque nœud. Aucune modification ou profilage des applications n'a été nécessaire non plus, contrairement à d'autres travaux sur la littérature, tels que [48, 57, 149, 270], lesquels proposent des modifications significatives sur la plateforme ou sur le *scheduler*, ou encore demandent qu'un profilage des applications à exécuter soit réalisé au préalable.

2.2.2.2 Usage opportuniste de ressources dans une plateforme de Fog/Edge Computing

Le terme *Fog Computing* exprime l'idée de disposer des services directement autour des utilisateurs et des sources de données, en opposition aux ressources distantes présentes dans les plateformes *cloud* [28]. Différents termes peuvent être associés à cette idée de disposer des ressources en « bordure » (« *edge* ») du réseau pour la mise en place de services au plus proche des utilisateurs. On peut parler de *Edge Computing* [187], *Mobile Cloud Computing* [110], *cloudlets* [222], *Osmotic Computing* [261] et bien évidemment, grilles pervasives [192]. Tous ces termes partagent cette idée de valoriser les ressources à proximité en les utilisant afin de mieux satisfaire certaines exigences des applications comme un meilleur temps de réponse, une latence réduite ou le maintien de la confidentialité des données traitées. Le *Fog Computing* se présente ainsi comme une alternative, voir un complément, aux plateformes *cloud*, notamment publiques, ou à une vision dans laquelle ces plateformes seraient utilisées en exclusivité. Le principe est assez simple : utiliser les capacités de calcul et de stockage des ressources en bordure du réseaux (et donc à proximité des utilisateurs finaux et des sources des données) pour la réalisation de tâches habituellement déléguées à des infrastructures de type *cloud*.

Ce principe de garder les données au plus proche de leur lieu de production (au lieu de tout envoyer à une infrastructure de calcul distante) peut être particulièrement bénéfique pour les applications de type *big data*, au vu des volumes de données habituellement impliquées. En effet, le transfert d'importantes masses de données vers une infrastructure *cloud* peut conduire à une surconsommation de la bande passante et, par conséquent, à des délais non-négligeables, qui peuvent, à leur tour, ralentir le traitement de ces données et potentiellement toute prise de décision basée sur celles-ci. Selon Huang & Wu [110], garder des données en local (c.à.d. à proximité de leur site de production) offre deux bénéfices majeurs : (i) une réduction dans le trafic réseau (notamment pour les réseaux sans fil) ; et (ii) une certaine facilité dans le chargement de données vers les plateformes *cloud*, vu que celles-ci ne seront plus les données brutes qui y seront transférées, mais uniquement les données après un premier traitement à proximité.

Dans le cas du projet PER-MARE, nous avons voulu proposer une alternative aux *clusters* Hadoop à travers une plateforme de *Fog Computing* indépendante. Pour cela, nous avons proposé la plateforme CloudFIT, développée par l'Université Reims Champagne Ardennes [236]. A travers cette plateforme, l'objectif était surtout de pouvoir considérer l'usage des ressources disponibles sur l'environnement pour l'exécution d'applications de type *Big Data*, sans avoir recours à une plateforme lourde tel que la plateforme Hadoop.

Néanmoins, l'usage de ressources en bordure du réseau pour la réalisation de tâches de calcul, aussi simples soient elles, présente quelques inconvénients. Tout d'abord, la connectivité de ces nœuds ne peut pas être garantie, ils ne sont pas aussi fiables qu'un serveur sur le *cloud* [101]. Par ailleurs, ces ressources se retrouvent souvent dispersées sur des supports personnels ou véhiculaires, par conséquent, hétérogènes et connectées par intermittence [110]. Il s'agit donc des ressources hétérogènes et volatiles, à l'instar des ressources d'une grille pervasive. Selon Coronato & De Pietro [62], les grilles pervasives doivent être capables de s'auto-adapter et s'auto-configurer afin d'accueillir

des dispositifs mobiles. Dans ces environnements, le challenge ne se limite donc pas à l'hétérogénéité des dispositifs disponibles, mais également la volatilité de ces dispositifs. Ceux-ci n'étant pas dédiés aux tâches de calcul, ils peuvent facilement se connecter et se déconnecter du réseau au gré de leurs déplacements (ou de ceux de leurs propriétaires), de la disponibilité du réseau, ou encore de l'état de leur alimentation. Un dispositif peut ainsi à tout moment « apparaître » sur l'environnement, devenant ainsi disponible pour les tâches de calcul, et tout aussi soudainement disparaître, devenant inaccessible de manière temporaire ou permanente. Alors que dans la section précédente (Section 2.2.2.1), l'objectif principal était de traiter l'hétérogénéité des ressources sur la plateforme Hadoop, l'objectif ici était surtout de gérer la dynamique de l'environnement et la volatilité de ses ressources, dans une plateforme *fog*.

Ainsi, la plateforme CloudFIT [236] a été conçue pendant le projet PER-MARE afin de permettre l'exécution d'applications Java, et notamment de type *Big Data* visées par le projet, utilisant des ressources disponibles en bordure du réseau. Pour cela, la plateforme repose sur un réseau pair-à-pair (P2P) dans lequel les dispositifs participants (aussi appelés nœuds) se partagent les tâches à exécuter et les données. La gestion de la volatilité de ces dispositifs se fait à la fois par l'usage du réseau P2P (et plus précisément d'un *overlay network* P2P), et par le caractère réparti de la gestion de tâches. En effet, les nœuds disponibles forment une communauté (semblable aux groupes mentionnés dans la section 2.1), représentée dans la Figure 26. Cette communauté partage un ensemble de données à travers une DHT (*Distributed Hash Table*) et un ensemble de tâches à réaliser (organisées sous la forme d'une « *bag of tasks* »). Chaque nouveau nœud souhaitant participer à l'effort de calcul peut alors demander à rejoindre la communauté. Il peut faire cette demande à n'importe quel autre nœud appartenant à la communauté. Dès qu'il l'intègre, il reçoit des autres nœuds la liste de tâches à réaliser et il devient candidat à stocker des données (des données qu'il produit lui-même ou des répliques des données présentes sur les autres nœuds).

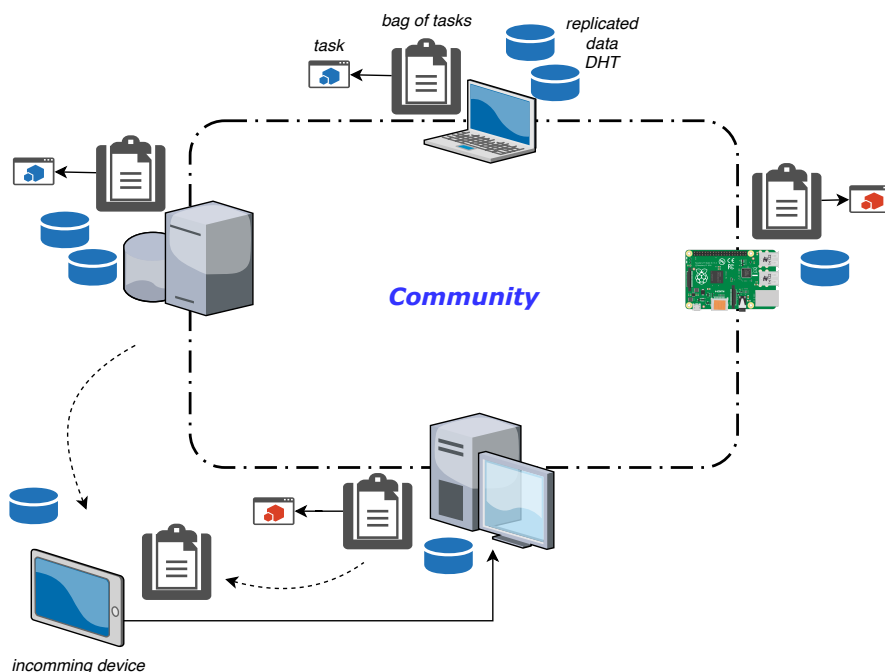


Figure 26. Représentation de l'entrée d'un nouveau nœud sur une communauté CloudFit.

Il est à noter une distinction entre les tâches à exécuter dans CloudFIT. Celles-ci s'organisent de manière hiérarchique en deux niveaux d'abstraction : les *jobs* et les *tasks*. Les *jobs* sont des ensembles de *tasks* à exécuter, alors que celles-ci (*tasks*) représentent les unités de calcul à réaliser. Ainsi, une

application de type *MapReduce* peut être vue comme l'enchaînement de deux *jobs*, le *Map* et le *Reduce*, chacun composé de plusieurs *tasks* (par exemple, la réalisation de la fonction *Map* sur un bloc de données précis). Cette organisation hiérarchique permet d'exprimer aisément le parallélisme intra et inter *job* ainsi que les dépendances à ces deux niveaux : des dépendances entre *jobs* distincts (e.g. un *job Reduce* ne peut démarrer qu'après que le *job Map* soit terminé) et des dépendances entre les *tasks* d'un même *job* (une *task* qui doit s'exécuter avant ou après une autre *task*).

A partir du moment qu'un nœud intègre la communauté, il a accès à l'ensemble de *jobs/tasks* qui restent à réaliser. Il peut également ajouter de nouveaux *jobs* (et par conséquent ses *tasks*) à la liste. Ceux-ci seront retransmis aux autres membres de la communauté qui pourront alors contribuer à son exécution de manière collaborative. Chaque nœud peut en effet choisir des *tasks* à réaliser parmi les *jobs* incomplets (c.a.d. pas encore finalisés). Il choisit la *task* de manière aléatoire et se met immédiatement à l'exécuter. Dès qu'il la termine, il en informe les autres membres de la communauté qui suppriment alors la *task* de leur propre liste. Les résultats obtenus sont également répartis dans la communauté par la DHT.

C'est cette distribution des tâches (*jobs & tasks*) et des données (à travers la DHT) qui permet de gérer la volatilité de l'environnement. D'une part, si un nœud disparaît ou simplement quitte la communauté (e.g. il se déconnecte du réseau) avant d'avoir terminé l'exécution d'une *task*, les autres nœuds peuvent encore exécuter la même *task*, puisqu'elle se retrouve toujours dans leur liste de tâches à réaliser. D'autre part, la réplication des données par la DHT assure que les résultats déjà obtenus par un nœud ne soient pas perdus à son départ, puisqu'ils seront répliqués sur d'autres nœuds de la communauté. Les nœuds peuvent ainsi intégrer ou quitter la communauté sans que cela implique l'arrêt de l'application ou la perte des données ou d'une tâche de calcul.

La réplication des données avec l'usage d'une DHT nous a permis aussi de mieux gérer la manipulation de grands volumes de données nécessaires à une application de type *Big Data*. A l'instar de ce qui est fait par Hadoop, les données sont réparties avec un certain nombre de répliques disponibles pour éviter qu'elles soient perdues en cas de déconnexion ou de panne. Mais contrairement à Hadoop, cette réplication n'est pas gérée par un nœud central (le *NameNode*). Par ailleurs, il est à observer que, comme pour Hadoop, cette réplique n'est pas « totale », c'est-à-dire avec chaque donnée répliquée sur chaque nœud disponible. Les nœuds étant hétérogènes, ils ne disposent pas tous de la même capacité de stockage. Ils ne disposeront donc pas de toutes les données disponibles, mais uniquement d'un sous-ensemble, selon la distribution établie par la DHT.

Cet aspect est particulièrement important dans le cadre du projet PER-MARE car il nous a permis de minimiser les problèmes liés aux transferts des données tout en permettant la volatilité des nœuds dans un environnement pervasif. En effet, les premières expériences que nous avons mené dans le cadre de ce projet nous ont démontré l'impact important de ces transferts sur ces environnements. Lors de nos premières expériences, nous avons utilisé une autre plateforme, CloudFIT n'étant pas encore disponible. Il s'agissait alors d'une bibliothèque de calcul réparti nommée CONFIT [86], également proposée par l'Université de Reims Champagne Ardenne. Celle-ci utilisait une topologie en anneau dans laquelle les données étaient répliquées sur chaque nœud participant à l'anneau. Nos premières expériences ont ainsi démontré que, dans le cadre d'une application de type *Big Data*, une telle réplication impliquait un important transfert des données sur le réseau, conduisant rapidement à une surcharge de celui-ci (surtout dans le cas de réseaux sans fil) et, par conséquent, à une perte de performance importante.

Dans [235], nous avons ainsi comparé les temps d'exécution obtenus par un benchmark (l'application WordCount, très utilisée pour illustrer les plateformes *Big Data*) avec la bibliothèque CONFIT et avec la plateforme Hadoop (cf. Figure 27). La Figure 27 illustre les résultats obtenus avec l'exécution de ce benchmark sur Hadoop et sur CONFIT utilisant différentes tailles de block de données (plusieurs blocks de 512Ko, plusieurs blocks de 1Mo et un seul block contenant la totalité des données). Sans surprise la plateforme Hadoop surclasse toutes les exécutions avec la bibliothèque CONFIT dès que le volume

de données dépasse 8Mo, quelle que soit la taille de bloc utilisée. Cependant, ce qui frappe sur la Figure 27 est la dégradation très rapide et très significative des performances à partir de ce palier avec la bibliothèque CONFIIT, et ceci alors que l'environnement utilisé correspondait à celui d'un cluster de calcul traditionnel (deux machines AMD Opteron 275 de 2.2 GHz comptant 4 cœurs chaque, interconnectés par un réseau Gigabit Ethernet). Quelle que soit la taille du bloc utilisée, les pertes ont été extrêmement importantes. En étudiant ce phénomène, nous avons identifié plusieurs causes possibles dont la principale était la surcharge entraînée par la communication et la réplication des données entre les nœuds.

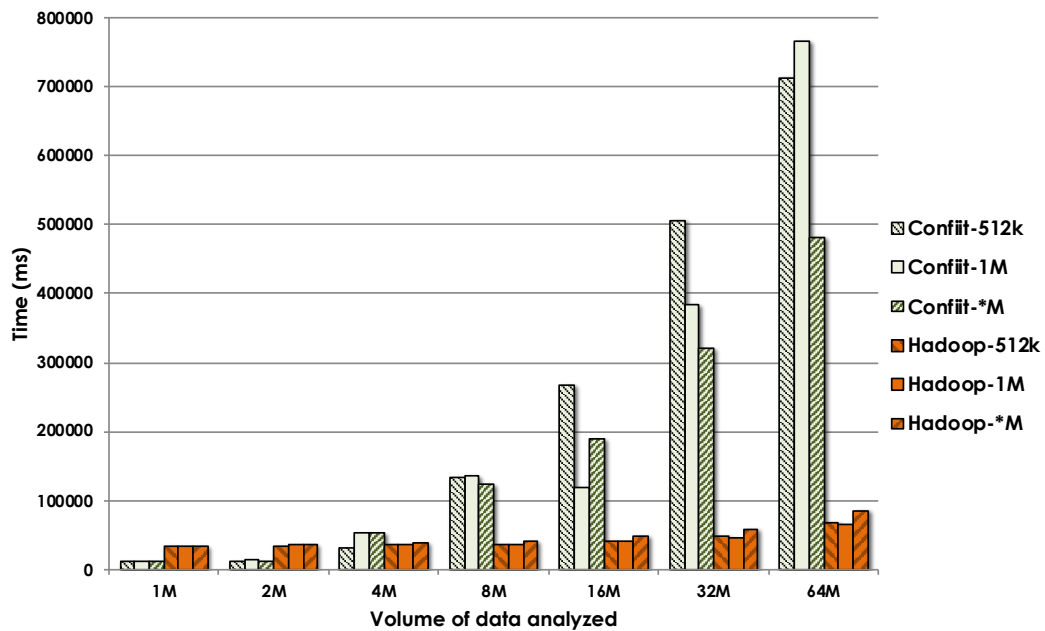


Figure 27. Comparaison entre les temps d'exécution du benchmark WorldCount avec la plateforme Hadoop et la bibliothèque de calcul réparti CONFIIT [235].

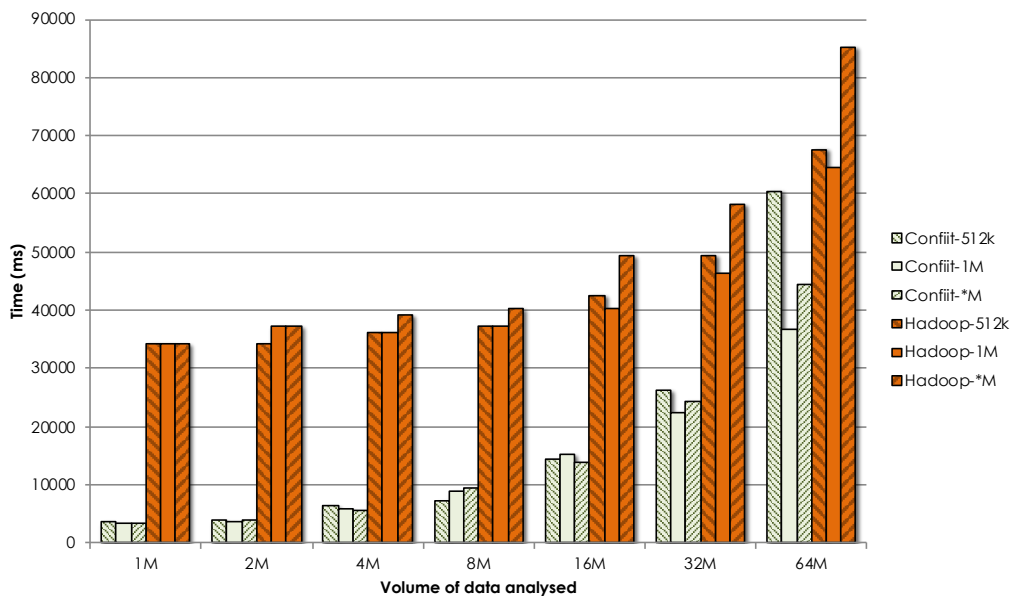


Figure 28. Comparaison des temps d'exécution entre Hadoop et CONFIIT modifié avec l'usage de la bibliothèque P2P FreePastry [234].

Cette première expérience nous a incité à utiliser un réseau pair à pair afin de garantir la volatilité des dispositifs, mais elle nous a également incité à repenser la distribution des données entre les nœuds. En effet, une structure pair-à-pair, avec un groupe de nœuds coopérant pour assurer l'exécution d'un ensemble de tâches, nous a semblé bien plus appropriée aux environnements pervasifs puisqu'elle se passe de la présence d'un nœud « master » ou d'un élément centralisé, comme c'est le cas dans Hadoop. Il nous a semblé important de pouvoir trouver un équilibre pour réduire la réplication des données entre les nœuds tout en minimisant les risques de perte de données, afin de réduire les efforts de communication nécessaires pour le maintien de la communauté. Ces hypothèses de travail ont ainsi été vérifiées lors d'une deuxième expérience (détaillée dans [234]), dans laquelle nous avons modifié l'application utilisant CONFIIT afin qu'elle puisse utiliser un réseau P2P (implémenté avec la bibliothèque FreePastry¹⁴). Cette nouvelle implémentation limitait les échanges entre les nœuds, et on aperçoit sur les résultats illustrés par la Figure 28 un gain de performance important alors que l'environnement d'exécution utilisé pour le test est resté sensiblement le même. Ces nouvelles performances ont été atteintes notamment grâce à la réduction de la surcharge de communication entre les nœuds, dont le protocole et le volume de messages échangés ont été sensiblement réduits.

Ces expériences nous ont également mis en garde sur les risques de saturation du réseau et des nœuds concernés dans le cas d'applications de type *Big Data*. Le fait de considérer un environnement pervasif, en bordure du réseau, avec des ressources hétérogènes et surtout pas dédiés au calcul, fait que cette saturation puisse se faire sentir de manière bien plus importante que dans le cadre d'un *cluster* HPC dédié, comme celui utilisé pour ces expériences, où l'infrastructure réseau est pensée pour supporter la communication à très haut débit entre les nœuds, eux-mêmes taillés pour la haute performance. La Figure 29 propose un bon aperçu de l'effet que la qualité de l'infrastructure peut avoir sur la performance d'une application. Cette image illustre les résultats obtenus [233] par une même application (le benchmark WordCount), avec les mêmes volumes de données (un corpus de texte issu du projet Gutenberg¹⁵) sur une infrastructure stable (trois nœuds sur une grille de calcul), comportant des dispositifs taillés pour l'haute performance (nœud « Romeo » au Tableau 4) et une grille pervasive, composée des nœuds hétérogènes (voir Tableau 4), connectés par un réseau WiFi 802.11g (par opposition au Infiniband de 40Gbps utilisé dans le *cluster*) et partagés avec d'autres applications (comme, par exemple, l'anti-virus). Bien évidemment le *cluster* dédié obtient des performances nettement supérieures, mais surtout, on voit que la grille pervasive est bien plus sensible à l'augmentation dans le volume de données manipulées. L'analyse de ces résultats démontre qu'en outre l'hétérogénéité des dispositifs impliqués, le réseau influence de manière significative les performances, surtout sur ces environnements en « bordure » du réseau, qui peuvent se retrouver facilement saturés par la communication de données nécessaires aux applications de type *Big Data*. L'optimisation de ces communications devient ainsi un point clé pour la réussite de ces applications.

¹⁴ <https://www.freepastry.org>

¹⁵ <https://www.gutenberg.org>

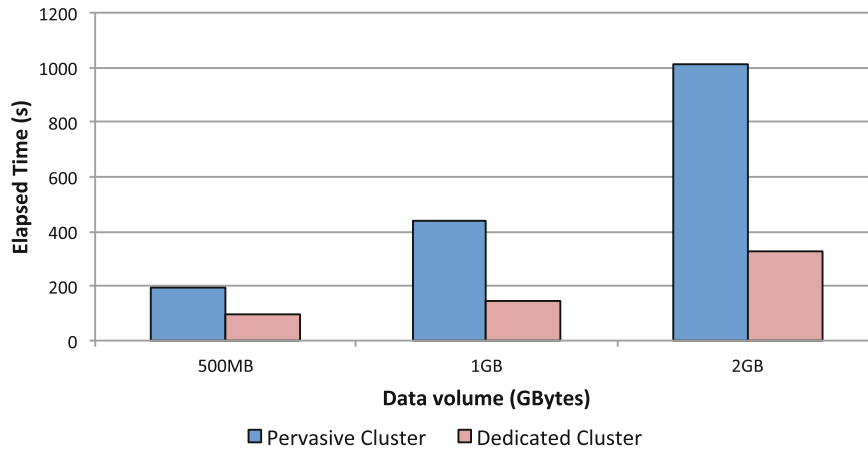


Figure 29. Comparaison entre l'exécution d'une même application dans un cluster dédié et dans une grille pervasive [233].

Tableau 4. Indications sur les dispositifs utilisés lors des expérimentations.

Dispositif	Processeur	GHz	Cœurs	Mémoire	Système d'exploitation
MacBook Air	Intel Core i7-4650U	1,7	2	8 Go	Mac OS 10.10.3
HP Pavillon dv6	Intel Core i5-2450M	2,5	2	8 Go	Windows 7
Lenovo U110	Intel Core 2 Duo L7500	1,6	2	4 Go	Ubuntu Linux 15.4
Romeo	Bi-Intel Xeon E5-2650	2,6	8	32 Go	RedHat Enterprise Linux

Ces premiers résultats ont ainsi motivé la création de la plateforme CloudFIT. En effet, la bibliothèque CONFIIT présentait plusieurs limitations que rendaient difficile son utilisation pour le projet PER-MARE. Tout d'abord, elle était conçue pour exécuter uniquement des tâches indépendantes (*i.e.* sans aucune dépendance entre les tâches), ce qui n'est pas vraiment le cas d'applications de type *MapReduce*, visées par le projet PER-MARE, où les tâche *Reduce* dépendent des résultats des tâches *Map*. Puis l'architecture logicielle interne à cette bibliothèque ne nous permettrait pas de la faire évoluer aisément, conduisant ainsi à la définition d'une nouvelle architecture (voir Figure 30), celle-ci évolutive, pour la plateforme CloudFIT. Celle-ci a été conçue comme une toute nouvelle plateforme permettant l'exécution des tâches de calcul, peu importe leur nature (dépendante ou indépendante, pour des applications de type *Big Data* ou pour le calcul intensif, etc.), sur un environnement de *fog computing*. Cette nouvelle architecture adoptée par la plateforme CloudFIT (cf. Figure 30) repose sur une séparation claire entre les responsabilités de chaque composant : la gestion de tous les aspects communication est déléguée à l'ORB (*message broker*), inspiré des *middleware* CORBA ; la gestion de la communauté (et avec elle, des listes de *jobs/tasks* à réaliser propre à chaque communauté) est déléguée au *community manager* ; alors que la gestion des ressources et l'allocation des tâches est contrôlée par le *resource manager* (le *context manager* et son influence seront détaillés au chapitre 4). Cette séparation permet ainsi l'évolution de chaque élément de manière indépendante, sans que ça puisse affecter de manière significative les autres composants.

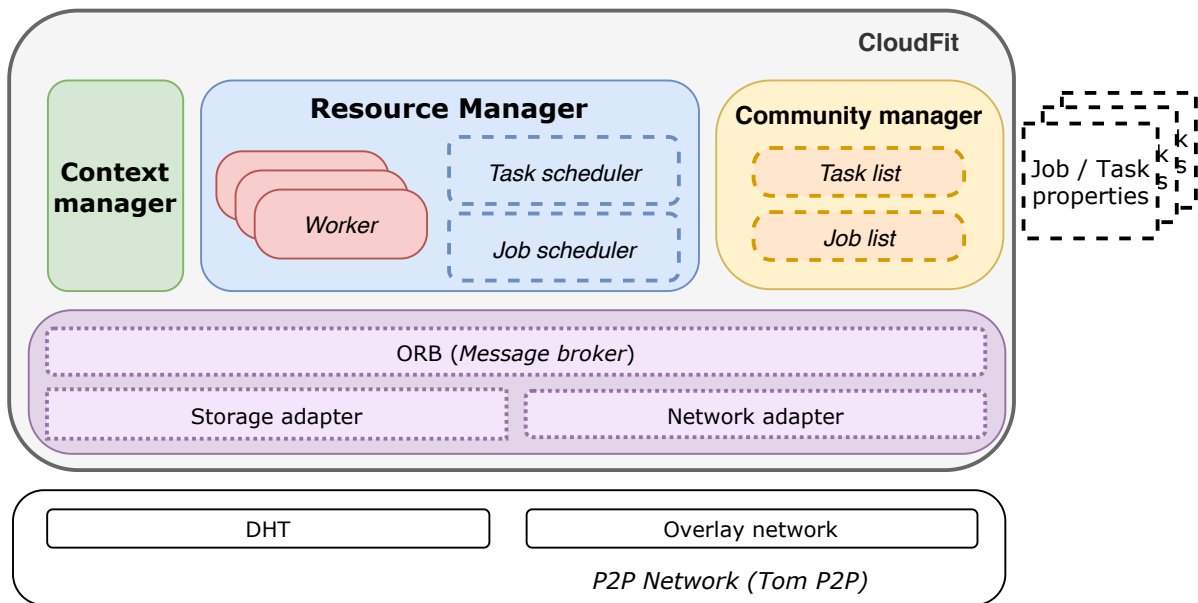


Figure 30. Architecture logicielle de la plateforme CloudFIT.

La gestion de la volatilité de l'environnement est ainsi assurée notamment par l'ORB, mais pas uniquement. Le *resource manager* va également contribuer à cette gestion à travers sa politique de *scheduling* répartie. En effet, dans le projet PER-MARE nous avons testé une politique de gestion de tâches répartie, dans laquelle chaque nœud appartenant à la communauté décide de manière indépendante quelle tâche exécuter parmi les tâches inachevées. La coordination entre les nœuds est ainsi minimale, puisqu'ils ne communiquent que les résultats obtenus une fois la tâche achevée. Les membres de la communauté partagent les tâches à exécuter mais chacun décide seul laquelle exécuter en priorité. Ainsi, le départ d'un nœud de la communauté ne compromet pas la réalisation d'un *job*, puisque les autres nœuds continuent à exécuter les *tasks* appartenant à ce *job*. Inversement, l'intégration d'un nouveau nœud à la communauté et le partage des tâches qui en découle permettent à celui-ci de rapidement prendre part à l'exécution. La Figure 31 illustre ce scénario dans lequel un premier nœud (id n° 3) quitte la communauté, puis un nouveau nœud (id n° 5) l'intègre.

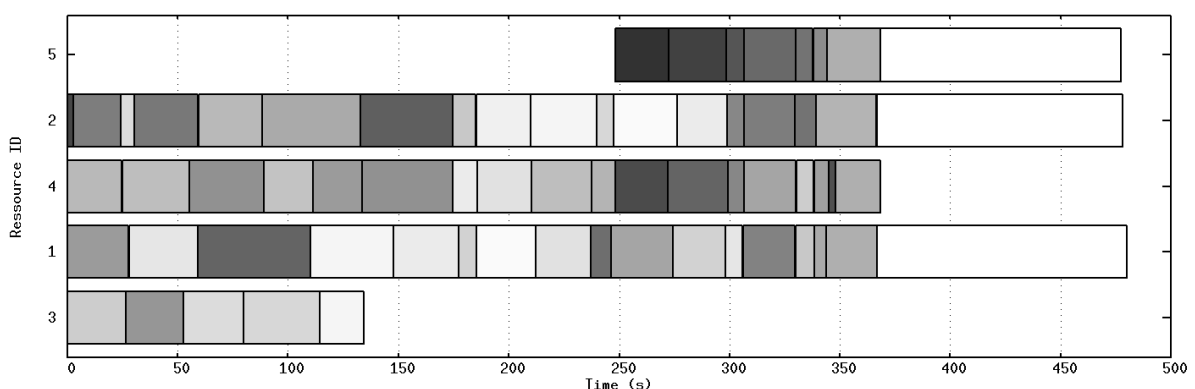


Figure 31. Exécution d'une application sur une grille pervasive, avec un nœud qui quitte la communauté et un autre qui l'intègre à posteriori [236].

Cette politique de *scheduling* plutôt naïve, où chaque nœud choisit de manière aléatoire la prochaine tâche à exécuter, nous a permis de mieux étudier les effets de l'hétérogénéité de l'environnement sur l'exécution des *jobs*. Nous avons réalisé plusieurs tests dans lesquels des dispositifs très hétérogènes intègrent la communauté et nous avons observé l'effet de cette hétérogénéité sur l'exécution, comme l'illustre la Figure 32. Dans cette expérimentation, nous avons exécuté l'application WordCount sur un

ensemble de quatre nœuds distincts : un RaspberryPi¹⁶ (Pi) ; deux ordinateurs portables (un MacBook Air et un Lenovo U110, décrits dans la Tableau 4) ; ainsi qu’une machine virtuelle¹⁷ (VM) exécutant sur un des ordinateurs portables (le MacBook Air en occurrence). Les RaspberryPi ayant des capacités inférieures aux autres nœuds, on pouvait s’attendre qu’il exécute moins de tâches que ses voisins. Or même si ce nœud prend effectivement plus de temps dans l’exécution de chaque tâche (illustré dans la Figure 32 par la longueur de chaque carré), surtout en début d’exécution, on observe que le RaspberryPi continue l’exécution du *job* alors que les autres nœuds l’ont déjà terminé. En effet, nous avons pu constater sur ce type de nœud une certaine surcharge qui faisait que celui-ci perdait bien souvent des messages des nœuds voisins indiquant qu’une tâche a bien été finalisée. Sans ces messages de contrôle, la liste de *tasks* à exécuter n’était pas mise à jour et le *resource manager* continuait à lancer l’exécution des *tasks* déjà finalisées. Les résultats de celles-ci étant déjà disponibles sur la DHT, ces exécutions étaient vite avortées, ce qui explique la longueur assez réduite des tâches en fin d’exécution.

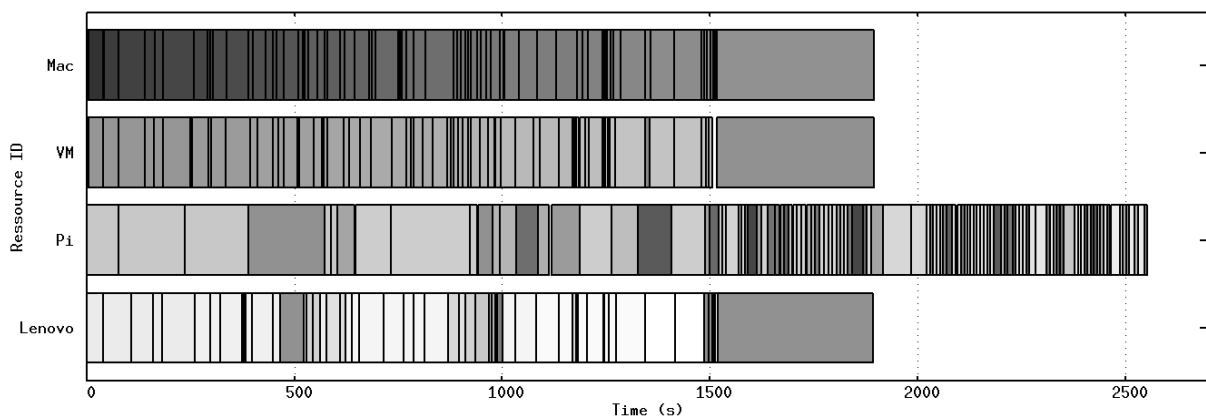


Figure 32. Exécution de l’application WordCount sur un environnement hétérogène [236].

Ces résultats démontrent l’intérêt d’un *scheduler* sensible au contexte, à l’instar de ce que nous avons fait avec la plateforme Hadoop (cf. section 2.2.2.1), afin de ne pas surcharger les nœuds aux capacités limitées. Nous avons ainsi continué le travail sur ces aspects essentiels pour les environnements de *fog computing* une fois le projet PER-MARE terminé. Nous discutons ces travaux réalisés à posteriori dans le chapitre 4.

2.2.3 Originalité & Impact

Le projet PER-MARE (2013-2014) s’est intéressé au domaine du *Big Data*, un domaine en pleine ascension au moment de la proposition de ce projet. La proposition centrale du projet d’utiliser les grilles pervasives pour l’exécution ces applications était alors très innovante pour l’époque, puisqu’à ce moment l’usage de plateformes dédiées (*cluster* ou *cloud*) était encore largement dominant, comme l’illustre le *Gartner Hype Cycle* de 2012 représenté dans la Figure 33. Il est important d’observer que les termes *fog/edge computing* étaient encore largement méconnus. Ces termes sont apparus autour de 2012 [28, 88, 58, 187] et ont gagné plus d’attention bien plus récemment, comme un complément au *cloud computing*, notamment dans le cadre de l’IoT et de l’analyse de données.

¹⁶ RaspberryPi 2, processeur Arm Cortex A7 de 0,9GHz, avec 1Go de mémoire vive et système d’exploitation Raspbian Wheezy.

¹⁷ Machine virtuelle exécutant le système d’exploitation Debian 8.2 sur une VirtualBox, comportant 1 seul cœur et 1Go de mémoire vive.

L'entrée du terme « *edge IA* » en référence à l'usage du *fog/edge computing* pour l'Intelligence Artificielle (IA) dans le Gartner Hype Cycle de 2019 (cf. Figure 34) illustre bien le gain d'intérêt de l'industrie par ce type de solution. Le scénario type considéré par beaucoup [28, 56, 58, 80, 166, 187, 220, 258, 261] est celui d'une production massive des données grâce à l'IIoT, données qui reçoivent un premier traitement à proximité du lieu de production, grâce aux plateformes de *fog computing*, avant leur envoi, déjà traitées, vers les plateformes de *cloud computing*. Ces traitements réalisés par les plateformes de *fog* permettraient ainsi de réduire le volume total de données envoyées vers le *cloud* ou encore de traiter des problèmes de sécurité ou de respect de la vie privée, par exemple, grâce à l'anonymisation des données *in situ*. Le projet PER-MARE se positionne ainsi comme un précurseur, proposant l'usage de ressources de proximité pour le traitement des données, issues ou non de l'IIoT.

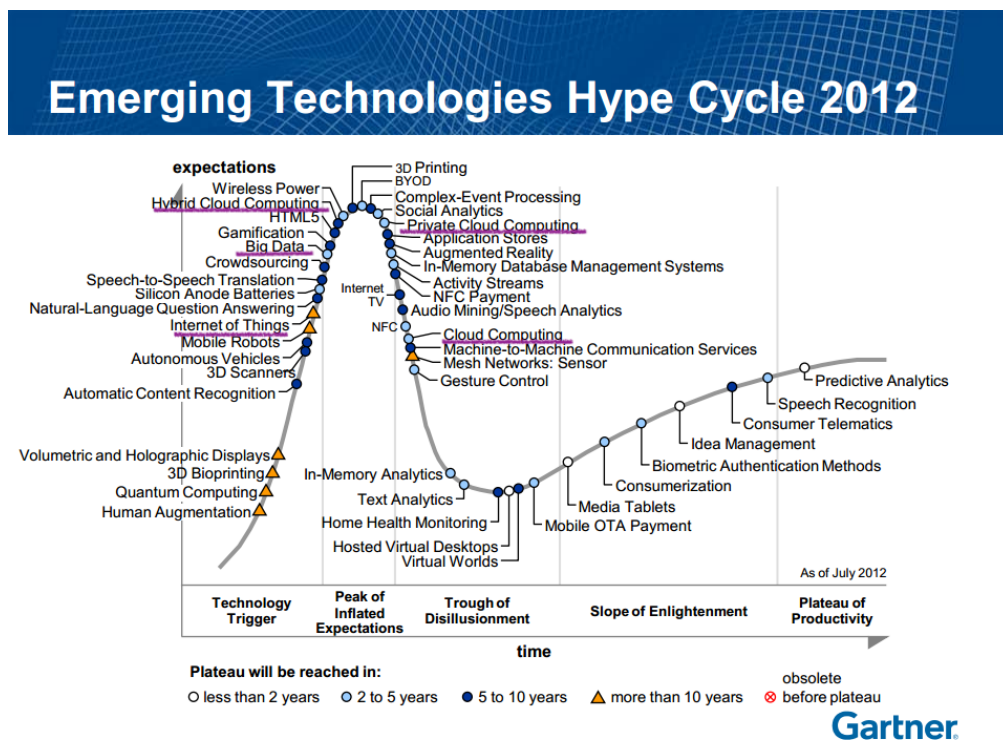


Figure 33. Gartner Hype Cycle de technologies émergentes pour l'année 2012¹⁸.

Aujourd'hui, avec le développement des solutions pour le *fog computing* [56, 80, 113, 187, 247, 258] et surtout le développement de cette vision du *fog* combiné au *cloud computing* pour le traitement de masses des données (potentiellement *Big Data*) produites par l'IIoT, les propositions du projet PER-MARE sont encore plus d'actualité, d'autant plus qu'elles demeurent innovantes par rapport à la littérature. En effet, la vision qui prédomine aujourd'hui est celle des plateformes *fog* comme une étape intermédiaire avant l'envoi des données sur les plateformes *cloud*. Dans cette vision dominante, le *fog computing* n'existe pas sans le *cloud computing*. Cette dépendance, observable dans plusieurs travaux (p.ex. [56, 261]), est soulignée par Alrawais et al. [8], pour qui l'objectif même du *fog computing* est de réduire le volume et le trafic des données vers les serveurs *cloud*, diminuant ainsi la latence et augmentant la qualité de service. Or dans le projet PER-MARE, on s'est affranchi, dès le départ, de cette contrainte. Nous n'avons pas considéré dans le projet la présence (ou plutôt la disponibilité) constante d'une plateforme dédiée, soit-elle un *cluster* ou un *cloud*, pour l'exécution des applications à un moment ou à un autre. L'objectif a toujours été un usage opportuniste des ressources

¹⁸ Source : <https://res.infoq.com/news/2012/08/Gartner-Hype-Cycle-2012/fr/resources/hype1.png>

disponibles quel qu'elles soient. En s'affranchissant de cette contrainte, nous avons pu considérer l'usage de n'importe quelle ressource à portée de main, soient-elles des ressources issues de l'IoT (comme les RaspberryPI utilisés dans nos expérimentations), des ordinateurs portables ou même des clusters ou des ressources sur le cloud.

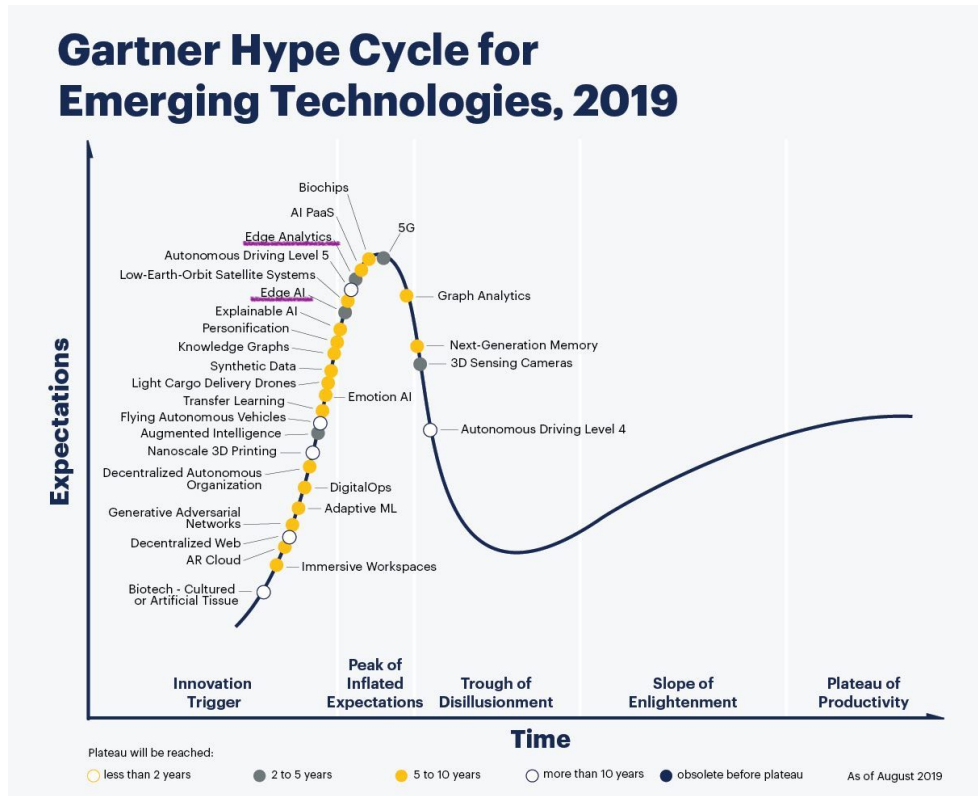


Figure 34. Gartner Hype Cycle de technologies émergentes pour l'année 2019¹⁹.

Autre point intéressant des contributions du projet PER-MARE est l'usage des informations de contexte pour le *scheduling* des tâches de calcul. Contrairement à d'autres propositions datant de la même époque (p.ex. [48, 57, 149]), les propositions réalisées dans le projet [45, 46] se sont avérées très peu intrusives, demandant un effort minimale de la part des développeurs. En effet, aucun profilage ou changement n'est nécessaire à ces applications (contrairement, par exemple, à [149]), permettant un usage transparent de la notion de contexte pour la gestion de l'environnement. Les résultats obtenus avec Hadoop et avec la plateforme CloudFIT [45, 46, 233, 236] démontrent l'intérêt de notre hypothèse de départ qui pointait l'importance de la prise en charge du contexte pour une meilleure prise en compte des environnements pervasifs. Les derniers résultats du projet [46, 236] démontrent bien l'effet de l'hétérogénéité de ces environnements dans les performances et nous ont poussé à continuer nos travaux dans cette voie, notamment dans le cadre des plateformes de *fog computing*. Dans la littérature aussi, le *scheduling* a bien été identifié comme un défi majeur pour la réussite de ces plateformes [94, 106], même si les travaux tenant compte des informations de contexte demeurent minoritaires [31, 79]. Nous abordons cette question en particulier dans le chapitre 4, dans lequel nous considérons, entre autres, l'usage des plateformes de *fog computing* dans les Systèmes d'Information (S.I.). En effet, ces travaux ouvrent des perspectives intéressantes pour les S.I. qui

¹⁹ Source : <https://www.gartner.com/smarterwithgartner/5-trends-appear-on-the-gartner-hype-cycle-for-emerging-technologies-2019/>

disposent déjà de nombreuses ressources disponibles en interne, en plus de celles pouvant être allouées sur les plateformes de *cloud* publiques. Il en résulte un environnement pervasif d'une extrême hétérogénéité pour lequel une gestion intelligente et opportuniste des ressources est encore nécessaire.

2.2.4 Bibliométrie

Les recherches autour du projet PER-MARE ont donné lieu à des nombreuses publications, qui se sont étalées de 2013 à 2016, bien au-delà de la durée officielle du projet (2013-2014). Dans cette section, nous essayons d'analyser l'impact de ces publications en termes de citations. Comme pour le chapitre précédent, nous avons considéré les citations visibles à partir de la plateforme scholar.google.fr, laquelle condense les informations issues des nombreuses autres sources (IEEEExplore, SpringerLink, ScienceDirect, arxiv, etc.), mais également à partir des plateformes www.researchgate.net et www.semanticscholar.org, ainsi que du site de l'éditeur le cas échéant. Les citations sont organisées en deux catégories : celles jusqu'à 2017 (datant donc des 5 premières années à partir du début du projet) ; et celles à partir de 2018. A ces catégories s'ajoutent les autocitations, qui ont été comptabilisées à part. Ainsi, la liste ci-dessous regroupe les publications considérées lors de cette analyse. La Tableau 5 présente les chiffres obtenus à partir de cette analyse, alors que la Figure 35 illustre la proportion de ces citations.

- **3PGCIC 2013** [235] : Steffemel, L. A.; Flauzac, O.; Charao, A. S.; Barcelos, P. P.; Stein, B.; Nesmachnow, S.; Kirsch Pinheiro, M. & Diaz, D., "PER-MARE: Adaptive Deployment of MapReduce over Pervasive Grids", *8th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC'13)*, **2013**, 17-24.
- **UBICOMM 2014** [45] : Cassales, G.W.; Charão, A.S.; Kirsch-Pinheiro, M.; Souveyet, C. & Steffemel, L.A. « Bringing Context to Apache Hadoop », In: Jaime Lloret Mauri, Christoph Steup, Sönke Knoch (Eds.), *8th International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2014)*, August 24 - 28, **2014**, Rome, Italy, ISBN: 978-1-61208-353-7, IARIA, 252-258.
- **JCS 2014** [234] : Steffemel, L. A., Flauzac, O., Charao, A. S., P. Barcelos, P., Stein, B., Cassales, G., Nesmachnow, S., Rey, J., Cogorno, M., Kirsch-Pinheiro, M. & Souveyet, C., "Mapreduce challenges on pervasive grids", *Journal of Computer Science*, 10(11), July **2014**, 2194-2210.
- **CLIoT 2015** [233] : Steffemel, L. & Kirsch-Pinheiro, M. "CloudFIT, a PaaS platform for IoT applications over Pervasive Networks", In: Celesti A., Leitner P. (eds). *3rd Workshop on CLOUD for IoT (CLIoT 2015)*. Advances in Service-Oriented and Cloud Computing (ESOC 2015). Communications in Computer and Information Science, vol 567, **2015**, 20-32.
- **CN4IoT 2015** [236] : Steffemel, L.A. & Kirsch Pinheiro, M. "When the cloud goes pervasive: approaches for IoT PaaS on a mobiquitous world". In: Mandler B. et al. (eds), *EAI International Conference on Cloud, Networking for IoT systems (CN4IoT 2015)*, *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (LNICST)*, **2015**, 347–356.
- **JAIHC 2016** [46] : Cassales, G.W.; Charão, A.S.; Kirsch-Pinheiro, M.; Souveyet, C. & Steffemel, L.-A. "Improving the performance of Apache Hadoop on pervasive environments through context-aware scheduling", *Journal of Ambient Intelligence and Humanized Computing*, 7(3), **2016**, 333-345.
- **Big2DM 2015** [237] : Steffemel, L.A. & Kirsch-Pinheiro, M., "Leveraging Data Intensive Applications on a Pervasive Computing Platform: the case of MapReduce", *1st Workshop on Big Data and Data Mining Challenges on IoT and Pervasive (Big2DM)*, London, UK, June 2 - 5, 2015. *Procedia Computer Science*, vol. 52, Jun 2015, Elsevier, 1034–1039. doi: 10.1016/j.procs.2015.05.102.

- **ANT 2015** [44] : Cassales, G.W., Charao, A., Kirsch-Pinheiro, M., Souveyet, C. & Steffemel, L.A., "Context-Aware Scheduling for Apache Hadoop over Pervasive Environments", *The 6th International Conference on Ambient Systems, Networks and Technologies (ANT 2015)*, London, UK, June 2 - 5, 2015. *Procedia Computer Science*, vol. 52, Jun 2015, Elsevier, 202–209. doi: 10.1016/j.procs.2015.05.058.

En plus de ces articles, directement issus des thématiques abordées par le projet PER-MARE, celui-ci a également permis que d'autres collaborations sur le thème de l'usage de ressources hétérogènes puissent se mettre en place entre les membres du projet. Ces collaborations qu'on peut appeler satellites ont également donné lieu à des publications, listées ci-dessous :

- **ANT 2014** [83] : Engel, T.A., Charao, A., Kirsch-Pinheiro, M., Steffemel, L.A. "Performance Improvement of Data Mining in Weka through GPU Acceleration", *5th International Conference on Ambient Systems, Networks and Technologies (ANT 2014)*, Hasselt, Belgium, June 2 - 5, 2014. *Procedia Computer Science*, vol. 32, 2014, Elsevier, pp. 93–100.
- **JAIHC 2015** [82] : Engel, T.A., Charao, A., Kirsch-Pinheiro, M., Steffemel, L.A. "Performance Improvement of Data Mining in Weka through Multi-core and GPU Acceleration: opportunities and pitfalls", *Journal of Ambient Intelligence and Humanized Computing*, Springer, June 2015. doi:10.1007/s12652-015-0292-9.

Tableau 5. Nombre de citations par article, organisées par année de citation.

Référence	Année	Total	≤ 2017	> 2017	self
3PGCIC 2013	2013	11	4	0	7
JCS 2014	2014	7	2	1	4
Ubicomm 2014	2014	3	0	1	2
CN4IoT 2015	2015	7	0	2	7
CLIoT 2015	2015	6	3	1	2
Big2DM 2015	2015	4	1	1	2
ANT 2015	2015	9	4	3	2
JAIHC 2016	2016	9	1	6	2
ANT 2014	2014	9	2	6	1
JAIHC 2015	2015	6	1	4	1
Total / %		73	24,66 %	34,25 %	41,10 %

Le projet PER-MARE se construisant au fil des années, le nombre d'autocitations sur certaines publications, notamment celles établissant les bases de travail ([234] et [235]) et la plateforme CloudFIT ([236]), est naturellement élevé. Il est cependant à observer que le nombre de citations pour certains travaux (et notamment [46], [45] et [82]) sont en progression depuis 2017. Ceci s'explique surtout par une certaine démocratisation des applications *Big Data* et des applications de *data analyse* (sujet traité par les deux dernières publications).

Pour finir, il est à observer une volonté délibérée dans le projet PER-MARE de privilégier les publications de type « *open access* » gratuites avec comité de lecture. Parmi la dizaine de publications présentées ci-dessus, la moitié est en accès libre. Celles-ci contribuent à hauteur de 48,83% des citations (hors autocitations).

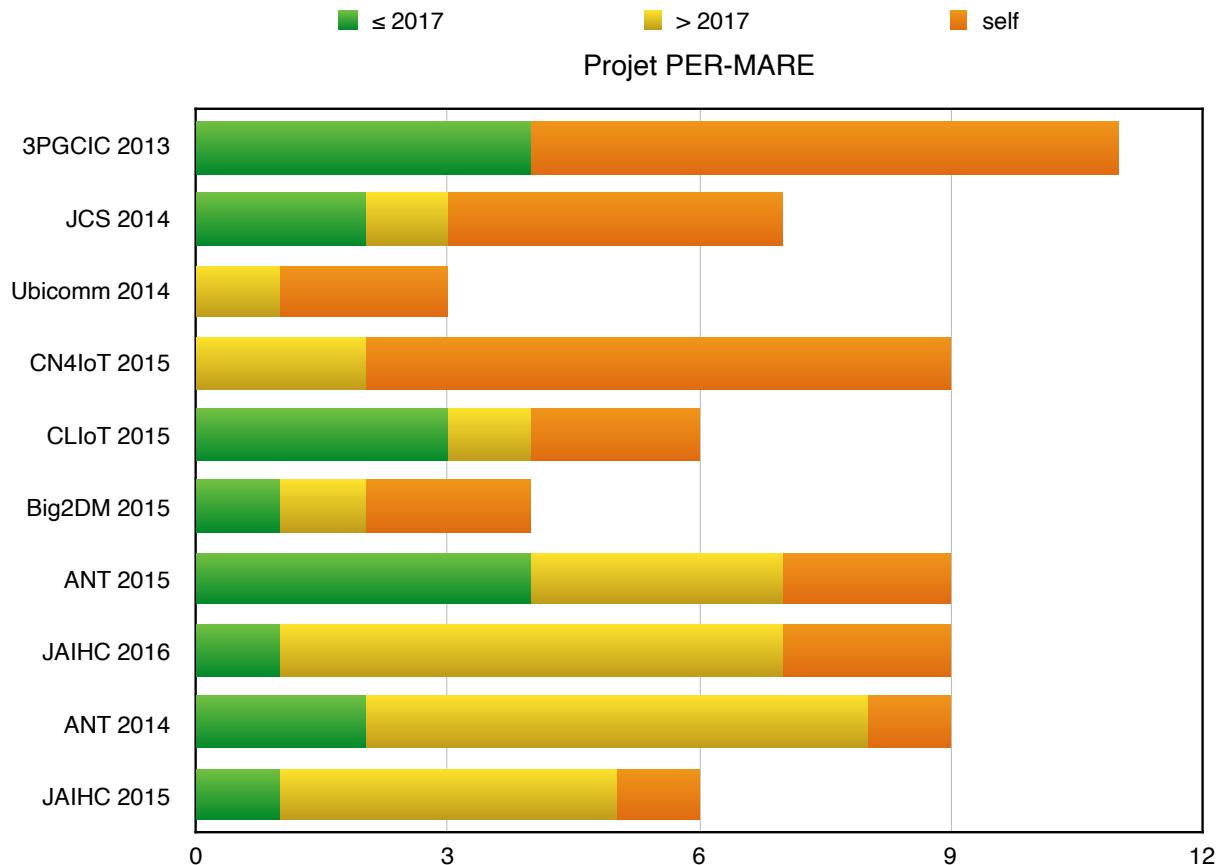


Figure 35. Graphique illustrant la distribution des citations dans le temps.

2.3 Bilan

Dans ce chapitre, nous avons discuté des travaux se déroulant de 2008 à 2010 (pour les travaux sur le *context grouping*) et de 2013 à 2015 (pour ceux sur le projet PER-MARE) dans le cadre de deux projets distincts : IST-MUSIC pour le premier (cf. section 2.1), et PER-MARE pour le second (cf. section 2.2).

Le fil conducteur de ces travaux a été la prise en compte des environnements pervasifs. Contrairement aux travaux présentés dans le chapitre 1, dans lesquels l'élément humain était au centre de l'analyse, ici l'attention s'est portée entièrement sur ces environnements. On s'est intéressé particulièrement à leur caractère hétérogène et dynamique. Celui-ci soulève des nombreuses questions, dont l'usage opportuniste des ressources pour des tâches de calcul, pour lesquels la prise en compte du contexte d'exécution peut s'avérer un élément déterminant, d'où la relevance des travaux comme le nôtre sur la distribution de l'information de contexte au sein de ces environnements.

La prise en compte de l'environnement et des ressources qu'il offre est aujourd'hui une tendance à la hausse avec le développement des plateformes de *fog/edge computing* et autres concepts associés [31, 56, 80, 110, 187, 258]. Tous ces concepts se partagent une même vision : l'objectif est de se servir des ressources avec une certaine capacité de calcul qui seraient disponibles (ou potentiellement sous-exploitées) dans l'environnement autour de l'utilisateur ou de la donnée. Ce modèle de calcul à proximité est particulièrement intéressant lorsqu'il est associé à l'IoT. Les données et/ou les

traitements issus/liés à l'IoT peuvent ainsi être pris en charge par des dispositifs proches des objets à l'origine de ces données ou traitements. Ceci permettrait, par exemple, de réduire de manière sensible le volume de données transmises sur les réseaux, ce qui représente un avantage non négligeable dans un contexte de *Big Data*, ou encore de réduire le temps de réaction suite à une prise de décision.

La disponibilité aujourd'hui d'une masse sans précédente de données, issues entre autres de l'IoT, associée à l'offre de calcul disponible (qui ce soit sur le *cloud* ou à travers les plateformes de *Fog Computing*) ouvre de nouvelles perspectives d'applications, surtout liées à l'utilisation des techniques de *data analyse*. On parle ici de ce que certains appellent « *Edge AI* » ou de l'usage de plateformes de *Fog Computing*, en association aux plateformes de *cloud*, pour l'application de techniques d'Intelligence Artificielle, et notamment le *Deep Learning*, sur des données issues de l'IoT pour la plupart. Il s'agit d'un tout nouveau champ d'application pour les ressources déjà disponibles dans les Systèmes d'Information, qui pourraient en bénéficier sur des nombreux secteurs : Industrie 4.0, logistique, mais aussi SIRH, finance, etc.

Quel que soit le domaine d'application envisagé, pour que ces scénarios deviennent une réalité, il faut une prise en compte adéquate du contexte d'exécution pour que l'hétérogénéité de ces environnements soit prise en considération. L'adaptation au contexte devient ainsi essentielle, comme le soulignent les résultats que nous avons obtenus avec le projet PER-MARE.

3 Contexte dans l'orientation service

Parmi les questions soulevées par l'Informatique Pervasive se pose bien évidemment la gestion de contexte, mais également la question de l'interopérabilité face à l'hétérogénéité des environnements pervasifs. Dans ce cadre, l'orientation service est apparue comme une approche viable pour le traitement de ces questions, apportant des solutions intéressantes aux problèmes posés, mais aussi en bénéficiant notamment de la sensibilité au contexte. L'orientation service (*Service Oriented Computing* - SOC) peut être vue comme un paradigme informatique fondé sur la notion de service comme unité de base pour une conception et un développement plus rapides, fiables et à moindre coût d'applications réparties sur des environnements hétérogènes [190, 189, 191]. Un service peut être défini comme une entité indépendante, dotée d'interfaces bien définies et pouvant être invoquée de manière standard, sans requérir de son client une quelconque connaissance sur la manière dont le service réalise réellement ses tâches [114]. Il s'agit d'éléments logiciels auto-décrits, indépendants de la plateforme et accessibles par une interface standard [7]. Clients et fournisseurs de service sont ainsi indépendants, communiquant uniquement grâce à l'interface de ces services. Les services peuvent ainsi être exposés, publiés, découverts, composés et négociés à la demande d'un client et invoqués par d'autres applications [191].

L'orientation service peut ainsi être caractérisée par son faible couplage, l'indépendance par rapport aux aspects technologiques et l'extensibilité [114]. C'est ce faible couplage qui rend la notion de service particulièrement attractive pour les environnements pervasifs, puisque ces environnements se caractérisent par la volatilité de leurs éléments [256]. En effet, l'orientation service, à travers son faible couplage et le caractère auto-descriptif des services, permet, d'une part, de mieux gérer la volatilité des environnements pervasifs et de mieux isoler les applications par rapport aux différentes technologies impliquées. D'autre part, la prise en compte de la notion de contexte a permis l'apparition de services pouvant mieux s'adapter à leur contexte d'exécution et donc à ces environnements très dynamiques.

Depuis 2007, la notion de services dans le domaine de l'Informatique Pervasive occupe une partie importante de mes recherches. Différentes problématiques de recherche ont ainsi pu être traitées au fil des années : (i) l'identification d'éléments de contexte pertinents pour un service (ou une application) ; (ii) la sélection du service (ou de l'implémentation) le plus approprié en fonction du contexte d'exécution ; (iii) la prise en compte de l'utilisateur et de ses objectifs métiers dans la définition de ces services ; et (iv) l'anticipation des besoins utilisateur pour un comportement plus proactif. Ces différentes problématiques ont donné lieu à plusieurs contributions présentées dans ce qui suit. L'élément commun à ces propositions, au-delà de la notion de contexte et du domaine dans lequel elles s'inscrivent (l'Informatique Pervasive), est la notion de service : dans chacune de ces contributions, on retrouve la vision d'un système comme un ensemble de services proposés à l'utilisateur, et dont l'adaptation au contexte s'avère nécessaire. Les sections qui suivent résument les contributions proposées et leur impact.

3.1 Fouille de contexte (« *Context Mining* »)

3.1.1 Rappel de la problématique

Un challenge particulièrement important pour le développement d'applications sensibles au contexte est l'identification des éléments de contexte qui seront observés par l'application [16,74 ,96,97]. En effet, l'information de contexte étant un élément clé pour le fonctionnement de ces applications, l'identification des éléments pertinents pour celles-ci devient une tâche capitale dans leur développement, obligeant les concepteurs de ces applications à anticiper leurs combinaisons et leurs caractéristiques pertinentes avant l'implémentation [16]. Le même s'applique aux services sensibles au contexte, dont le comportement doit s'adapter au contexte d'exécution.

Même si cette épineuse question demeure, à ce jour, sans réponse satisfaisante, elle permet de soulever une deuxième question annexe : serait-il possible de caractériser l'usage d'un service (ou d'une application) par l'observation de son contexte d'exécution ? En d'autres termes, en observant le contexte d'exécution d'un service, serait-il possible de déterminer des éléments de contexte récurrents capables de caractériser l'usage de ce service (ou de cette application) par un utilisateur lambda ?

Ce travail de « fouille de contexte » découle particulièrement de cette question. L'objectif ici est donc d'utiliser les techniques de fouille de données sur des données de contexte, afin d'essayer d'identifier des éléments de contexte pertinents dans l'usage d'un service (ou d'une simple application²⁰).

3.1.2 Contribution & Impact

Dans le présent travail, réalisé essentiellement entre 2012 et 2014, d'abord dans le cadre du travail de master de Ali Jaffal [119], puis pendant le début de sa thèse [118], il a été proposé une méthodologie permettant d'associer l'usage d'un service à des situations caractéristiques, reconnues grâce aux éléments de contexte observés. On identifie ainsi, à partir des données issues de l'usage, des éléments de contexte permettant de caractériser l'utilisation d'un service. L'objectif ici est de pouvoir reconnaître l'influence d'un (ou plusieurs) élément(s) de contexte dans le choix d'un service par un utilisateur. Pour cela, cette méthodologie propose l'application de l'Analyse de Concepts Formels (ACF) [203,266], laquelle permet d'organiser les services en plusieurs *clusters* en fonction des éléments de contexte observés pendant leur usage.

L'Analyse de Concepts Formels (en anglais, *Formal Concept Analysis*) fournit une approche de structuration de connaissances à partir de données complexes. Elle permet de construire sur la base d'une relation binaire une hiérarchie de concepts traduisant une association forte entre un sous-ensemble d'objets et un sous-ensemble d'attributs. Son objectif principal serait de déterminer des paires d'ensembles d'objets et d'ensembles d'attributs qui se définissent mutuellement et de manière unique, appelées concepts formels [119]. L'ACF est centrée sur cette notion de concept qui, de manière informelle, peut être vue comme un groupement d'objets et de leurs attributs communs. Un treillis de concepts traduit ensuite la relation d'ordre hiérarchique entre les concepts formels, et peut être utilisé à des fins de *clustering*, de classification, de prédiction ou d'approximation [118].

A travers l'ACF, on peut regrouper un ensemble d'objets en fonction des propriétés partagées entre eux. Les objets ici représentent les services, alors que les propriétés correspondent aux éléments de contexte communs observés pendant l'usage de ces services. Il en résulte plusieurs « classes », permettant de regrouper les services utilisés et les éléments de contexte observés pendant leur utilisation. La Figure 36, issue de [117], illustre des résultats obtenus à partir de l'analyse de données réelles correspondant à l'observation de l'usage d'une tablette Android pendant 69 jours. Plusieurs éléments de contexte ont pu être observés comme la localisation de l'utilisateur, les SSID des réseaux utilisés, ainsi que les applications utilisées à un instant t . La Figure 36 présente ainsi un treillis obtenu avec l'ACF montrant l'usage des applications en fonction du SSID réseau²¹. Cette analyse permet de classer ensemble les applications et les réseaux dans lesquels ces applications ont été utilisées, caractérisant ainsi des *patterns* d'usage de l'utilisateur. Par exemple, l'application « *Alarm* » n'a été utilisée qu'à partir du réseau « *Network_4* ». L'analyse de ces classes offre une vue sur le comportement d'un utilisateur face aux services (représentés ici par les applications) proposés par un

²⁰ Pour les travaux de fouille de contexte, nous traitons de manière indistincte les notions de service et d'application, puisque le focus ici sont simplement des unités dont l'exécution est sollicitée par l'utilisateur.

²¹ Pour des raisons de respect de la vie privée, les noms des réseaux ont été anonymisés.

système et permet ainsi de mieux comprendre l'impact que certains éléments de contexte pourraient avoir dans l'usage de certains services.

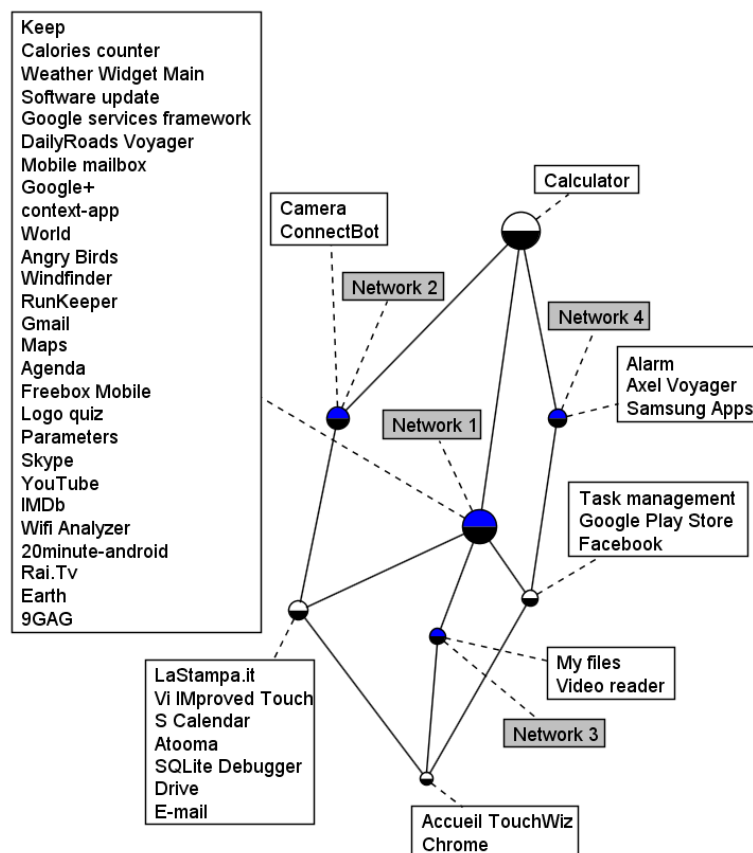


Figure 36. Application de l'ACF pour l'analyse du contexte d'usage d'applications mobiles [117].

Le choix de l'ACF est un élément particulièrement innovant dans ce travail. La plupart des travaux utilisant la fouille de contexte [206,229] le font à travers des méthodes statistiques, comme les Réseaux Bayésiens [206] ou les chaînes de Markov [161, 271]. L'usage de l'ACF était alors inédit au moment de l'apparition de ce travail (2012-2014). Par ailleurs, la plupart de ces travaux [206,229,161] font usage des techniques de fouille de données pour la reconnaissance des activités de l'utilisateur (*e.g.* courir, marcher, dormir, etc.), et non pour l'analyse de la pertinence d'un élément de contexte comme ici. Dans ce cadre, on peut citer [205, 206] comme une approche semblable, puisque ces auteurs cherchent à apprendre, à partir des données, des associations entre les éléments de contexte et des actions à prendre par le système (les actions prenant la place ici des services). L'adoption de l'ACF apporte une caractéristique particulièrement intéressante à ce travail : une classification « multi-classes ». En effet, contrairement à beaucoup des méthodes statistiques de *clustering*, l'ACF permet le chevauchement entre les différentes « classes » identifiées à partir des données. Autrement dit, un même service peut être placé simultanément dans différents concepts formels, identifiant ainsi plusieurs situations, reconnues à travers un ensemble d'éléments de contexte, où ce service a été invoqué. Par exemple, dans le treillis illustré par la Figure 37, issu de [116], l'application « Gmail » est associée à plusieurs de ces « classes » (*i.e.* à plusieurs concepts formels), représentant le fait que cette application est utilisée en plusieurs situations distinctes, caractérisées par différents éléments de contexte.

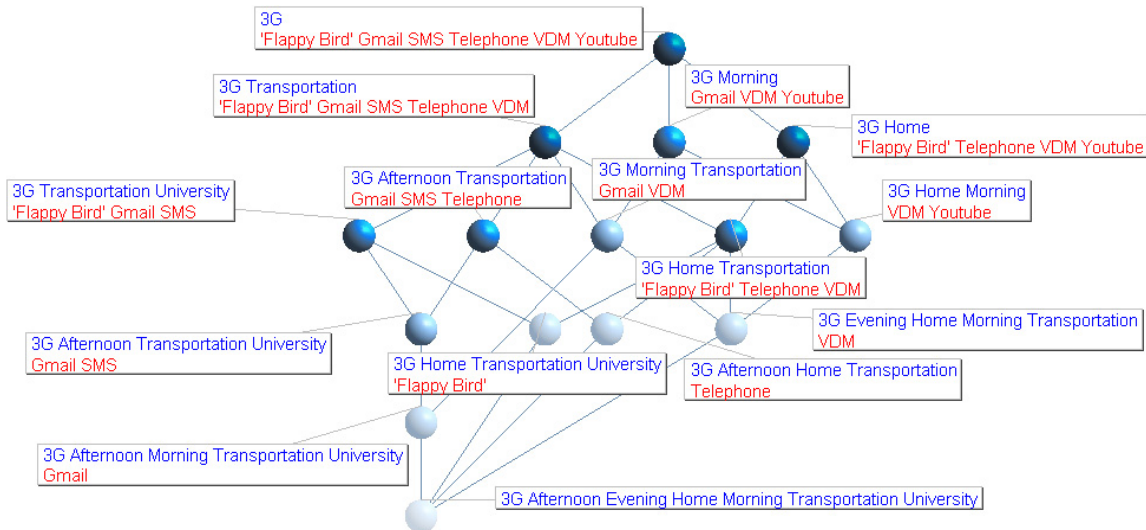


Figure 37. Treillis associant l'usage de certaines applications à leur contexte d'usage [116].

Cependant, l'impact le plus intéressant de ce travail ne se trouve peut-être pas dans ses résultats proprement dits, mais surtout dans les difficultés surmontées pour les obtenir. Pour valider ce travail, différentes expérimentations ont été conduites : d'abord une expérimentation avec des données réelles [117, 116], obtenues à l'aide d'un logiciel permettant le *monitoring* d'un dispositif mobile (des applications invoquées et de leur contexte d'exécution), puis une enquête auprès d'un groupe d'étudiants (dont les résultats sont mentionnées dans [120]). Ces expériences démontrent qu'il est possible d'établir un lien entre une application (ou un service) et des éléments de contexte, notamment lorsque l'utilisateur a des habitudes d'usage réguliers (c.a.d. qu'il a l'habitude d'utiliser certaines applications à des situations précises, comme par exemple, l'application réveil à la maison, les jeux pendant le temps de transport, etc.). Le processus d'analyse de ces données à travers la méthodologie proposée [119, 117] a également mis en lumière plusieurs limitations des méthodes de fouille de données par rapport aux données de contexte et à ses caractéristiques.

Tout d'abord, l'hétérogénéité des données de contexte apparaît comme un obstacle majeur pour une analyse généralisée et totalement automatisée de ces données. Bien souvent les méthodes de fouille de données s'appliquent à certaines « catégories » des données. Par exemple, la FFT (*Fast Fourier Transformation*) utilisée dans [206] est particulièrement adaptée aux données numériques, alors que l'ACF appliquée dans ce travail s'adapte davantage aux données symboliques (*i.e.* des étiquettes). Les données de localisation démontrent particulièrement bien cette question. Un prétraitement a été nécessaire afin de traduire les données de localisation, extraites lors de la première expérimentation, vers des « étiquettes » (*localisation_1, localisation_2, etc.*) pouvant être plus facilement exploitées par l'ACF. Les données de coordonnées GPS ont ainsi dû être regroupées en plusieurs régions géographiques, auxquelles ces étiquettes ont été affectées. Étant donné l'hétérogénéité qui caractérise les données de contexte, difficile d'imaginer alors l'application d'une seule et unique technique pour l'analyse des données quel que soit l'élément de contexte considéré, et en même temps, difficile de ne pas considérer le risque de perte d'information relatives aux possibles liens existants entre les différents éléments de contexte à force de fractionner l'ensemble de données en fonction de leur nature.

Il paraît alors claire que les techniques de fouilles de données actuelles ne semblent pas encore assez adaptées pour gérer l'hétérogénéité des données de contexte sans une phase de prétraitement ou de *training* conséquente. Or ces phases de prétraitement et de *training* peuvent aussi poser problème dans le cadre de l'Informatique Pervasive. La plupart du temps, ces phases sont « *off-lines* », c'est-à-

dire qu'elles doivent se réaliser en dehors de l'exécution normale d'une application et souvent sous l'œil attentif d'un expert. Il en résulte très souvent des modèles qui sont eux utilisés lors de l'exécution de l'application.

La question qui se pose alors est : peut-on envisager toujours ces phases de prétraitement et de *training* ? Il paraît évident que ces phases ne peuvent pas être réalisées avec des données de chaque utilisateur concerné. On peut donc imaginer l'application des techniques proposées sur des jeux de données prédéfinies, obtenues en laboratoire ou dans des phases de tests avec des utilisateurs choisis au préalable. Les modèles gérés suite à ces phases se présentent comme des modèles génériques appliqués à n'importe quel utilisateur. Or, la seconde expérience menée a démontré le caractère très personnel de cette relation entre une application et son contexte d'exécution habituel. En gros, le comportement d'un utilisateur pouvant être très particulier par rapport à d'autres, la pertinence d'un élément de contexte par rapport à une application peut être totalement différente entre un utilisateur et un autre. Même si l'échantillon utilisé pour la 2^{ème} expérience ne comportait que des individus ayant un même profil (que des jeunes entre 20 et 25 ans, étudiants dans une même formation, particulièrement sensibilisés aux nouvelles technologies), ils ont présenté des comportements très distincts entre eux, et très souvent, distincts aussi de l'utilisateur ayant réalisé la 1^{ère} expérience. La pertinence de localisation dans les services invoqués illustre bien cette différence : alors que la localisation permet de caractériser un bon nombre d'applications dans le cadre de la 1^{ère} expérience, celle-ci paraît nettement moins pertinente pour les utilisateurs dans la seconde expérience.

Dans ces conditions, on peut alors se demander quelle pourrait être la pertinence d'un tel modèle générique. L'exemple des certains détecteurs de mouvement, comme les objets connectés commercialisés par Withings²², illustre bien ce cas : dans le modèle Pulse de Withings, le comptage de pas et de la distance parcourue peut s'avérer très différent (et pas forcément adapté à la réalité) en fonction de la taille de l'utilisateur et de sa foulée. On a observé, dans ce modèle, qu'une personne de plus petite stature avait systématiquement moins de pas au compteur qu'une personne de plus grande stature, alors que dans la réalité, c'était souvent l'inverse qui se passait. L'application d'algorithmes d'apprentissage comme le suggère [148, 271] semble alors inévitable, surtout ceux d'apprentissage non-supervisé. L'essor des techniques de *Machine Learning* [156], et notamment de *Deep Learning*, ouvre des nouvelles possibilités, même si ces techniques demeurent sensibles à la question de l'hétérogénéité des données ou de leur prétraitement/*training*, entre autres.

A partir de ces constats réalisés au cours de ces travaux, nous avons voulu analyser à quel point les difficultés que nous avons rencontrées avec l'application de l'ACF pouvaient également survenir lors de l'usage d'autres méthodes de *Machine Learning*. Ainsi, en 2019-2020, un travail de collaboration [23] a été mis en place visant à étudier l'usage de techniques de *Machine Learning* (ML) avec des données de contexte. A travers une étude de la littérature sur différents domaines (notamment l'Informatique Pervasive, l'IoT et le *Machine Learning*), nous avons analysé comment les approches de ML sont utilisées pour la fouille de contexte et les conditions nécessaires pour que cette analyse puisse tenir compte des particularités propres aux informations de contexte. Combinant des mots clés tels que « *context prediction* », « *context awareness* », « *machine learning* », « *data mining* » ou encore « IoT », nous avons identifié 200 articles, dont 30 (tous postérieurs à 2013 et la moitié après 2016) ont été analysés en détail. Cette étude [23] nous a permis de souligner que plusieurs caractéristiques propres aux informations de contexte pouvaient impacter l'usage de techniques de ML. Tout d'abord, plusieurs auteurs, tels que [91, 207], reconnaissent l'importance de la qualité des données pour la réussite de ces techniques. Or les données de contexte sont, par définition, hétérogènes et incertaines [256, 52, 103]. Souvent issues de capteurs, ces données peuvent être erronées, dispersées et incomplètes, ce qui peut avoir un impact négatif sur le temps d'exécution, la complexité ou même la qualité des modèles générés par le ML. De même, certaines techniques de ML vont demander une distribution plutôt équilibrée des données dans les classes disponibles, ce qui peut aussi poser

²² <http://www.withings.com>

problème lors qu'on considère des données de contexte, dont une telle distribution n'est pas toujours possible, notamment en fonction du caractère dynamique et incertain de ces informations.

Tous ces problèmes liés à la qualité des données sont généralement traités par les différentes approches de *Machine Learning* dans leur phase de prétraitement (p.ex. [9,153]). En effet, l'étude que nous avons menée [23] a présenté des indices indiquant que les approches de ML reposent très fortement dans les phases de prétraitement. Nous avons également observé que ces approches considèrent le plus souvent les informations de contexte comme une donnée ordinaire, sans vraiment tenir compte des spécificités propres à la notion de contexte, reconnues par la littérature [52, 24, 103], comme l'extrême hétérogénéité, la dynamique, l'incomplétude et son caractère incertain.

Ces limitations, que nous avons également observés lors de l'application de l'ACF, interrogent sur une possible généralisation de l'usage des techniques de ML pour la fouille de contexte à grande échelle, c.a.d. à l'échelle de tout un Système d'Information (SI). En effet, dans [23], nous avons soutenu une vision d'une « *context mining facility* », autrement dit, de la fouille de contexte comme un service proposé par un SI à toutes ses applications. Les informations de contexte sont aujourd'hui de plus en plus faciles à capturer. Elles ouvrent des multiples opportunités d'usage au sein des SI (adaptation au contexte, recommandation de services ou de contenu, prédiction / anticipation de besoins ou des actions des utilisateurs, prise de décision, etc.). On peut donc envisager la généralisation de ces comportements qu'on pourrait qualifier de « *smart* » ou « *intelligent* », en proposant le support à la fouille de contexte comme un service propre à un SI.

Or la clé de voûte de cette vision repose sur la possibilité d'offrir des mécanismes de raisonnement à l'échelle de tout un système pouvant ainsi évoluer avec celui-ci et exécuter de manière continue (pour être disponible à toutes les applications à tout moment), avec un minimum d'intervention humaine [23]. Ce changement d'échelle a un fort impact sur les mécanismes de ML et sur la gestion de contexte de manière générale : au lieu de penser ce support pour des applications précises, avec des objectifs et des données précis, il faut le penser pour tout un système, avec son écosystème d'utilisateurs (décideurs, employés, clients...) et d'applications supportant les différents processus et corps de métiers qui cohabitent dans une organisation. On ne peut pas forcément choisir un sous-ensemble précis de données à observer ou des méthodes à appliquer sans risquer de pénaliser les futures applications et services. Un même service de fouille de contexte devrait être capable de supporter les besoins de multiples applications (existantes ou futures). Des nouvelles applications mais également des nouvelles informations de contexte pourraient être intégrées au système à tout moment, en fonction de leur disponibilité. Dans ces conditions, les phases de prétraitement et un possible besoin d'intervention humaine pour le bon usage d'une technique de ML peuvent poser problème. D'une part, la réalisation de ces phases de manière générique, visant différents besoins et différentes méthodes ne serait guère aisée, et, en même temps, la non-réalisation de ces phases pourraient entraîner une perte d'information puisqu'une partie de données récoltées pourraient rester ignorées. D'autre part, l'arrêt du service pour la réexécution de certaines phases, comme le prétraitement des données ou l'entraînement, peut également avoir un impact négatif sur le système au fur et à mesure que d'autres applications ou services, potentiellement critiques pour l'organisation, peuvent être impactés.

Heureusement, les techniques de ML se démocratisent et continuent à évoluer. Même si aujourd'hui leur réussite repose encore beaucoup sur des décisions de design (e.g. choix de l'algorithme, nettoyage/préparation appropriés des données, choix d'hyper-paramètres, etc.), faisant de l'expertise (humaine) du « *data scientist* » difficile à remplacer, de plus en plus de *framework* et d'outils, comme AutoML [112], sont disponibles ou en développement. Ces derniers visent bien souvent un usage simplifié des techniques de ML, voir une certaine automatisation dans l'exécution de leur *pipeline*. On peut donc espérer que les difficultés identifiées dans [23] pourront s'amenuiser avec le temps, pour que cette vision d'un « *context mining facility* » puisse voir le jour.

Concernant notre travail avec l'ACF, il convient d'observer que cette même phase de prétraitement rend presque inviable un usage totalement automatique de la méthode proposée. D'ailleurs, se basant sur des données existantes, l'approche que nous avons proposée dans [119, 117] ne se positionne pas dans le cadre d'un nouveau développement (le développement d'une nouvelle application ou d'un nouveau service), mais surtout dans le cadre d'un processus d'amélioration continue d'un catalogue de services existant. Elle pourrait ainsi être particulièrement intéressante pour les Systèmes d'Information, dans lesquels l'analyse de l'usage actuel du système et de ses services pourrait conduire à des évolutions et à des améliorations dans l'offre de services.

Enfin, ces difficultés rencontrées dans l'analyse de données contextuelles ont conduit à un changement de focus dans la thèse d'Ali Jaffal [118], où certains de ces travaux ont été conduits. Ces difficultés ont démontré des lacunes dans l'application de l'ACF à des données hétérogènes comme les informations contextuelles, mais également les difficultés pour l'application de cette technique surtout pour les non experts. La thèse s'est alors concentrée sur cette méthode de l'analyse et la possibilité de lui apporter des changements permettant une meilleure prise en main et une analyse plus facile pour des utilisateurs non experts. L'Informatique Pervasive est devenue ainsi un cas d'étude pour la thèse, et pas son sujet principal, au vu notamment des limitations pour une application automatique de cette méthode aux systèmes sensibles au contexte.

3.1.3 Bibliométrie

Ces travaux concernant la fouille de contexte ont démarré en 2012, avec le travail de master d'Ali Jaffal et se sont poursuivis jusqu'à 2014, désormais comme partie intégrante de la thèse de celui-ci, sous l'encadrement de Bénédicte Le Grand. Trois articles directement issus de ces travaux peuvent être cités ici [117, 116, 120], auxquels s'ajoute notre dernier article sur l'applicabilité des techniques de *Machine Learning* aux informations de contexte [23]. Par ailleurs, ces travaux ont mis en avant un usage potentiel de l'ACF avec les données de contexte, ce qui a ouvert la porte à plusieurs collaborations sur d'autres domaines, et notamment le BPM (*Business Process Modeling*). Plusieurs publications mentionnées ci-dessous (Adaptive 2013, BPMDS 2013, Chapitre BPM 2014, CoMoRea 2020) sont issues de ces collaborations qui n'auraient pas eu lieu sans ce travail sur la fouille de contexte. L'ensemble de ces publications sont listées ci-dessous. Comme pour les chapitres précédents, chaque publication a été analysée, à l'aide du site scholar.google.com, par rapport au nombre de citations. Ces citations ont été organisées en deux catégories, en fonction de leur date de publication : antérieur à 2016 (jusqu'à 3 ans après la première publication) ou postérieur à 2016. Les autocitations ont également été comptabilisées. Le Tableau 6 détaille les données obtenues, illustrées par la Figure 38.

- **ANT 2015** [116] : Jaffal, A. ; Grand, B. L. ; Kirsch-Pinheiro, M., « Refinement Strategies for Correlating Context and User Behavior in Pervasive Information Systems », *1st Workshop on Big Data and Data Mining Challenges on IoT and Pervasive (Big2DM), 6th International Conference on Ambient Systems, Networks and Technologies (ANT-2015)*, Procedia Computer Science, vol. 52, Jun 2015, pp.1040-1046. DOI : <http://dx.doi.org/10.1016/j.procs.2015.05.103>
- **Ubicomm 2014** [117] : Jaffal, A. ; Kirsch-Pinheiro, M. & Le Grand, B., « Unified and Conceptual Context Analysis in Ubiquitous Environments », In : Jaime Lloret Mauri, Christoph Steup, Sönke Knoch (Eds.), *8th International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2014)*, August 24 - 28, 2014, ISBN 978-1-61208-353-7, IARIA, pp. 48-55.
- **EGC 2016** [120] : Jaffal, A.; Grand, B. L. & Kirsh-Pinheiro, M., « Extraction de connaissances dans les Systèmes d'Information Pervasifs par l'Analyse Formelle de Concepts », *Extraction et Gestion des Connaissances (EGC 2016)*, Revue des Nouvelles Technologies de l'Information, RNTI-E-30, 2016, pp. 291-296

- **AdaptiveCM 2013** [136] : Kirsch-Pinheiro, M. & Rychkova, I., « Dynamic Context Modeling for Agile Case Management », In : Y.T. Demey and H. Panetto (Eds.), *2nd International Workshop on Adaptive Case Management and other non-workflow approaches to BPM (AdaptiveCM 2013), OnTheMove Federated Workshop (OTM 2013 Workshops)*, LNCS 8186, Graz, Austria, 9-13 September **2013**, Springer, pp. 144–154, 2013.
- **BPMDS 2013** [211] : Rychkova, I. ; Kirsch Pinheiro M. & Le Grand B., « Context-Aware Agile Business Process Engine: Foundations and Architecture », In : Nurcan, S., Proper, H., Soffer, P., Krogstie, J., Schmidt, R., Halpin, T. & Bider, I. (Eds.), *Enterprise, Business-Process and Information Systems Modeling, Proceedings of the 14th Working Conference on Business Process Modeling, Development, and Support (BPMDS 2013)*, Lecture Notes in Business Information Processing, vol. 146, Valence : Espagne, **2013**, pp. 32-47.
- **Chapitre BPM 2014** [212] : Rychkova I. ; Kirsch-Pinheiro M. & Le Grand B., « Automated Guidance for Case Management: Science or Fiction? », In : Ficher, L. (Ed.), *Empowering Knowledge Workers: New Ways to Leverage Case Management*, Series BPM and Workflow Handbook Series, Future Strategies Inc., **2014**, pp. 67-78. ISBN : 978-0-984976478
- **CoMoRea 2020** [23] : Ben Rabah, N.; Kirsch Pinheiro, M.; Le Grand, B.; Jaffal, A. & Souveyet, C., “Machine Learning for a Context Mining Facility”, *16th Workshop on Context and Activity Modeling and Recognition, 2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, **2020**, pp.678-684

Tableau 6. Analyse bibliométrique des publications liées à la fouille de contexte.

	Année	Total	≤ 2016	> 2016	self
ANT 2015	2015	2	1		1
UbiComm2014	2014	7	1	1	5
EGC 2016	2016				
AdaptiveCM 2013	2013	5	2	2	1
BPMDS 2013	2013	10	4	5	1
Chapitre BPM 2014	2014	2	1	1	
CoMoRea 2020	2020				
Total / %		21	42,86 %	28,57 %	28,57 %

Comme on peut observer à travers le Tableau 6, ces articles comptent encore un nombre réduit de citations, la plupart concentrées sur les premières années de leur apparition (cf. Figure 38). Ce comportement n'est pas étonnant, puisque, d'une part, la fouille de contexte, surtout celle visant à identifier la pertinence d'une information de contexte, demeure un domaine assez confidentiel et en développement. Peu de travaux encore traitent cette question, la majorité des travaux se concentrant sur l'identification des activités d'un utilisateur. D'autre part, l'usage de la notion de contexte dans le BPM n'est pas non plus généralisé, même si cet usage se développe de plus en plus (e.g. [197,219]), ce qui explique le nombre plus important de citations pour les publications dans ce domaine. Au vu de ces informations, il paraît évident que l'impact principal de ces travaux ne peut pas se mesurer, pour l'instant, en termes de citations, mais bel et bien en fonction des opportunités qu'il a ouvert et les collaborations qu'il a permis de développer. Au-delà des publications précédemment mentionnées

(Adaptive 2013, BPMDS 2013, Chapitre BPM 2014, CoMoRea 2020), nées des échanges avec d'autres chercheurs, les échanges autour de la fouille de contexte ont également abouti à une invitation au jury de thèse d'Arun Ramakrishnan [206], à la KULeuven, en tant que membre extérieur du jury.

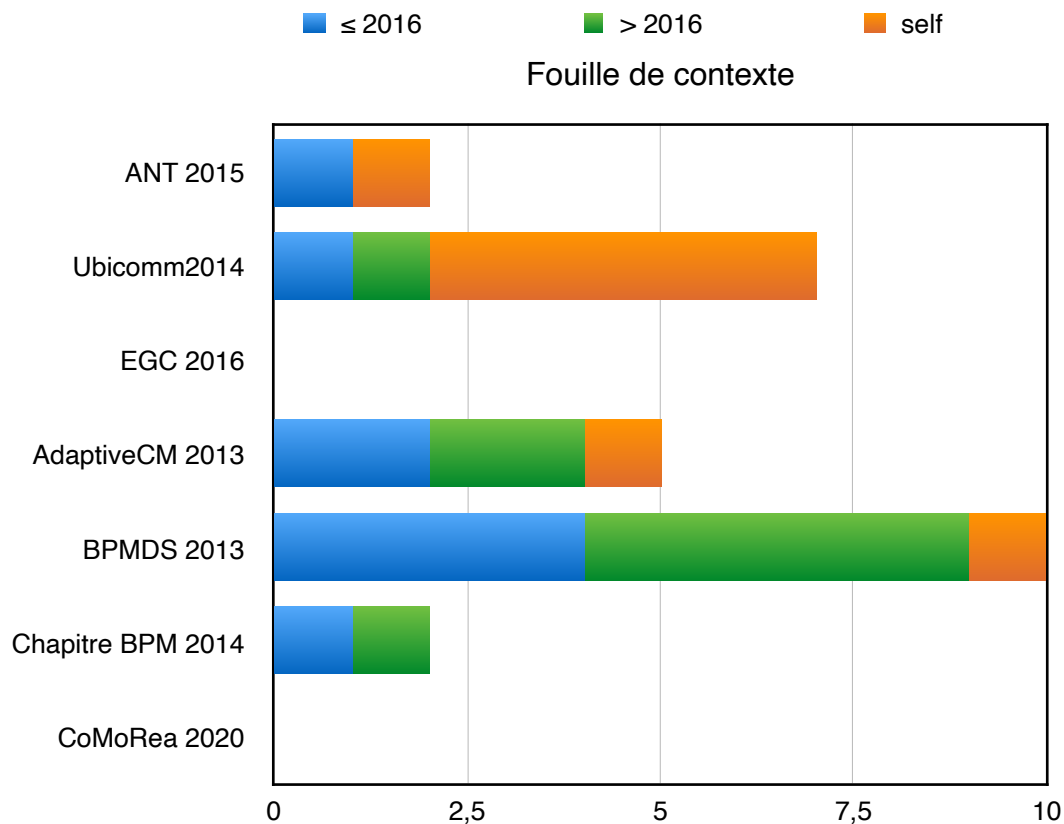


Figure 38. Évolution des citations liées à la fouille de contexte.

3.2 Sélection de services par comparaison de graphes

3.2.1 Rappel de la problématique

Dans un environnement pervasif, caractérisé par son dynamisme et son hétérogénéité, un même service peut aisément disposer de plusieurs implémentations distinctes. En effet, par définition, un service propose une interface permettant à un client de l'invoquer sans avoir connaissance de son implémentation. C'est ce principe d'une unité invocable à travers une interface « *standard* » qui rend l'orientation service un paradigme aussi intéressant pour gestion de l'interopérabilité. Derrière cette interface, il est possible d'imaginer différentes implémentations possibles (par exemple, proposées par différents fournisseurs ou offrant différentes qualités). Les environnements pervasifs étant particulièrement hétérogènes, l'existence de différentes implémentations pour une même interface devient intéressante, voir même souhaitable, afin de mieux supporter la variabilité de ces environnements.

Il se pose alors la question du choix de l'implémentation la mieux adaptée aux circonstances d'exécution. Ces circonstances peuvent être assimilées au contexte d'exécution de ce service (ou de son client). Or l'information de contexte étant de nature incertaine et souvent incomplète, le manque d'information peut avoir un impact non négligeable sur le mécanisme de sélection. Dans ce cas, une deuxième question se pose : comment choisir l'implémentation d'un service la mieux adaptée au contexte d'exécution, sachant que cette information risque d'être incomplète ? Les travaux décrits ici s'attaquent précisément à cette question.

3.2.2 Contribution & Impact

Réalisé dans le cadre du projet européen MUSIC, entre 2008 et 2009, ces travaux proposent un mécanisme de sélection de service sensible au contexte basé sur la comparaison de graphes, inspiré notamment des mesures de similarité proposées dans le cadre de travaux sur les collecticiels, présentés précédemment (chapitre 1).

A partir d'une première sélection, dans laquelle uniquement les aspects fonctionnels sont pris en considération, ce mécanisme cherche à identifier l'implémentation (répondant déjà à la demande fonctionnelle) la mieux adaptée au contexte d'utilisation courant. Pour cela, le mécanisme considère que chaque implémentation est conçue en ciblant des situations précises (e.g. un service visant particulièrement des clients dans une certaine zone géographique, ou demandant des ressources particulières pour un affichage chez le client). Ces situations sont décrites par un contexte « requis », associé à la description OWL-S du service. Celui-ci est comparé alors au contexte d'exécution du client. Ces deux descriptions (le contexte requis et le contexte client) sont interprétées en tant que graphes, dans lesquels les éléments de contexte sont vus comme des nœuds et leurs relations comme des arcs les reliant. Les deux graphes sont comparés à l'aide de différentes mesures de similarité : d'abord des mesures dites « locales », qui vont comparer les nœuds individuellement, puis des mesures « globales », qui regardent l'ensemble de deux graphes. La Figure 39 illustre cette comparaison entre deux graphes : les éléments de contexte (illustrés par les nœuds « *conceptA* », « *conceptB* », etc.) sont comparés deux par deux par les mesures « locales », alors que les graphes dans leur totalité sont analysés par les mesures globales. Enfin, les résultats obtenus par ces mesures sont utilisés pour ordonner les implémentations répondant aux aspects fonctionnels, permettant ainsi de sélectionner celle étant la plus favorable au contexte d'exécution côté client.

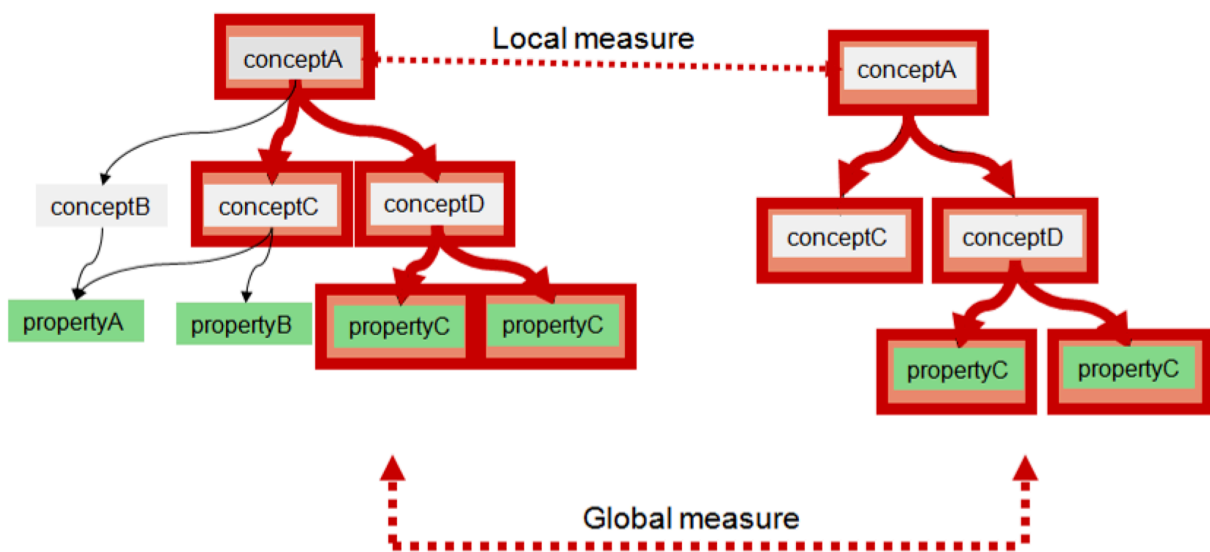


Figure 39. Comparaison entre deux graphes de contexte à travers les mesures de similarité locales et globale [256].

On remarque, dans ce mécanisme, que la notion de contexte est analysée comme une exigence non-fonctionnelle associée à chaque implémentation d'un service. L'objectif ici est alors de départager les différentes implémentations pouvant satisfaire la demande formulée par le client et de choisir celle qui semble correspondre au mieux au contexte d'utilisation.

Faisant partie du projet MUSIC, ces travaux utilisent le modèle de contexte proposé dans ce projet [262,210], lequel organise l'information de contexte en entité (l'entité que l'on observe, e.g. l'utilisateur ou le terminal), le « *scope* » (ce qu'on observe de l'entité, par exemple, la localisation, la

mémoire disponible, etc.), et la représentation (comment l'information est structurée), avec les métadonnées associées à l'observation. Ces métadonnées sont remontées par le *middleware* MUSIC [218] lors de l'observation du contexte d'exécution (c.a.d. lors de l'acquisition de l'information de contexte) et agissent comme des indicateurs de qualité pour cette information. Ces métadonnées peuvent ainsi être utilisées par les mesures de similarité afin de mieux juger de la fiabilité des informations de contexte récoltées, et par conséquent, de la fiabilité des résultats obtenus par la mesure elle-même [255, 256]. Ceci permettrait de mieux tenir compte du caractère incertain de l'information de contexte.

Par ailleurs, les mesures de similarité « locales » vont également tenir du « *scope* » affiché par chaque « nœud » du graphe (c.a.d. chaque élément de contexte). Ces mesures sont, en effet, associées à un *scope* précis (par exemple, la localisation) auquel elle s'applique, ce qui permet, d'une part, de tenir compte de la sémantique propre à ce « type » d'information de contexte, et d'autre part, de limiter les comparaisons à uniquement les nœuds de même nature (*i.e.* ayant le même *scope*). Bref, on ne compare que des informations qui sont comparables. Enfin, en cas de représentation distincte pour un même *scope* (*e.g.* la localisation représentée d'un côté sous forme de coordonnées GPS et de l'autre sous forme d'adresse postale), des fonctions de transformation sont disponibles dans le *middleware* MUSIC, permettant le passage d'une information d'une représentation à une autre, toujours dans le même *scope* [255,256].

L'approche par graphe proposée cible particulièrement l'incomplétude de l'information de contexte. On mesure, à travers notamment les mesures globales, le degré de couverture du graphe défini par le contexte d'utilisation sur celui défini par le contexte requis. C'est l'implémentation ayant la meilleure couverture qui sera proposée au client. Il n'est donc pas nécessaire de disposer de l'intégralité des informations de contexte attendues pour effectuer la sélection d'un service. Si certains nœuds, correspondant à certains éléments de contexte, ne trouvent pas de correspondance dans le contexte d'utilisation observé, ceci n'empêchera pas l'algorithme de calculer un degré de couverture et d'ainsi pouvoir classer l'implémentation en question par rapport aux autres.

L'incomplétude de l'information de contexte n'est pas particulièrement prise en considération par les mécanismes de sélection de service sensible au contexte tels que [22, 244, 248]. Ceux-ci s'intéressent peu (voir pas de tout) à cette question, dont l'impact peu néanmoins être très important sur ces mécanismes. Un exemple de cet impact se trouve justement au sein du projet MUSIC. Celui-ci se base sur la définition des « fonctions d'utilités » (*utility function*) permettant au *middleware* de décider sur l'usage d'une implémentation ou d'une autre pour chaque composant de l'application²³ [123,218]. Ces fonctions étant conçues par le concepteur de l'application lui-même, elles peuvent être particulièrement sensibles à l'absence d'une information de contexte. Si on peut supposer que les composants directement chargés sur le *middleware* ont pu être conçus par le même concepteur de l'application, le même ne peut pas être dit sur les services issus d'autres fournisseurs pouvant être intégrés à cette application. On peut donc facilement se retrouver avec des informations de contexte inconnues (soit par des problèmes de lecture, soit tout simplement parce que non prévues par le concepteur), mais nécessaires au bon fonctionnement de certains services.

Une autre particularité de ces travaux repose sur la présence d'un contexte « requis » par un service : on analyse le contexte de l'utilisateur face à des « contraintes » propres au service, représentées par ce contexte « requis ». Chaque service a ainsi des « exigences » par rapport au contexte de son client, c'est-à-dire le service est proposé pour être exécuté idéalement dans un certain contexte. Il peut tout à fait être exécuté sur d'autres contextes, mais les résultats ne seront peut-être pas optimaux. Le contexte d'exécution côté client est ainsi comparé à ce contexte demandé (ou « idéal ») pour le

²³ Note : le *middleware* MUSIC [218] utilise à la fois la notion de composant et celle de service. La première est utilisée notamment pour représenter des unités d'exécution locales, alors que les services sont vus comme des unités d'exécution présentes dans l'environnement.

service, les deux descriptions étant vues comme deux graphes et comparés à l'aide de mesures de similarité mentionnées ci-dessus. Il s'agit d'une approche assez inhabituelle pour la littérature, dans laquelle régnait, à l'époque de ces travaux (2008), surtout l'idée que seul le client peut porter des exigences face à un service. A nouveau, le projet MUSIC en est l'illustration : c'est l'application (le client donc) qui définit ses exigences, à travers les fonctions d'utilité, pour l'usage de ses composants ou des services extérieurs. On peut néanmoins citer [244] comme un exemple de proposition supposant aussi des exigences côté service. En effet, pour [244], aussi bien l'utilisateur que le service possèdent des préférences concernant l'information de contexte, préférences utilisées par la plateforme pour l'adaptation des services.

Enfin, on peut également percevoir ici l'influence des travaux réalisés précédemment, notamment les mesures de similarité. Même si celles-ci n'ont pas été utilisées directement, elles ont servi d'inspiration et ont été adaptées à l'orientation service dans le cadre du projet MUSIC. A leur tour, les présents travaux ont également influencé d'autres, comme la thèse de Salma Najjar [170] (dont l'apport est discuté dans les prochaines sections), où on voit réapparaître un contexte requis dans la description des services. La mise en œuvre aussi suit la même inspiration, avec un élément « *context* » ajouté à la description de service et pointant sur une description extérieure en XML, qui peut être mise à jour plus facilement.

3.2.3 Bibliométrie

Les travaux décrits ici ont donné lieu à trois publications, dont un chapitre de livre. Par ailleurs, ces travaux ont permis la réalisation d'une quatrième publication (Chapitre IGI 2009), où une prise en charge globale de l'adaptation au contexte est analysée. Ces publications sont listées ci-dessous. Elles ont été organisées en groupes²⁴ (voir Tableau 7), en fonction de la date de leur publication (antérieures à 2013, entre 2013 et 2016, et postérieures 2016), en plus des autocitations. La Figure 40 illustre ces résultats.

- **NFPSLA-SOC 2008** [135] : Kirsch-Pinheiro, M.; Vanrompay, Y. & Berbers, Y., « Context-aware service selection using graph matching ». In: Paoli, F. D.; Toma, I.; Maurino, A.; Tilly, M. & Dobson, G. (Eds.), *2nd Non Functional Properties and Service Level Agreements in Service Oriented Computing Workshop (NFPSLA-SOC'08), at ECOWS 2008*, CEUR Workshop proceedings, vol. 411, **2008**.
- **CAMPUS 2009** [255] : Vanrompay, Y.; Kirsch-Pinheiro, M. & Berbers, Y., "Context-Aware Service Selection with Uncertain Context Information", *Context-Aware Adaptation Mechanism for Pervasive and Ubiquitous Services 2009 (CAMPUS 2009)*, *Electronic Communications of the EASST*, vol. 19, **2009**.
- **Chapitre IGI 2011** [256] : Vanrompay, Y.; Kirsch-Pinheiro, M. & Berbers, Y., "Service Selection with Uncertain Context Information", In: Stephan Reiff-Marganiec and Marcel Tilly (Eds.), *Handbook of Research on Service-Oriented Systems and Non-Functional Properties: Future Directions*, IGI Global, pp. 192-215, **2011**.
- **Chapitre IGI 2009** [202] : Preuveneers, D.; Victor, K.; Vanrompay, Y.; Rigole, P.; Kirsch Pinheiro, M. & Berbers, Y. « Context-Aware Adaptation in an Ecology of Applications ». In: Dragan Stojanovic (Ed.), *Context-Aware Mobile and Ubiquitous Computing for Enhanced Usability: Adaptive Technologies and Applications*, IGI Global, **2009**.

²⁴ Comme pour les sections précédentes, ces publications ont été analysées, à l'aide du site scholar.google.com, par rapport au nombre de citations. Les chiffres présentés dans le Tableau 7 sont issus de cette analyse.

La première de ces publications (NFPSLA-ECOWS 2008) est celle qui décrit les principes de ce mécanisme. Les publications suivantes (CAMPUS 2009 [255] et Chapitre IGI 2011 [256]) ont permis par la suite de l'améliorer, en ajoutant la prise en compte de la qualité par les métadonnées et les fonctions de transformation. La dernière de ces publications (Chapitre IGI 2011 [256]) est celle comptant le plus d'autocitations, car elle résume, de manière assez complète, les contributions des publications précédentes. Elle a ainsi été utilisée comme référence pour les travaux postérieurs à celui-ci.

Tableau 7. Nombre de citations des publications concernées par ces travaux.

Référence	Année	Total	≤ 2013	> 2013 et ≤ 2016	> 2016	self
NFPSLA-ECOWS 2008	2008	43	22	11	3	7
CAMPUS 2009	2009	9	1	4	4	0
Chapitre IGI 2009	2009	19	7	6	0	6
Chapitre IGI 2011	2011	11	0	3	0	8
Total / %		78	28,21 %	41,03 %	3,85 %	26,92 %

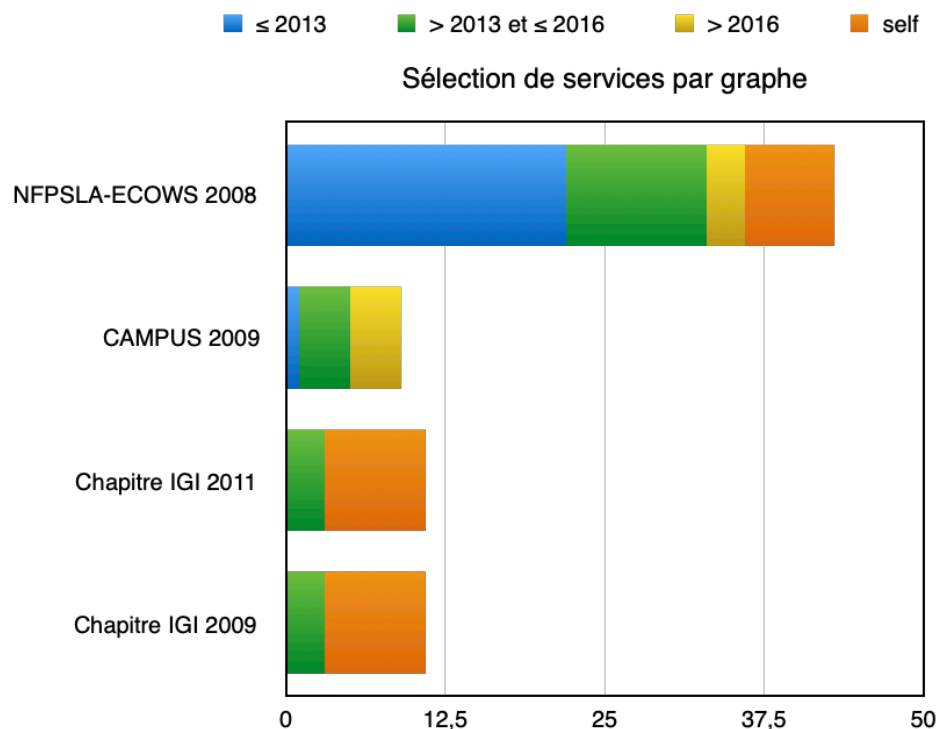


Figure 40. Citations par publication.

Parmi ces publications, celle comptabilisant le plus de citations demeure la première, présentant les principes fondateurs du mécanisme. Ces citations s'étalent notamment sur les 8 premières années suivant sa publication (de 2008 à 2016), pour s'estomper après cette date. Cette période (2008 à 2016) correspond à une période où l'orientation service a été très largement discutée dans l'Informatique

Pervasive. Aujourd'hui, la notion de service, telle que décrit dans la communauté SOA, a peu à peu cédé sa place à la notion de micro-service. Même si conceptuellement, les deux notions sont fondamentalement semblables, la notion de micro-service est perçue aujourd'hui comme plus avantageuse car plus adaptée à l'implémentation dans un contexte de virtualisation et de *Cloud Computing*. Quoiqu'il en soit, les propositions présentées par ces travaux restent valables et pourraient, le cas échéant, être adaptées à l'univers micro-service.

3.3 Sélection de service par l'intention et le contexte

3.3.1 Rappel de la problématique

L'introduction des nouvelles technologies (*smartphones*, IoT, Big Data, etc.) entraîne des profonds changements dans les organisations et leurs Systèmes d'Information (SI). Ceux-ci font désormais face à un environnement devenu pervasif. Ces systèmes et leurs utilisateurs sont ainsi confrontés à une hétérogénéité grandissante qu'ils doivent dorénavant gérer. Puisque ces systèmes se transforment et deviennent de plus en plus complexes par l'arrivée de nouvelles technologies, il devient capital de pouvoir masquer l'hétérogénéité de celles-ci et des services offerts par ces systèmes afin que les utilisateurs puissent se concentrer davantage sur les objectifs à atteindre que sur ces technologies. Plus que jamais, ces systèmes ont besoin d'évoluer, pour que la vision donnée par Weiser [264], d'une informatique ubiquitaire, invisible à l'utilisateur, devienne une réalité. Ces nouveaux systèmes, qu'on peut appeler SI Pervasifs (et qui seront traités en profondeur dans le prochain chapitre), ont ainsi un besoin accru de transparence. L'objectif de ces travaux est donc de renforcer la transparence des SI en cachant aux utilisateurs les détails techniques concernant les services invoqués, à travers la prise en compte de la notion de contexte, mais aussi la prise en compte des objectifs métiers de l'utilisateur, représentés par la notion d'intention.

3.3.2 Contribution & Impact

Réalisés entre 2009 et 2012 dans le cadre de la thèse de Salma Najar [170], ces travaux proposent un mécanisme de sélection de services tenant compte à la fois du contexte d'utilisateur et de son intention. La notion d'intention correspond à une notion largement reconnue dans le domaine de l'Ingénierie des Besoins [269, 115, 252, 213]. Elle y est utilisée notamment pour représenter les objectifs métiers de l'utilisateur et a déjà été associée à la notion de service par le passé [84, 121, 164, 214, 221].

L'utilisation de la notion d'intention dénote le focus résolument centré utilisateur adopté par les présents travaux et qui les distingue des travaux précédents sur la sélection de services (cf. section 3.2). Même si les travaux précédents tenaient compte du contexte dans lequel un service était sollicité, ils considéraient surtout le contexte d'une application cliente qui demande l'exécution d'un service, alors dans les présents travaux, on considère directement l'utilisateur, avec ses besoins et son contexte d'utilisation. Le focus principal pour l'analyse de la notion de contexte est donc l'utilisateur lui-même. On retrouve ce même focus centré utilisateur dans les travaux sur les Systèmes d'Information Pervasifs qui seront discutés plus tard (cf. Chapitre 4).

La notion d'intention est au cœur de la première hypothèse sur laquelle se base ces travaux. En effet, trois hypothèses majeures guident les présents travaux. Tout d'abord, l'idée selon laquelle un utilisateur demanderait la réalisation d'un service afin de satisfaire un objectif. Autrement dit, c'est parce qu'il a un besoin précis qu'un utilisateur demande la réalisation d'un service à un Système d'Information.

La deuxième hypothèse sur laquelle se base ces travaux est à peu près la même qui celle guidant les travaux sur la sélection de services par graphe (cf. section 3.2), c'est-à-dire qu'un service peut disposer de plusieurs implémentations distinctes, plus ou moins adaptées à un certain contexte d'usage. En

d'autres termes, on suppose qu'un service est conçu pour être invoqué idéalement dans un certain contexte d'utilisation. Ceci n'empêche pas son usage dans d'autres situations, mais les résultats peuvent ne pas être optimaux.

Enfin, la troisième hypothèse guidant ces travaux est celle qui les caractérise au mieux. Elle consiste à dire qu'un besoin émerge dans un certain contexte, lequel influence à son tour le choix de l'implémentation. En d'autres termes, un utilisateur invoquerait un service parce qu'il a un besoin métier (donc un but à atteindre), lequel émerge dans un certain contexte, qui va à son tour être déterminant pour le choix de l'implémentation à exécuter. La Figure 41 schématise cette hypothèse avec une intention qui surgit dans un contexte, dans lequel elle est « valable ». Ce contexte va ainsi favoriser l'apparition de cette intention, laquelle pourra être satisfaite par un service. Celui-ci s'exécute dans un contexte (C_χ à la Figure 41), mais il suppose également un contexte requis ($C_{\chi R}$ à la Figure 41) pour mieux satisfaire cette intention.

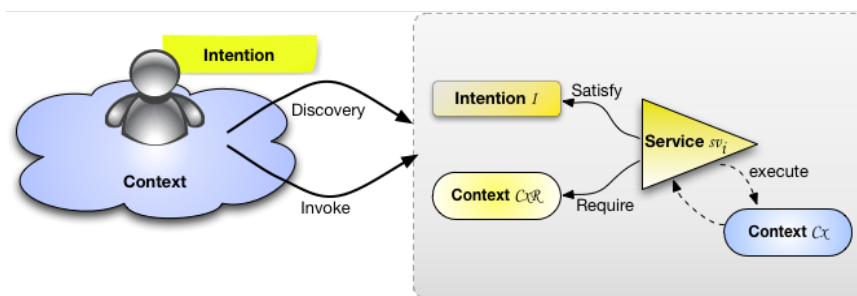


Figure 41. Illustration de la relation entre un service, une intention et un contexte [175].

Les deux premières hypothèses sont partagées avec d'autres travaux dans la littérature. Pour la première, on peut souligner les travaux sur les services intentionnels dans le domaine des SI, tels que [121,214,164], lesquels associent à chaque service un but, représenté par une intention. Pour la seconde, on peut souligner les nombreux travaux sur des services sensibles au contexte dans l'Informatique Pervasive (e.g. [22, 244,248]), lesquels associent la notion de contexte aux services ou à leur usage. Par contre, la troisième hypothèse est particulière à nos travaux. Ceux-ci ont été parmi les premiers (en 2009) dans la mise en relation entre les intentions d'un utilisateur, son contexte d'utilisation et les services invoqués. On peut citer [221, 84] en tant qu'exemples de travaux combinant la notion d'intention à celle de contexte dans le cadre d'un service. Cependant, il convient de souligner que contrairement à ces travaux, les nôtres proposent une description sémantique détaillée aussi bien de la notion d'intention que celle de contexte, ce qui n'est pas forcément le cas d'autres travaux dans la littérature tels que [221, 84] qui ciblent davantage l'un ou l'autre de ces aspects.

Ainsi, à l'instar de [244, 248, 22] et même [256], les présents travaux proposent une modélisation sémantique des services, à travers un descripteur sémantique en OWL-S [159] étendu. Par contre, contrairement à ces travaux [244, 248, 22], les nôtres relient un service aux intentions qu'il permet de satisfaire en plus de son contexte d'exécution et de son contexte requis. On retrouve à nouveau, l'élément « context » qui renvoie à une description extérieure pouvant être mise à jour rapidement (comme dans nos travaux précédents sur la sélection de service, cf. section 3.2), accompagné d'une description sémantique de l'intention (ou des intentions, si composition) associée(s) au service. La Figure 42 illustre des éléments de cette description, à la fois intentionnelle (Figure 42a) et contextuelle (Figure 42b) d'un service. Dans cette figure, on retrouve la description d'une intention qui est associée à un service (Figure 42a), ainsi qu'un contexte requis (Figure 42b) dans lequel ce service serait le plus apte à satisfaire l'intention associée.

<pre> 1 ... 2 <service:Service rdf:ID="Parachute_Service"> ... </service:Service> 3 <profile:Profile rdf:ID="Parachute_Service"> 4 ... 5 <profile:serviceName xml:lang="en"> ContextParachuteService </profile:serviceName> 6 <profile:textDescription xml:lang="en"> 7 This service provides a facility to organise and book parachute jumping sessions. 8 </profile:textDescription> 9 ... 10 <eipfile:context rdf:resource="http://193.55.96.54/iSOA/ExtensionOWL-S/ ... ContextDescription.xml#condition1"/> 11 ... 12 <eipfile:Intention rdf="http://193.55.96.54/iSOA/ExtensionOWL-S/Intention.owl#intention"> 13 <eipfile:Verb rdf="http://193.55.96.54/iSOA/ExtensionOWL-S/ ... Intention.owl#intention.verb.make"> Make </eipfile:Verb> 14 <eipfile:Target rdf="http://193.55.96.54/iSOA/ExtensionOWL-S/ ... Intention.owl#intention.target.object.sport"> 15 Parachute jumping 16 </eipfile:Target> 17 <eipfile:Parameter rdf="http://193.55.96.54/iSOA/ExtensionOWL-S/ ... Intention.owl#intention.parameter"> 18 <eipfile:Mean rdf="http://193.55.96.54/iSOA/ExtensionOWL-S/ ... Intention.owl#intention.way.mean.internet"> 19 On-line reservation 20 </eipfile:Mean> 21 </eipfile:Parameter> 22 </eipfile:Intention> 23 ... </pre>	<pre> 1 <?xml version="1.0" encoding="UTF-8"?> 2 <ctx:context xmlns:ctx="http://193.55.96.54/iSOA/ContextSchema" 3 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" 4 xmlns:xs="http://www.w3.org/2001/XMLSchema" 5 xsi:schemaLocation="http://193.55.96.54/iSOA/ContextSchema ContextSchema.xsd"> 6 <ctx:conditionSet> 7 <ctx:condition rdf:ID="condition1"> 8 <ctx:contextElement> 9 <ctx:hasEntity ... resource="http://193.55.96.54/iSOA/Context.owl#concept.entityType.user"/> 10 <ctx:hasScope ... resource="http://193.55.96.54/iSOA/Context.owl#concept.contextScope.userprofile"/> 11 ... 12 <ctx:contextValueSet> 13 <ctx:contextValue> 14 <ctx:hasScope ... resource="http://193.55.96.54/iSOA/Context.owl#concept.contextScope.profile.age"/> 15 ... 16 <ctx:value> [18-60] </ctx:value> 17 </ctx:contextValue> 18 </ctx:contextValueSet> 19 <ctx:contextElement> 20 <ctx:contextElement> 21 <ctx:hasEntity ... resource="http://193.55.96.54/iSOA/Context.owl#concept.entityType.user"/> 22 <ctx:hasScope ... resource="http://193.55.96.54/iSOA/Context.owl#concept.contextScope.location"/> 23 ... 24 <ctx:contextValueSet> 25 <ctx:contextValue> 26 <ctx:hasScope ... resource="http://193.55.96.54/iSOA/Context.owl#concept.contextScope.location.city"/> 27 ... 28 <ctx:value>Grenoble, France</ctx:value> 29 </ctx:contextValue> 30 </ctx:contextValueSet> 31 </ctx:contextElement> 32 </ctx:condition> 33 <ctx:condition rdf:ID="condition2"> ... </ctx:condition> 34 </ctx:conditionSet> 35 <ctx:state> ... </ctx:state> 36 </ctx:context> </pre>
--	---

(a)

(b)

Figure 42. Description d'une intention (a) et d'un contexte requis (b) associés à un service [172].

Cette description sémantique des intentions s'inspire des travaux de [121, 213, 198]. Elle suppose l'existence d'une ontologie de domaine décrivant les intentions, leurs synonymes, hypéronymes et hyponymes. Cette ontologie, dont un extrait décrivant une intention « *prepare a proposal* » est présentée dans la Figure 43, est particulièrement importante pour la désambiguïsation de l'intention, qui pourrait être exprimée de différentes manières. Ceci implique, en revanche, une application en « monde fermé », dans notre cas, un Système d'Information, où on peut supposer l'existence d'un tel référentiel. D'autres travaux, comme [164] utilisent une démarche semblable, supposant l'application des services intentionnels à un domaine précis (le domaine médical, dans le cas de [164]), où une ontologie de domaine peut aisément être envisageable. En d'autres circonstances, des efforts ou des aides pour la formulation (ou la reformulation) des intentions, tels que suggéré par [6], peuvent être considérés afin d'aider (ou de guider) les utilisateurs dans l'expression de ces intentions.

```

1 <intention:Intention rdf:ID="PREPARE_PROPOSAL_INTENTION">
2
3 <!--The principal Intention of the service -->
4
5 <intention:PrincipalIntention rdf:ID="PREPARE_PROPOSAL_PRINCIPAL_INTENTION">
6 <intention:Verb rdf:resource="http:// www.crinfo.univ-paris1.fr/ExtensionOWL-S
7 /Intention.owl#concept.intention.verb.prepare">
8   Prepare
9 </intention:Verb>
10
11 <intention:Target rdf:resource="http:// www.crinfo.univ-paris1.fr/ExtensionOWL-S/
12 Intention.owl#concept.intention.target">
13 <intention:Result rdf:resource="http:// www.crinfo.univ-paris1.fr/ExtensionOWL-S/
14 Intention.owl#concept.intention.target.result.Proposal">
15   Proposal
16 </intention:Result>
17 </intention:Target>
18 </intention:PrincipalIntention>
19
20 .....
21
22 </intention:Intention>

```

Figure 43. Extrait de l'ontologie d'intention [170].

Du même, la description d'un contexte d'exécution et d'un contexte requis associés à un service a demandé la formalisation de la notion de contexte à travers un méta-modèle générique, illustré par la Figure 44, donnant lieu à un modèle lui-même exprimé sous la forme d'une ontologie [170, 169]. Cette modélisation s'inspire de [262] ainsi que de nos travaux sur la modélisation de contexte (cf. Chapitre 1). Comme ces travaux, ce méta-modèle est fondé sur le principe de la séparation claire entre le sujet observé et les éléments de contexte (nommés respectivement « sujet observé » et « élément de contexte » dans le méta-modèle présenté dans la Figure 44), mais contrairement aux travaux précédents, l'ontologie proposée se présente sous la forme d'un modèle multi-niveaux comprenant un premier niveau composé de concepts généraux (e.g. profil utilisateur, activité, localisation, connexion réseau, etc.), et d'un ou plusieurs niveaux dans lesquels des éléments de contexte plus spécifiques sont définis (e.g. température, latence, etc.), illustrés par la Figure 45. Cette organisation par niveau facilite la réutilisation et promeut l'extensibilité du modèle, lequel supporte également, grâce au méta-modèle, une composition hiérarchique d'éléments de contexte [170], ce qui n'était pas possible, par exemple, avec les modèles [262] et [137].

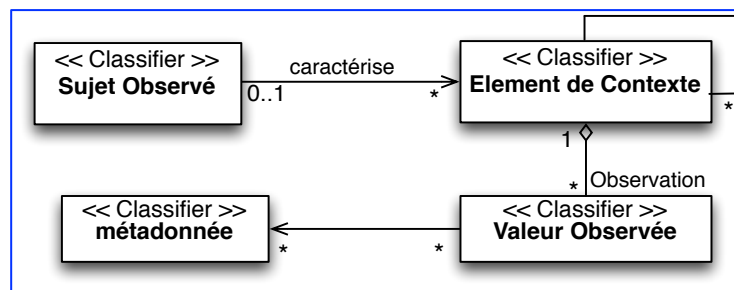


Figure 44. Méta-modèle de contexte proposé dans [170].

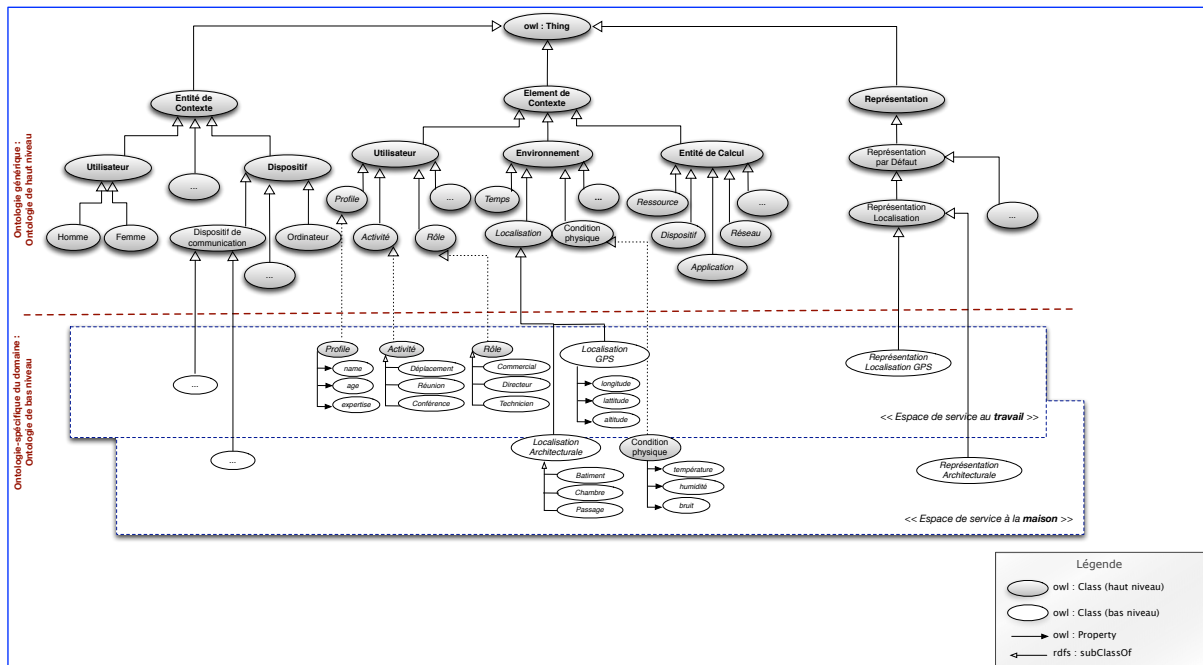


Figure 45. Ontologie multi-niveaux de contexte issu de [170].

En ce qui concerne le mécanisme de sélection lui-même, il s'agit d'un mécanisme à deux étapes, tirant profit aussi bien de l'ontologie d'intention que celle décrivant les éléments de contexte. Dans la première étape, l'intention demandée par l'utilisateur et passible d'être satisfaite par les services

offerts est analysée et mise en relation avec celles déclarées par les services (à travers leurs descripteurs). Différentes opérations sont utilisées pour cette mise en relation (*exact*, *plugin*, *subsumes*, *synonym*, *hyponym*, *hypernym*, *fail*), lesquelles se basent sur les relations présentes dans l'ontologie. L'intention est décomposée en verbe et cible, lesquels sont analysés séparément pendant cette première étape, à l'aide de ces opérations. Même si, dans la description de service, la notion d'intention est décrite de manière très détaillée, inspiré du formalisme défini par [198], seuls ces deux aspects (verbe et cible) sont pour l'instant pris en considération. La composition des intentions notamment n'est pas encore prise en compte dans le processus de sélection. Par ailleurs, il convient d'observer à travers cette première étape que la recherche d'un service se fait directement par son intention, et non plus selon les fonctionnalités offertes par celui-ci, comme traditionnellement dans l'approche SOA. La description fonctionnelle d'un service (correspondant aux fonctionnalités attendues par le service, exprimées à travers les opérations qu'il est capable de réaliser) demeure ainsi cachée de l'utilisateur, qui cherche un service uniquement à travers son intention (c'est-à-dire son objectif métier).

Dans la seconde étape, le mécanisme compare, à l'aide de l'ontologie de contexte, les éléments de contexte présents dans le contexte requis par le service et ceux présents dans le contexte d'utilisation courant, avant d'évaluer la satisfaction des conditions exprimées dans le contexte requis pour ces éléments. Ces conditions sont exprimées à travers un langage de requête simple, inspirée de [210]. Les valeurs présentes dans le contexte d'utilisation courant (côté utilisateur) sont ainsi confrontées à ces conditions, afin d'évaluer leur niveau de satisfaction. Le score obtenu par chaque service sur les deux étapes est ensuite comptabilisé et utilisé pour classer les services disponibles. Il convient alors d'observer que, à l'instar de nos travaux précédents (cf. section 3.2), un service peut être proposé ici à l'utilisateur même si son contexte requis ne correspond pas totalement au contexte courant de l'utilisateur.

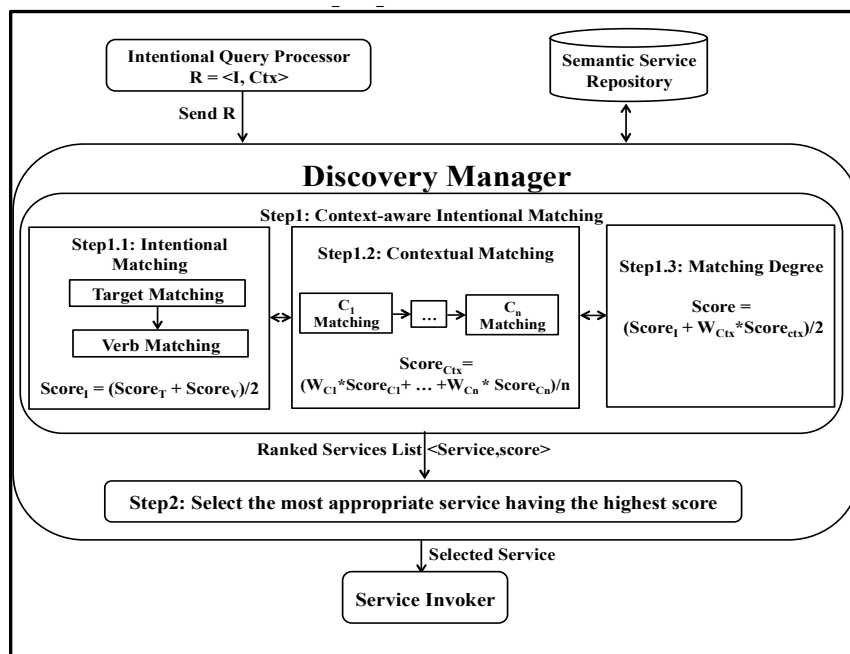


Figure 46. Vision globale du mécanisme de sélection de service par intention et contexte [181].

La Figure 46 résume ainsi le processus de sélection proposé : celui-ci dispose d'un ensemble de services disponibles (illustrés par le répertoire en haut à droite de la Figure 46), sémantiquement décrits avec une (ou plusieurs intentions) intentions et un contexte, suivant les ontologies illustrés par la Figure 42.

Il reçoit en entrée une requête composée d'une intention souhaitée par l'utilisateur et de son contexte courant (en haut à gauche de la Figure 46). Celle-ci est utilisée dans les deux phases décrites ci-dessus, d'abord avec l'analyse de l'intention (« *intentional matching* » dans la Figure 46), puis celle du contexte (« *contextual matching* » dans la Figure 46). Chaque phase donne lieu à un score qui est utilisé pour le calcul du score global affecté au service, selon lequel il est classé par rapport à la demande. Il convient d'observer dans Figure 46 l'usage d'un coefficient aussi bien pour les éléments de contexte de manière individuelle (W_{ci}) que pour le score obtenu dans la 2^{ème} phase (W_{ctx}), ce qui permet de paramétrer le mécanisme en régulant l'influence de l'information de contexte dans le résultat final.

Le mécanisme proposé se trouve ainsi sur la même ligne des travaux comme [188], qui proposent l'usage des relations *exact*, *plugin* et *subsumes* pour la découverte de services, ou encore [210], qui proposent un langage de requête dédié aux informations de contexte, basé sur le modèle proposé par [262].

L'innovation dans ces travaux de sélection de service proposés dans la thèse de Salma Najar réside notamment dans la mise en place du triplet < *intention*, *contexte*, *service* >. Le mécanisme de sélection cherche en effet à satisfaire une intention exprimée par l'utilisateur et dont le service se déclare comme étant capable de satisfaire. Puis, il classe les services, en cherchant ceux les mieux adaptés par rapport au contexte courant de l'utilisateur. On considère ainsi qu'un service est proposé pour satisfaire une intention dans un certain contexte, où cette intention émerge. Cette proposition, issue de la troisième hypothèse mentionnée précédemment, va donc plus loin qu'une vision purement intentionnelle, telle que [121, 164, 214], ou purement contextuelle [22, 244, 248], vers une vision systémique où ces deux aspects sont indissociables. Il s'agit d'une vision centrée utilisateur dans laquelle la recherche se fait par l'intention de cet utilisateur, placée dans un contexte donné, et non plus par les fonctionnalités proposées par un service. L'utilisateur (et non plus simplement une application cliente ou un dispositif) est remis au centre des préoccupations à travers son intention, sans négliger le contexte dans lequel cette intention émerge.

3.3.3 Bibliométrie

Comme pour les travaux discutés précédemment, ceux-ci ont également donné lieu à plusieurs publications (au total de 7), dont une revue. Ces travaux sont aussi mentionnés sur deux autres publications incluant également nos travaux sur la prédiction de service (cf. section 3.4). Celles-ci ont donc été comptabilisés dans la prochaine section. Les publications²⁵ ainsi sélectionnées ont été organisées en différents groupes (voir Tableau 8), en fonction de leur date de publication (antérieures à 2013, entre 2013 et 2016, et postérieures 2016), en plus des autocitations (« *self* »). La Figure 47 illustre ces résultats obtenus.

- **CIAO 2009** [171] : Najar, S.; Saidani, O.; Kirsch-Pinheiro, M.; Souveyet, C. & Nurcan, S., "Semantic representation of context models: a framework for analyzing and understanding". In: J. M. Gomez-Perez, P. Haase, M. Tilly, and P. Warren (Eds), *Proceedings of the 1st Workshop on Context, information and ontologies (CIAO 09), European Semantic Web Conference (ESWC'2009)*, Heraklion, Greece, June 2009, ACM, pp. 1-10.
- **Service 2011** [173] : Najar, S.; Kirsch Pinheiro, M. & Souveyet, C., "Bringing context to intentional services". *3rd Int. Conference on Advanced Service Computing, Service Computation'11*, Rome, Italy, pp. 118-123, 2011. Best Paper Awards.
- **REFS 2011** [172] : Najar, S.; Kirsch Pinheiro, M. & Souveyet, C., "The influence of context on intentional service". *5th Int. IEEE Workshop on Requirements Engineerings for Services*

²⁵ Comme précédemment, ces publications ont été analysées, à l'aide du site scholar.google.com, par rapport à leur nombre de citations. Les chiffres résultants de cette analyse sont résumés dans le Tableau 8.

(REFS'11), *IEEE Conference on Computers, Software, and Applications (COMPSAC'11)*, Munich, Germany, pp. 470-475, **2011**.

- **WEWST 2011** [174] : Najar, S.; Kirsch Pinheiro, M. & Souveyet, C., "Towards Semantic Modeling of intentional pervasive System", *6th International Workshop on Enhanced Web Service Technologies (WEWST'11), European Conference on Web Services (ECOWS'11)*, Lugano, Switzerland, **2011**, pp. 30-34.
- **IJAIS 2012** [169] : Najar, S. ; Kirsch-Pinheiro, M. & Souveyet, C., "Enriched Semantic Service Description for Service Discovery: Bringing Context to Intentional Services", *International Journal On Advances in Intelligent Systems*, volume 5, numbers 1 & 2, June **2012**, pp. 159-174, IARIA Journals / ThinkMind, ISSN: 1942-2679.
- **ICWS 2012** [181] : Najar, S.; Kirsch-Pinheiro, M.; Souveyet, C. & Steffanel, L. A., "Service Discovery Mechanisms for an Intentional Pervasive Information System". *Proceedings of 19th IEEE International Conference on Web Services (ICWS 2012)*, Honolulu, Hawaii, 24-29 June **2012**, pp. 520-527.
- **UbiMob 2012a** [182] : Najar, S. ; Kirsch-Pinheiro, M. ; Steffanel, L. A. & Souveyet, C., « Analyse des mécanismes de découverte de services avec prise en charge du contexte et de l'intention ». In : Philippe Roose & Nadine Rouillon-Couture (dir.), *8èmes Journées Francophones Mobilité et Ubiquité (UbiMob 2012)*, June 4-6, **2012**, Anglet, France. Cépaduès Editions, pp. 210-221. ISBN 978.2.36493.018.6.

Tableau 8. Analyse bibliométrique des publications liées à la sélection de services par intention et contexte.

Référence	Année	Total	≤ 2013	> 2013 et ≤ 2016	> 2016	self
CIAO 2009	2009	51	14	8	11	18
Service 2011	2011	6	2	2	0	2
REFS 2011	2011	13	1	2	2	8
WEWST 2011	2011	7	0	3	0	4
IJAIS 2012	2012	4	0	0	1	3
ICWS 2012	2012	18	4	4	3	7
UbiMob 2012a	2012	1	0	0	0	1
Total / %		100	21,00 %	19,00 %	17,00 %	43,00 %

Parmi ces publications, celle comptant le plus de citations (CIAO 2009 [171]) correspond à un cadre permettant l'analyse et la comparaison entre différents modèles de contexte. Celui-ci a été largement utilisé sur nos travaux ultérieurs justement pour l'analyse des modèles de contexte proposés. Le fait de cibler l'analyse de modèles de contexte de manière générale peut également expliquer l'impact plus significatif, en matière de citations, de cet article par rapport aux autres publications mentionnées ici. Il convient également de souligner que le nombre significatif d'autocitations dans ces travaux se justifie par leur continuité, beaucoup parmi eux ayant servi de base à des travaux ultérieurs.

Enfin, on observe également que la plupart des citations se concentre dans les premiers 4-6 ans suivant les publications (de 2009 à 2016), ce qui correspond aussi à une période où la sélection de services

correspondait à une problématique de recherche encore largement traitée dans la littérature en Informatique Pervasive.

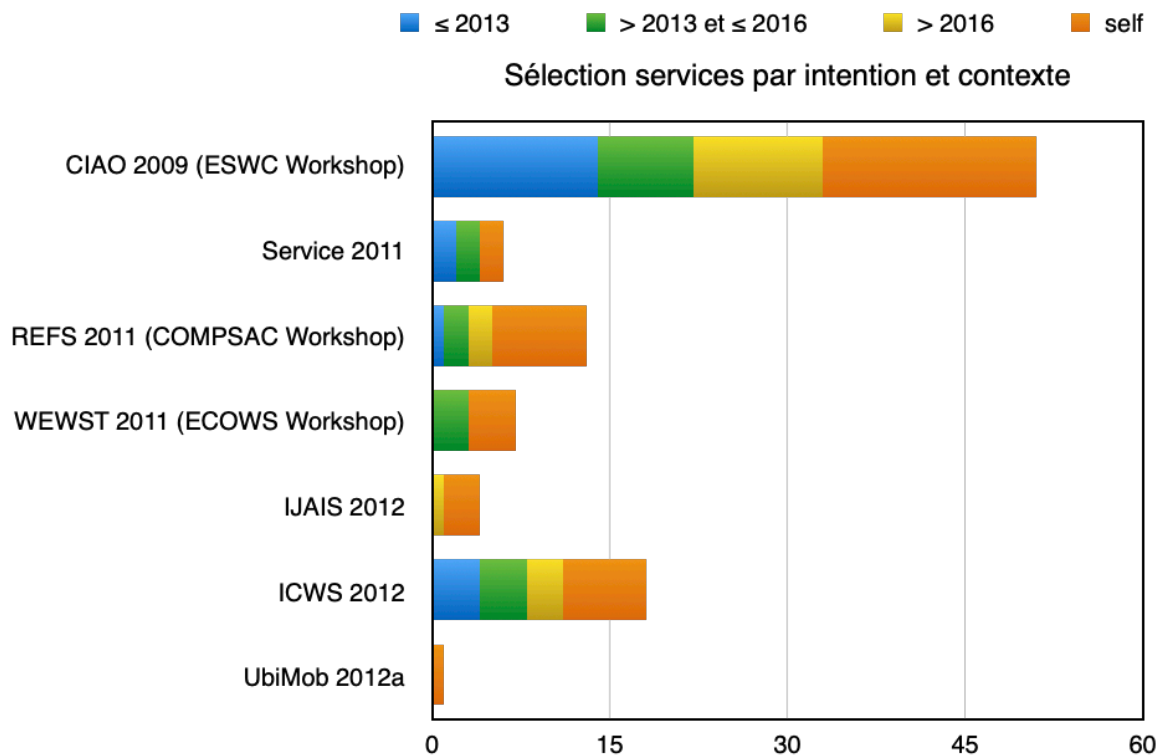


Figure 47. Évolution des citations dans le temps par référence.

3.4 Prédiction de services

3.4.1 Rappel de la problématique

Toujours dans le cadre de l'évolution des Systèmes d'Information (SI) vers des SI Pervasifs, ces nouveaux systèmes se caractérisent, entre autres, par un besoin accru de transparence, conformément à ce que nous avons discuté précédemment (cf. section 3.3). Celle-ci est plus que jamais nécessaire pour cacher des utilisateurs finaux la complexité technologique qui caractérise désormais les SI. Les utilisateurs ainsi libérés des contraintes technologiques pourraient, on l'espère, se concentrer davantage sur les tâches ajoutant de la plus-value à l'organisation. Pour cela, un autre besoin se fait également sentir : la proactivité. En effet, les utilisateurs attendent un SI de plus en plus « intelligent », capable d'anticiper leurs besoins et d'y répondre de manière appropriée. Selon Bauer & Dey [16], nous pouvons déjà témoigner d'une tendance vers des systèmes de plus en plus sophistiqués, appelés « intelligents », « sensibles au contexte », « adaptatifs », etc., pour lesquels la notion de contexte est centrale, au fur et à mesure qu'ils prennent conscience du contexte dans lequel ils sont utilisés et, de manière intelligente, adaptent leur exécution. Nous pensons que la démocratisation de ce type de comportement qu'on pourrait considérer comme « intelligent » fait naître chez les utilisateurs une certaine attente : ils attendent désormais des logiciels et des systèmes qu'ils soient plus intelligents, qu'ils puissent reconnaître leur situation, leur comportement et qu'ils s'adaptent de manière aussi réactive que proactive. Il nous paraît donc évident que les Systèmes d'Information doivent désormais comprendre davantage les habitudes de leurs utilisateurs afin de pouvoir anticiper leurs besoins et leur proposer de manière proactive les services correspondant à leurs besoins.

3.4.2 Contribution & Impact

Tout comme les travaux sur la sélection intentionnelle et contextuelle de services (cf. section 3.3), les travaux sur la prédiction de services font également partie de la thèse de Salma Najjar [170]. Dans ces travaux, réalisés entre 2012 et 2014, un mécanisme de prédiction de services est proposé. Celui-ci permet, à partir de l'historique et du contexte courant de l'utilisateur, d'anticiper les prochaines intentions et contexte d'usage de celui-ci et d'ainsi fournir, de manière proactive, le prochain service que cet utilisateur aurait probablement demandé. L'objectif ici serait de puiser dans les traces d'usage et sur le contexte d'utilisation des séquences d'intentions et contextes récurrents et, à travers ces informations, pouvoir anticiper les sollicitations de l'utilisateur, lui proposant ainsi un système qui répond de manière proactive à ses besoins.

L'hypothèse de base de cette proposition est celle de l'habitude : au cours de ses activités professionnelles, un utilisateur d'un SI peut acquérir certaines habitudes de travail, liées à sa routine professionnelle (par exemple, un commercial qui sollicite sur sa tablette des données sur le client en arrivant chez celui-ci). On considère ainsi qu'il est possible de caractériser, et donc d'identifier automatiquement, ces habitudes à travers les intentions derrière ces actions (le pourquoi avoir sollicité un service) et le contexte dans lequel ces intentions surgissent.

Il convient d'observer que les présents travaux de prédiction considèrent également le triplet $\langle \textit{intention}, \textit{contexte}, \textit{service} \rangle$ à l'instar de nos travaux sur la sélection de service intentionnelle et contextuelle, présentés précédemment (cf. section 3.3). En outre, on considère ici les traces issues de ce mécanisme de sélection.

Ainsi, à partir des traces des sollicitations précédentes, en appliquant un algorithme de *clustering*, le mécanisme proposé va d'abord regrouper les situations semblables déjà observées, en les comparant à l'aide de différentes mesures de similarité. Chaque situation est représentée par le triplet $\langle \textit{intention}, \textit{contexte}, \textit{service} \rangle$ issu d'une précédente demande de l'utilisateur (et donc d'un processus précédent de sélection de service). Puis, ces *clusters* sont organisés, à l'aide d'un modèle de chaîne de Markov, lequel calcule, à partir des données connues sur ces situations, la probabilité de passer d'un *cluster* (c.a.d. d'une situation « type ») à un autre. Grâce à ce modèle, et à partir du moment où la situation courante (c.a.d. l'intention exprimée, le contexte courant et le service sélectionnée pour répondre à cette demande) est suffisamment semblable à un *cluster* déjà identifié, il est possible d'anticiper la prochaine situation (intention, contexte et service) la plus probable.

La Figure 48 illustre ainsi les différentes étapes de ce mécanisme. Dans la Figure 48a, nous avons une vue globale du mécanisme, avec, au début, la phase de *clustering*, qui regroupe les situations observées dans l'historique en plusieurs clusters, comme l'indique la Figure 48b. Chacun de ces *clusters* est identifié par un « *centroïde* » qui représente au mieux l'ensemble de situations réunies dans le *cluster*. Les résultats de cette phase de *clustering* servent d'entrée pour la phase de classification (représentée dans la Figure 48c), laquelle, à partir de ces *clusters* et de l'historique, constitue un modèle de « comportement » de l'utilisateur, correspondant au modèle de chaîne de Markov mentionné précédemment. Ce modèle est enfin utilisé dans le processus de prédiction proprement parlé (illustré par la Figure 48d), lequel prend en entrée ce modèle de comportement, ainsi que l'intention et le contexte courant de l'utilisateur, qu'il essaie de placer dans un des *clusters* présents dans ce modèle pour essayer d'identifier la prochaine situation (représentée par le triplet $\langle \textit{intention}, \textit{contexte}, \textit{service} \rangle$ du *centroïde* correspond au prochain *cluster* retrouvé à travers ce modèle).

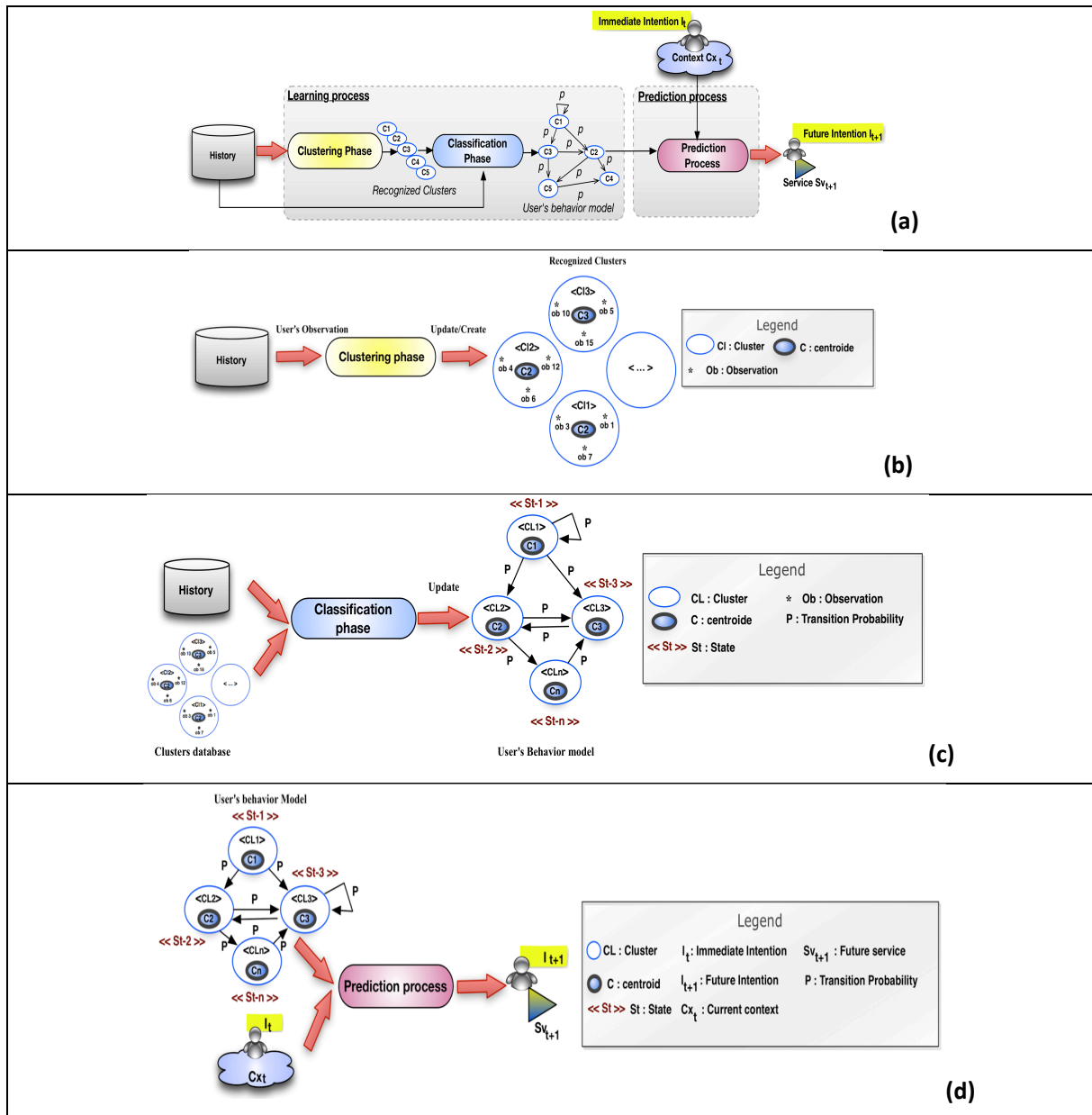


Figure 48. Différentes étapes du processus de prédiction de service proposé [178].

Le fait d'anticiper le triplet $\langle intention, contexte, service \rangle$ répond ainsi à l'hypothèse de l'habitude énoncée précédemment, mais également à l'hypothèse fondatrice du travail de Salma Najar, déjà employée dans les travaux sur la sélection de services (cf. section 3.3). Celle-ci établit qu'une intention émerge dans un contexte particulier, lequel influence le choix de service le plus apte à satisfaire l'intention. Ainsi, même si, *in fine*, ce mécanisme aboutit à la prédiction du prochain service à être invoqué par l'utilisateur, ce qu'il détermine n'est pas uniquement le prochain service, mais tout le triplet $\langle intention, contexte, service \rangle$. De cette manière, à partir d'une demande de l'utilisateur, exprimée par une intention, réalisée à partir d'un contexte donné (le contexte courant de l'utilisateur), le système pourra non seulement satisfaire la demande courante (grâce à processus de sélection de service présenté à la section 3.3), mais également anticiper la prochaine intention de l'utilisateur, accompagnée d'un contexte probable pour cette demande et d'un service pouvant la satisfaire.

C'est donc bien l'intention et le contexte dans lequel celle-ci émerge qu'on anticipe ici. Ce comportement est ainsi à distinguer d'autres travaux sur la prédiction de contexte, tels que [144, 161, 230, 257], qui se concentrent sur la prédiction du prochain contexte d'usage ou d'un élément

manquant sur le contexte observé. On peut également distinguer nos travaux de ceux, comme [107], qui font de la prédiction de services sensibles au contexte. Ces auteurs vont utiliser la notion de contexte afin de proposer des services personnalisés à leurs utilisateurs. Même si la notion de contexte est bel et bien présente dans ces travaux, dont [107], la notion d'intention, motivant le choix d'un service, est souvent ignorée. Dans un registre semblable, on peut citer [268] qui proposent un modèle de prédiction à partir des chaînes de Markov (à l'instar du nôtre) permettant de prédire le cheminement d'un consommateur sur un ensemble de services proposées. Ici, même si la notion d'intention n'est pas formalisée, elle est indirectement considérée puisque ces auteurs [268] essaient d'identifier, à partir des traces, les 'produits' motivant les précédentes visites de ces consommateurs (que les auteurs associent à leurs intentions de consommation).

De manière générale, la prédiction de services peut être comparée aux systèmes de recommandation [2, 37, 196], dont l'objectif est souvent de proposer un contenu personnalisé en fonction des traces d'autres utilisateurs, mais également aux travaux de recommandation dans les *business process* [39, 76, 104]. Dans ces derniers, l'objectif est souvent de pouvoir proposer à l'utilisateur la prochaine activité à réaliser dans un certain processus, ou encore d'adapter celui-ci pour plus de flexibilité. On peut citer, par exemple, Hermsillo *et al.* [104] qui proposent l'adaptation dynamique de processus exprimés en BPEL à travers le traitement d'événements en temps d'exécution et la programmation par aspect (AOP). Cette démarche rappelle également les récents travaux sur la fouille de processus [246, 54, 69], dont l'objectif est d'identifier des fragments de processus à partir de traces d'exécution. Dans ce registre, on peut citer Becker & Intoyad [18] qui proposent d'enrichir l'analyse des traces d'activités avec des informations contextuelles. Là encore, la notion d'intention et de contexte ne sont pas réunies dans une démarche d'anticipation, soit d'une activité ou d'un service. A ce jour, à notre connaissance, nos travaux restent un des rares à avoir combiné ces deux éléments (contexte et intention) dans le cadre d'une prédiction.

Par ailleurs, il est à souligner que le processus de *clustering* utilisé dans nos travaux se fait par similarité, ce qui implique que l'utilisateur n'a pas besoin de reproduire exactement une situation précédente. Ceci est particulièrement important, notamment en ce qui concerne la nature dynamique de l'information de contexte, qui peut varier entre les sollicitations. Si ces variations ne sont pas très significatives, il est toujours possible de classer la demande, même si celle-ci n'est pas exactement la même que celles observées précédemment.

Il est également à noter que ce processus de prédiction aboutit à l'identification des séquences d'intention d'un même niveau de granularité. On peut alors imaginer qu'une telle séquence pourrait en effet correspondre à une intention de plus haut niveau d'abstraction qui est décomposée dans cette séquence d'intention (suivant les opérations de composition d'intention discutés, par exemple, dans [121, 214]). L'usage d'un modèle autre que les chaînes de Markov utilisées ici²⁶ permettrait d'envisager l'identification d'autres relations permettant de composer des intentions plus complexes. On pourrait pour cela s'inspirer à nouveau des techniques utilisées dans des travaux sur la fouille de processus tels que [69,54].

²⁶ Dans ce travail, on utilise des chaînes de Markov en profondeur 1, puisqu'on se limite aux probabilités de passage d'un état n à un état $n+1$.

3.4.3 Bibliométrie

Ces travaux sur la prédiction de services ont donné lieu à 5 publications, citées ci-dessous. Ces citations²⁷ ont été organisées en deux groupes (antérieurs à 2016 et postérieurs à 2016), en fonction de leur date de publication, couvrant ainsi les premières années suivant l'apparition de ces publications et ces 4 dernières années. Le Tableau 9 regroupe ces informations, illustrées dans la Figure 49.

- **UbiMob 2012b** [183] : Najar, S. ; Kirsch-Pinheiro, M. & Souveyet, C., « Mécanisme de prédiction dans un système d'information pervasif et intentionnel », In : Philippe Roose & Nadine Rouillon-Couture (dir.), *8èmes Journées Francophones Mobilité et Ubiquité (UbiMob 2012)*, June 4-6, **2012**, Anglet, France. Cépaduès Editions, pp. 146-157. ISBN: 978.2.36493.018.6
- **ICWS 2014** [178] : Najar, S. ; Kirsch-Pinheiro, M. & Souveyet, C., "A context-aware intentional service prediction mechanism in PIS", In: David De Roure, Bhavani Thuraisingham & Jia Zhang (Eds.), *IEEE 21st International Conference on Web Services (ICWS 2014)*, 27 June - 2 July **2014**, Anchorage, Alaska, USA, IEEE CS, pp. 662-669. DOI : 10.1109/ICWS.2014.97
- **ANT 2014** [179] : Najar, S. ; Kirsch-Pinheiro, M. & Souveyet, C., "A new approach for service discovery and prediction on Pervasive Information System", *5th International Conference on Ambient Systems, Networks and Technologies (ANT-2014)*, *Procedia Computer Science*, vol. 32, **2014**, Elsevier, pp. 421–428. DOI : <http://dx.doi.org/10.1016/j.procs.2014.05.443>
- **Chapitre IGI 2013** [180] : Najar, S.; Kirsch Pinheiro, M.; Vanrompay, Y.; Steffemel, L.A. & Souveyet, C., "Intention Prediction Mechanism in an Intentional Pervasive Information System", In : Kolomvatsos, K., Anagnostopoulos, C., Hadjiefthymiades, S. (Eds.), *Intelligent Technologies and Techniques for Pervasive Computing*, IGI Global, **2013**, pp. 251-275. DOI: 10.4018/978-1-4666-4038-2.ch014, ISBN : 978-1-4666-4040-5.
- **JAIHC 2015** [177] : Najar, S.; Kirsch-Pinheiro, M. & Souveyet, C. "Service discovery and prediction on Pervasive Information System", *Journal of Ambient Intelligence and Humanized Computing*, vol. 6, issue 4, June **2015**, Springer Berlin Heidelberg, pp. 407-423. ISSN: 1868-5137. DOI: <http://dx.doi.org/10.1007/s12652-015-0288-5>

Tableau 9. Citations par publication et par intervalle.

Référence	Année	Total	< 2016	≥ 2016	self
UbiMob 2012b	2012	2	2	0	0
ICWS 2014	2014	1	0	1	0
ANT 2014 :	2014	18	2	14	2
Chapitre IGI 2013	2013	2	1	0	1
JAIHC 2015	2015	8	0	5	2
Total / %		31	16,13 %	64,52 %	19,35 %

Comme on peut s'apercevoir à travers le Tableau 9, le nombre de citations à ces articles est encore limité. Le caractère relativement récent de ces travaux explique en partie ce phénomène. Par contre, il est à noter, notamment sur la Figure 49, que la majorité des citations observées ont été réalisées à

²⁷ Comme pour les travaux précédents, l'impact de ces publications en matière de citations a été analysé à l'aide du site scholar.google.com et résumé dans le Tableau 9.

partir de 2016, suggérant ainsi une thématique encore en développement. Le récent essor des techniques de *Machine Learning* et de l'Intelligence Artificielle pourraient en effet apporter un nouveau regard à ces travaux, réalisés avant la démocratisation de ces techniques.

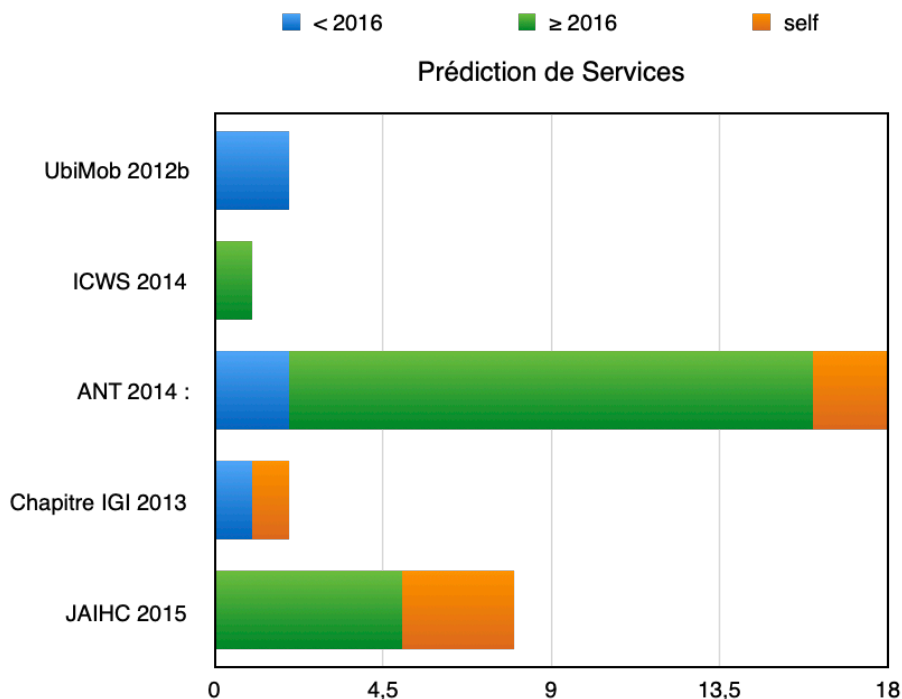


Figure 49. Citations par publication.

3.5 Bilan

Au cours des sections précédentes, plusieurs contributions dans le domaine de l'Informatique Pervasive ont été mises en avant : la fouille de contexte par ACF, la sélection de services basée sur le contexte d'utilisation et aussi la sélection de service basée sur ce contexte et l'intention de l'utilisateur, et enfin la prédiction de services. Ces travaux se sont étalés entre 2008 et 2014 (2012-2014 pour la fouille de contexte, 2008-2009 pour la sélection par graphe, 2009-2012 pour la sélection par intention et contexte et 2012-2014 pour la prédiction de services), impliquant deux thèses (Ali Jaffal 2012-2014 et Salma Najar 2009-2014), mais également un projet européen (le projet MUSIC en 2008 pour la sélection de services par graphe).

Le fil conducteur de ces travaux, au-delà de la notion de contexte, est l'usage de la notion de service : les fonctionnalités proposées par un système sont encapsulées derrière cette notion. Celle-ci offre un apport non négligeable à l'Informatique Pervasive de par l'interopérabilité qu'elle permet d'offrir, mais également par la transparence qu'elle propose, puisque différentes technologies peuvent se cacher derrière cette notion. Par ailleurs, le faible couplage obtenu à travers l'orientation service permet de mieux gérer la dynamique propre aux environnements pervasifs.

Les différentes contributions présentées ici s'illustrent notamment sur les services sensibles au contexte (*context-aware services*), principalement dans la sélection de ces services. Ces travaux se distinguent d'autres dans le domaine par la prise en compte de l'incertitude des données contextuelles et de l'intention de l'utilisateur. Les travaux sur la sélection de services apportent ici l'idée d'un contexte requis, lequel rend explicite le fait qu'une certaine implémentation a été conçue supposant un certain contexte d'exécution côté client. Le fait de poser ainsi des conditions sur le contexte utilisateur demeure minoritaire dans la littérature où le contexte d'exécution côté fournisseur reste

prioritaire (les nombreux travaux sur la gestion des SLA²⁸ et de QoS illustrent bien cet intérêt accru par une vision centrée fournisseur, s'il est ou non capable de fournir le service à souhait). Cependant la prise en compte du contexte de l'utilisateur dans le processus de sélection de service ouvre la porte à une meilleure prise en compte de cet utilisateur et permet de remettre celui-ci directement au centre du processus. On retrouve ici les bases d'une vision centrée utilisateur qui caractérise particulièrement nos travaux.

Les travaux sur la sélection de services par intention et contexte ont consolidé les bases de cette vision résolument centrée utilisateur, partant de l'hypothèse qu'un service est sollicité pour satisfaire une intention dans un contexte donné. L'intention émerge dans ce contexte qui influence également l'exécution du service choisi. Cette hypothèse reste innovante et ouvre des perspectives intéressantes pour les Systèmes d'Information, où la notion d'intention peut facilement être associée à des fonctionnalités et des services attendus ou à fort valeur ajoutée.

Par ailleurs, d'autres aspects sont aussi à souligner dans les contributions présentées ici, à commencer par l'application de l'ACF pour l'identification de situations d'usage à partir des données observées. L'usage de l'ACF est une approche innovante pour la fouille de données contextuelles, puisqu'elle permet une classification multi-classes de ces éléments. Celle-ci pourrait s'avérer particulièrement intéressante pour l'identification de situations. Une situation peut être vue comme un ensemble d'éléments de contexte caractérisant de manière significative les circonstances d'une action. Avec la notion de situation, on se retrouve dans un niveau de granularité supérieur à des simples éléments de contexte, à un niveau où on peut plus facilement attribuer une certaine sémantique du point de vue de l'utilisateur, d'où l'importance du caractère recouvrant de l'ACF. Le fait d'avoir des chevauchements possibles entre les classes offre plus de flexibilité pour une meilleure identification de ces situations. Par contre, les challenges pour un usage totalement automatisé (ou semi-automatisé, avec un minimum d'intervention humaine) de cette méthode subsistent.

Enfin, il est également à souligner que l'usage du triplet < *intention, contexte, service* >, notamment dans nos travaux sur la prédiction de services, demeure assez caractéristique à ces travaux. Il s'agit d'une approche plutôt innovante, qui permet d'envisager un fonctionnement plus proactif pour les Systèmes d'Information. Plus qu'anticiper le prochain service à offrir à un utilisateur, nos travaux sur la prédiction permettent d'anticiper sur la prochaine intention de l'utilisateur et un possible contexte d'apparition pour cette intention. On s'approche à nouveau de la notion de situation, avec un ensemble d'éléments de contexte caractérisés désormais par une intention. Par ailleurs, on retrouve encore ici la vision centrée utilisateur qui caractérise majoritairement nos travaux dans ce domaine. Cette vision est davantage développée dans le Chapitre 4, lequel se concentre sur la notion de Système d'Information Pervasif.

²⁸ *Service Level Agreement*

4 Contexte dans les Systèmes d'Information Pervasifs

Les travaux décrits dans ce chapitre s'intéressent particulièrement aux Systèmes d'Information et à leur « nouvelle génération », qu'on appelle ici *Systèmes d'Informations Pervasifs* (SIP), dans laquelle la notion de contexte est appelée à jouer un rôle central.

Pour mieux comprendre cette évolution des Systèmes d'Information (SI), il faut d'abord s'intéresser aux récentes transformations subies par les SI. Cette dernière décennie a été témoin des nombreuses évolutions technologiques et des nouveaux usages ayant fortement impacté les SI. Parmi les nouvelles tendances apparues au cours de ces dernières années, nous pouvons citer la 4G et le BYOD, l'IoT, le *Big Data*, le *Cloud Computing* et le *Fog Computing*, ou encore la démocratisation du *Machine Learning*.

Le développement des technologies mobiles, dont la 4G, a contribué à la démocratisation de l'accès Internet à un débit raisonnable n'importe où, ce qui a également contribué à l'entrée définitive dans les mœurs de la pratique BYOD (*Bring Your Own Device*). Le BYOD consiste à utiliser, dans son travail, son propre ordinateur personnel. Selon cette pratique, les employés utilisent leurs propres terminaux personnels pour travailler, navigant de manière transparente entre leurs espaces privés et leurs espaces de travail, au lieu de cumuler des terminaux multiples selon les circonstances, la localisation ou leur besoins professionnels [53]. Ce mélange entre matériel personnel et professionnel représente un changement significatif pour les départements IT (*Information Technology*) des organisations, lesquels traditionnellement gouvernent, déploient et contrôlent toutes les technologies utilisées par les employés/collaborateurs pour leurs activités professionnelles [78]. Aujourd'hui, il est devenu usuel (ou normal) d'utiliser ses propres dispositifs personnels (qui ne se limitent plus d'ailleurs aux simples ordinateurs portables) pour accéder au SI de son entreprise, où que l'on soit. L'accès « *anytime, anywhere* » à partir de n'importe quel type de terminal est devenu une réalité. Selon Andriole & Bojanova [10], l'usage de nouveaux dispositifs comme Microsoft HoloLens, Apple iWatch, et autres appareils Bluetooth, ouvrirait des nouvelles opportunités pour les entreprises au fur et à mesure que ces nouveaux dispositifs sont en train de changer la façon dont on navigue, on recherche, on achète et même la manière dont on vit. Il est donc naturel de penser que l'entrée de ces nouveaux dispositifs personnels dans les entreprises peut également changer la façon dont on travaille.

De la même manière, l'IoT ouvre également des nouvelles perspectives d'interaction avec l'environnement physique, et avec elles, des nouvelles perspectives business. Selon Sundmaeker et al. [243], on espère que les objets dont il est question dans l'IoT deviennent actifs, participant à différents aspects de la société : « *business, information and social process* » [243]. L'aspect informationnel demeure aujourd'hui peut-être celui le plus en vue au sein des organisations. A travers l'IoT, il est possible de récolter facilement (et même en continu) des informations issues de l'environnement physique, mais également d'agir sur cet environnement à travers des capteurs et d'actuateurs souvent reliés à des nano-ordinateurs connectés au réseau et doués d'un certain pouvoir de calcul. L'environnement physique peut alors devenir une partie intégrante des processus métiers et, par conséquent, du SI lui-même, comme l'atteste le récent essor de l'industrie 4.0, laquelle repose fortement sur l'IoT et les données issues de celle-ci, comme l'a pu remarquer Lu [154].

Les données récoltées à partir de l'IoT viennent étoffer une masse de données déjà très importante disponible au sein des organisations. Les plateformes de *Big Data* permettent aujourd'hui de mieux maîtriser ces données et d'enfin pouvoir les exploiter convenablement. La popularisation des *Data Lakes* [186], construits souvent sur des plateformes comme HDFS, issue du projet Apache Hadoop, est une excellente illustration de l'entrée définitive du *Big Data* dans les organisations. Cet immense volume de données est désormais à la disposition des *data scientists* qui peuvent en extraire de la valeur ajoutée, à travers de nombreuses techniques d'analyse, y compris issues du *Machine Learning*, dont la réussite dépend bien souvent de la disponibilité d'un tel volume de données, à condition, bien évidemment, d'avoir accès à une infrastructure permettant ce type d'exploitation.

Enfin, l'essor du *Cloud Computing* a permis à beaucoup d'organisations de rationaliser leur infrastructure informatique. Le *Cloud Computing* peut être vu comme la capacité d'accéder à un pool

de ressources détenu et maintenu par une tierce partie via Internet. Il ne s'agit donc pas d'une nouvelle technologie à proprement parler, mais d'une nouvelle manière de consommer les ressources informatiques [85]. Dans le modèle *cloud*, les ressources utilisées n'appartiennent plus à l'organisation, mais elles sont le plus souvent « louées » auprès d'un ou plusieurs fournisseurs en fonction des besoins de l'organisation. Les ressources sur le *cloud* sont ainsi perçues comme ayant un faible coût de maintenance, adoptant un modèle à la demande dans lequel on peut adapter sa consommation en fonction de ses besoins et ne payer que les ressources effectivement consommées. Cependant, l'adoption du modèle *cloud* est souvent accompagnée de quelques craintes liées à l'externalisation des données et de leur traitement. Ces craintes concernent notamment la sécurité, la confidentialité ou encore la latence réseau. Le choix entre déployer un certain service dans une ressource interne à l'organisation ou de l'externaliser sur une ressource *cloud* publique est devenu ainsi un choix aussi bien stratégique que technique. Les ressources sont, par conséquent, plus visibles et doivent désormais être gérées autrement que d'un point de vue exclusivement technique.

Le *Fog Computing*, que nous avons vu dans le chapitre 2, est venu renforcer cet aspect. Le *Fog Computing* peut être vu comme un nouveau paradigme permettant la distribution des calculs, du stockage et de la gestion de services au plus proche de l'utilisateur final, tout au long du continuum entre le cloud et les objets (IoT) et les terminaux [58]. Grâce aux plateformes de *Fog Computing*, il est désormais possible de considérer l'usage de ressources de proximité pour l'exécution de certains services. Il devient alors possible d'envisager l'usage des ressources autres que ceux des *data centers* ou des plateformes *cloud* pour l'exécution de services, ouvrant encore plus la perspective d'une rationalisation de l'usage des ressources disponibles.

Toutes ces nouvelles évolutions et tendances entrent petit à petit dans la composition des Systèmes d'Information conduisant à leur propre évolution. Nous sommes à l'aube d'une nouvelle génération de SI qu'on pourrait appeler SI Pervasifs tant par leur distribution au-delà des frontières de l'organisation, que par le caractère pervasif de l'environnement qu'ils intègrent. Grâce à ces nouvelles évolutions et pratiques, les SI peuvent s'étendre bien au-delà des limites physiques de l'organisation. Ils sont désormais accessibles partout, incluent des ressources internes et externes à l'organisation et peuvent même intégrer l'environnement physique. Les notions de ce qui est à l'intérieur ou à l'extérieur d'une entreprise sont devenues floues avec des processus utilisant des ressources autres que ceux du périmètre traditionnel de l'entreprise [47]. L'environnement est devenu de plus en plus hétérogène, intégrant des dispositifs de nature très variée, qui peuvent en plus être mobiles, ce qui rajoute le dynamisme à l'hétérogénéité. Nous avons donc des SI qui sont de plus en plus confrontés à un environnement hétérogène et dynamique, intégrant des ressources et des services internes et externes à l'organisation, voir même l'environnement physique entourant ces ressources mais aussi les utilisateurs ; des systèmes pour lesquels on attend davantage de souplesse et une certaine « intelligence » afin de mieux réaliser les activités de l'organisation et les besoins de ses utilisateurs.

Un SI Pervasif peut ainsi être vu comme une classe émergente de SI dans laquelle l'IT est graduellement embarquée dans l'environnement physique, capable d'accommoder les besoins et les désirs des utilisateurs quand nécessaire [146]. Le terme « *Pervasive Information Systems* » a été introduit par Joel Birnbaum en 1997 [25]. Dans cet article [25], l'auteur parle notamment de la technologie qui devient pervasive, et ainsi invisible aux yeux des utilisateurs : « *Today's schoolchildren don't think of TVs and telephones as technology—they can't imagine life without them. Tomorrow's children will feel the same way about computers, the networks connecting them, and the services they perform* »²⁹. Il s'agit bien de la même « invisibilité cognitive » dont font état Bell & Dourish [20]. Ces auteurs mentionnent une technologie invisible à nos yeux, puisqu'on l'utilise continuellement sans forcément la percevoir comme étant des ordinateurs. Birnbaum [25] parle, quant à lui, d'une technologie de l'information qui

²⁹ « Les enfants à l'école d'aujourd'hui [N.T. l'article date de 1997] ne pensent pas à la télé ou aux téléphones comme des technologies, ils ne peuvent pas imaginer la vie sans eux. Les enfants de demain se sentiront de la même manière à propos des ordinateurs, des réseaux les connectant, et des services qu'ils réalisent ».

devrait devenir intuitivement accessible à tous et qui devrait apporter suffisamment de valeur ajoutée pour justifier les investissements nécessaires. Au vu de évolutions et tendances discutées ci-dessus et des opportunités qu'elles dégagent pour les organisations, on peut dire qu'on y est arrivé. Et les conséquences pour les SI sont non-négligeables. Birnbaum [25] souligne notamment les attentes par rapport aux services offerts. Pour cet auteur, de la même manière que les personnes attendaient (en 1997) d'avoir une tonalité lorsqu'elles décrochaient le combiné d'un téléphone, les personnes vont (aujourd'hui donc) attendre qu'une information utile soit disponible et prête à l'usage. Bref, même si Birnbaum [25] ne définit pas précisément la notion de SIP tel que Kourouthanassis & Giaglis [146] le font, les éléments qu'il énumère dans son article, à savoir la technologie qui devient « invisible », l'importance des services et la plus-value de l'information, le changement de paradigme avec les personnes qui payent par l'usage, modifiant ce qui était avant un investissement de capital en service, etc., caractérisent assez bien ce en quoi se transforment les SI d'aujourd'hui.

On est, donc, face à l'émergence d'une nouvelle génération de SI qu'on appelle ici SI Pervasif et qu'on peut définir comme un SI qui s'étend au-delà des frontières physiques (et logiques) de l'organisation, qui intègre de manière plus ou moins transparente des nouvelles technologies et un environnement devenu lui-même pervasif (au sens technologiquement chargé) et dont on attend un comportement plus intelligent, aussi réactif que proactif. Pour Kourouthanassis & Giaglis [146], contrairement aux SI traditionnels, les SIP englobent un environnement plus complexe et dynamique, composé d'une multitude d'artefacts (et non plus uniquement des ordinateurs de type desktop), capables de percevoir le contexte de leurs utilisateurs et de gérer leur mobilité. On parle aussi dans la littérature de SI « mobile » [148], avec la notion de mobilité utilisée au sens large : spatiale, temporelle, mais aussi contextuelle. Krogstie *et al.* [148] parlent alors des systèmes caractérisés par leur dynamisme, par des changements de contexte fréquents (contexte spatio-temporel, environnemental, mais également relatif aux utilisateurs, à leurs tâches et même aux informations disponibles), nécessitant ainsi une importante capacité d'adaptation du système aux utilisateurs. Même si ces auteurs [148] mentionnent notamment l'adaptation des interfaces pour une meilleure interactivité avec les utilisateurs, quels que soient les terminaux utilisés, il est aisé d'imaginer que cette adaptation devrait aller jusqu'aux services proposés et à leur implémentation.

On peut distinguer cette nouvelle génération de SI (les SIP) des SI traditionnels par différentes caractéristiques, obtenues à partir de l'analyse de la littérature, qu'on pourrait même associer à des exigences qui doivent être observées lors de la conception d'un SIP [175] :

- **Hétérogénéité** : un SIP doit supporter l'hétérogénéité des dispositifs et des technologies qui composent les environnements pervasifs ;
- **Transparence** : un SIP doit être transparent, pouvoir cacher aux utilisateurs l'hétérogénéité propre aux environnements pervasifs ;
- **Sensibilité au contexte** : tout système dit pervasif doit être capable d'accomplir les fonctionnalités sollicitées, malgré des changements dans les conditions environnantes ou dans l'état du système [215]. Un SIP doit ainsi être capable de percevoir son environnement d'exécution et de s'adapter en conséquence ;
- **Satisfaction des besoins** : Un SIP doit être conçu pour répondre et satisfaire aux besoins des utilisateurs dans le cadre de leurs activités ;
- **Prédictibilité** : un SIP est censé répondre aux besoins métiers des utilisateurs de manière prédictible et contrôlée. Même s'il doit tirer profit de l'environnement dynamique et des opportunités qu'un tel environnement peut offrir, le comportement d'un SIP, avec les services et les fonctionnalités qu'il offre à ses utilisateurs, doit rester prédictible, afin d'assurer la gouvernance de ces systèmes et la confiance que les utilisateurs porteraient en ces systèmes.

On peut ainsi parler de SI caractérisés notamment par l'hétérogénéité et la dynamique des environnements et des ressources impliquées, mais également par leur besoin d'adaptation et de sensibilité au contexte. Contrairement aux SI traditionnels, dans lesquels l'utilisateur s'adapte bien souvent au système, les SIP doivent désormais prendre en compte l'environnement et le contexte

d'utilisation afin de s'adapter et de fournir aux utilisateurs le service correspondant au mieux à ses besoins et à son contexte. Il s'agit de systèmes dont l'intention serait d'augmenter la productivité de ses utilisateurs (et de ses infrastructures) par la mise à disposition de services adaptés, tenant compte de l'hétérogénéité de ces environnements devenus pervasifs. La sensibilité au contexte devient ainsi un élément clé puisqu'elle ouvre la possibilité d'adaptation à différents niveaux : modes d'interaction, services et accès aux informations, mais également infrastructure.

Toute cette évolution des SI vers des SI Pervasifs apporte avec elle des nombreux challenges pour lesquels un nouveau regard, adapté aux caractéristiques et aux contraintes de cette nouvelle génération, est nécessaire. Parmi ces challenges, deux concentrent notre attention depuis 2013-2014 : la conception et le pilotage de ces systèmes ; et la gestion de ressources.

Le premier challenge qui nous intéresse concerne donc la conception et le pilotage d'un tel système, confronté à une hétérogénéité et à un dynamisme peut-être sans précédent, mais toujours soumis aux contraintes (et aux pratiques) propres à un SI. En effet, un SIP est avant tout un SI. Son objectif principal est, en quelque sorte, de rendre accessible aux utilisateurs les fonctionnalités nécessaires au bon fonctionnement de l'organisation et de ses processus. Ces fonctionnalités peuvent prendre de nombreuses formes, en fonction des composants et des technologies disponibles, mais également en fonction du contexte dans lequel elles sont invoquées. L'hétérogénéité de cet environnement devenu pervasif induit plus de variabilité, nécessaire pour que le système puisse mieux s'adapter aussi bien au contexte d'utilisation qu'aux besoins des utilisateurs et de l'organisation. La conception et le pilotage d'un SIP deviennent ainsi un problème complexe à gérer, car il faut identifier les services qui doivent être accessibles aux utilisateurs et dans quelles circonstances, le tout en tenant compte de l'hétérogénéité de l'environnement et des services impliqués. Les SIP sont ainsi caractérisés par un besoin accru de transparence, à la fois pour les utilisateurs finaux et pour les concepteurs eux-mêmes, car il faut masquer la complexité de l'environnement pervasif sans pour autant la perdre totalement de vue (surtout pour les concepteurs). La question qui se pose alors est comment concevoir un tel système tout en maîtrisant sa complexité ? Ou, de manière plus précise, comment gérer les services qui doivent être proposés afin de satisfaire les besoins propres au système tout en garantissant la transparence nécessaire à ces nouveaux SI ?

Il ne faut pas oublier que on parle bien ici de systèmes qui existent, qui sont déjà bien implantés dans les organisations, et qui sont voués à évoluer. Il faut donc pouvoir piloter cette évolution, passer l'offre existante vers une mise à disposition plus large et mieux adaptée. Il s'agit bien de rendre les différents services offerts par le SI accessibles partout là où ils pourraient être nécessaires, au moment où ils le seraient, indépendamment des technologies impliquées.

Malheureusement, force est de constater que peu d'abstractions ou d'outils conceptuels existent aujourd'hui pour aider les concepteurs (ou les administrateurs) de tels systèmes dans ce passage d'un SI traditionnel vers un SI Pervasif. Les départements IT (nos traditionnels « Direction de Systèmes d'Information ») se retrouvent bien souvent seules face à ce challenge. La première contribution que nous présentons dans ce chapitre s'attaque donc à cette question, en proposant une abstraction pour ces systèmes. Cette abstraction, nommée « *Espace de Services* », se présente comme un outil conceptuel permettant de mieux maîtriser la complexité d'un SIP, ouvrant la voie à une meilleure gestion dans sa globalité.

Le second challenge qui nous intéresse ici concerne la gestion de ressources dans un SIP. Pendant des années, la notion de ressources (au sens ressources informatiques) permettant l'exécution de services n'a reçu que peu d'attention dans la modélisation et la conceptualisation des SI. Ceci s'explique en grande partie par le fait que les ressources disponibles dans un SI étaient majoritairement stables et homogènes. On parle notamment des « *data centers* » et de structures semblables, où tous les calculs (c.a.d. l'exécution des services proposés par le SI) ont lieu. Les ressources n'étaient ainsi pas perçues comme quelque chose de stratégique pour le SI : quel que soit le service, il serait exécuté dans ces structures avec une infrastructure plutôt stable. Or l'introduction du *Cloud Computing*, et plus

récemment du *Fog Computing*, changent la perception de ces ressources. On rajoute à cela la tendance actuelle vers un usage accru des micro-services dans les organisations, lesquels prônent un découpage plus fin des fonctionnalités permettant, par la même occasion, un déploiement plus aisé des applications. Il est désormais possible, à l'aide des micro-services, d'envisager un usage opportuniste des ressources disponibles, tel que soutenu par [168,261]. Toutes les conditions sont ainsi réunies pour qu'on puisse déployer de manière plus ou moins transparente les services offerts par un SI sur des ressources aussi variées que les ressources *cloud* (privé ou public), celles traditionnelles des *data centers*, des dispositifs réseaux, IoT, ou encore des terminaux mobiles. Toutes ces évolutions ont transformé la nature des ressources disponibles dans les SI. Ces ressources sont devenues davantage réparties, hétérogènes, disposées dans une infrastructure devenue elle-même plus dynamique. Le placement des services sur ces ressources devient ainsi un problème non-trivial.

Même si la gestion de ressources et le placement de tâches sur ces ressources soient des problèmes traités de longue date dans les systèmes répartis et la communauté HPC (*High Performance Computing*), comme le démontrent [231, 250], les caractéristiques des ressources impliquées dans un SIP rendent cette tâche encore plus complexe. Les ressources dans un SIP ressemblent, en termes d'hétérogénéité, à celles considérées par le *Fog Computing*. D'une part, elles englobent les serveurs exécutant dans les *data centers*, mais également des machines virtuelles sur de plateformes *cloud* ou encore de « *micro data centers* », des ressources type « serveur » spécialement déployées à proximité des utilisateurs pour un usage de type *Fog Computing*. D'autre part, nous avons de ressources type RaspberryPi et autres nano-ordinateurs, utilisées bien souvent pour et par l'IoT, ainsi que des ressources mobiles et/ou à usage personnel, tels que ordinateurs portables, *desktops*, tablettes et même *smartphones*. Chacune de ces ressources peut être à la fois source d'information et plateforme d'exécution pour certains services, car dotées d'une certaine capacité de calcul.

Nous sommes donc face à un pool de ressources très hétérogène, mais également dynamique puisque ces ressources peuvent aller et venir, devenir disponible ou disparaître à tout moment, au gré des circonstances (*e.g.* déplacement ou départ de son propriétaire/utilisateur, problèmes de connectivité ou d'alimentation, voir même fin de contrat). En outre, ces ressources ne sont pas forcément dédiées à l'exécution de ces services et n'appartiennent pas forcément à l'organisation (ils peuvent, par exemple, appartenir à des partenaires ou à des collaborateurs ; être utilisées pour l'exécution de services propres à l'organisation mais aussi pour des tâches privées).

Ainsi, les ressources disponibles dans un SIP peuvent être caractérisés par leur hétérogénéité et le dynamisme de l'environnement, tout comme les ressources considérées par le *Fog Computing*. Or plusieurs auteurs [94,101,106] soulignent le fait que ces caractéristiques rendent la gestion de ressources, et avec elle le placement de tâches dans ces ressources, plus difficiles, notamment lorsqu'on les compare aux plateformes de *Cloud Computing*.

Nous sommes donc intéressés à ce problème dans les SIP, et plus particulièrement à la question d'un usage opportuniste des ressources disponibles. En effet, comme tout SI, les SIP disposent d'un parc important de ressources disponibles. Ces ressources pouvant évoluer rapidement, se pose alors la question de leur usage de manière opportuniste, tant qu'elles resteraient disponibles et proposeraient des conditions favorables à l'exécution de certains services. La dernière partie de ce chapitre va donc s'intéresser à cette question d'une *gestion opportuniste de ressources dans les SIP*, avec des contributions qui viennent à la suite du projet PER-MARE et de la proposition de la plateforme CloudFit (cf. section 2.2.2.2), mais qui sont toujours en construction, à travers notamment la thèse de David Beserra, démarrée fin 2016.

4.1 Espace de services

4.1.1 Rappel de la problématique

Conformément à ce que nous avons exposé dans la section précédente, les SIP sont caractérisés par leur hétérogénéité. Cette hétérogénéité touche aussi bien les services disponibles et leurs implémentations que les ressources utilisées pour leur exécution. Les technologies impliquées sont multiples et conduisent à une complexité accrue. Cette complexité rend difficile la compréhension du système et sa prise en main, aussi bien de la part des utilisateurs finaux que de ceux devant assurer la conception et le pilotage de tels systèmes. Or comme l'a souligné Dey [75], lors que les utilisateurs expérimentent des difficultés pour établir un modèle mental de comment les applications fonctionnent, ils sont moins disposés à les adopter et à les utiliser. Le fait de ne pas comprendre un SIP et son fonctionnement peut affecter son acceptation, ou du moins celle de certains de ses composants, et ainsi compromettre son adoption et la transition d'un SI traditionnel vers un SIP. Vu le rôle stratégique des SI (et donc des SIP) au sein des organisations, en tant que support à leurs activités, les conséquences de cette non-adoption peuvent être très importantes.

Il nous paraît donc important que les utilisateurs d'un SIP puissent le comprendre sans pour autant avoir besoin de connaître ou comprendre l'ensemble de technologies impliquées. Du même pour les concepteurs, ceux-ci doivent maîtriser le SIP et ses fonctionnalités, sans forcément être encombrés par les détails relatifs aux technologies utilisées. En d'autres termes, la maîtrise de la complexité d'un SIP passe inéluctablement par la transparence. Celle-ci est nécessaire pour cacher l'hétérogénéité qui caractérise ces systèmes et qui touche aussi bien leurs ressources, infrastructures, services et usages. Cette transparence est nécessaire pour permettre aux utilisateurs de pouvoir focaliser uniquement les tâches qu'ils ont à accomplir et pas les technologies qu'ils auront derrière ces tâches, et pour que les concepteurs puissent réfléchir plus aisément aux services qui devront être offerts et dans quelles circonstances, sans avoir eux aussi à se concentrer, dans un premier moment, sur les technologies nécessaires.

Pour permettre d'atteindre ce niveau de transparence, des abstractions sont nécessaires pour permettre la représentation de ces systèmes. L'espace de services décrit dans ce qui suit se présente comme un outil conceptuel permettant une représentation abstraite d'un SIP avec les fonctionnalités qu'il est supposé remplir et les ressources considérées pour permettre leur exécution.

4.1.2 Contribution & Impact

L'espace de services se veut comme un outil conceptuel permettant une meilleure prise en main d'un SIP dans sa globalité. Il permet d'abstraire l'ensemble de fonctionnalités qui doivent être offertes et les conditions nécessaires pour cela, tout en masquant l'hétérogénéité de l'environnement. Une telle conceptualisation est en fait nécessaire afin de garder une certaine maîtrise sur le système et ses composants de manière plus transparente, sans compromettre son dynamisme.

Originellement proposé en 2013-2014 [129,175], l'espace de services se présente donc comme un ensemble de définitions formelles (cf. Tableau 10) permettant de définir conceptuellement un SIP comme un ensemble d'espaces dans lesquels des services et des ressources sont mis à disposition des utilisateurs en fonction du contexte dans lequel ils (service, ressources et utilisateurs) se trouvent.

A la base de la notion d'espace de services se trouve une vision centrée utilisateur qui a démarré en 2012 pendant la thèse de Salma Najjar [170] (discutée dans le chapitre 3) et qui continue aujourd'hui à être développée, comme l'atteste l'article [131], dans lequel la notion de ressource a été ajoutée aux définitions originelles. On retrouve alors les mêmes hypothèses sur lesquelles se base la thèse de Salma Najjar (cf. section 3.3) et le triplet *< intention, contexte, service >*. Un SIP est ainsi conceptualisé comme un système devant satisfaire les besoins, et donc les intentions, de ses utilisateurs. Ces intentions peuvent être satisfaites à travers un ensemble de services, lesquels sont conçus pour les

satisfaire dans un certain contexte (celui où émerge l'intention chez les utilisateurs). On retrouve ici les deux caractéristiques majeures de cet outil qui sont la notion de *service* et celle de *contexte*.

Tableau 10. Ensemble de définitions formelles définissant un espace de services (d'après [129, 131]).

Entités actives (services) : $\mathcal{A}^t = \{ sv_i \}$, où $sv_i = \langle I, \mathcal{F}, C\mathcal{X}\mathcal{R}, Co\mathcal{R}, \mathcal{P}\mathcal{r}, C\mathcal{X} \rangle$ (1)	Fonctionnalités et intentions : $\mathcal{F} = \{ f_j (in_j, out_j) \}$ $I = \{ \langle verb, target, param \rangle \}$ (5)
Entités passives (capteurs) : $\mathcal{P}^t = \{ cp_i \}$, où $cp_i = \{ O_{cpi}, C\mathcal{X} \}$ (2)	Observations (contexte observé) : $O_{cpi} = \{ \langle ob_j, t_j \rangle \}$, $ob_j = \langle sj, eo, v, \mathcal{M} \rangle$ (6)
Ressources : $\mathcal{R}^t = \{ r_i^t \}$, où $r_i^t = \langle Ca_i, Co_i, \mathcal{P}\mathcal{r}, C\mathcal{X} \rangle$ (3)	Contexte d'exécution : $C\mathcal{X} = \{ \langle ob_j, t_j \rangle \}$, $ob_j = \langle sj, eo, v, \mathcal{M} \rangle$ (7)
Définition d'un espace de services : $\xi^t = \{ e_i \mid e_i \in \mathcal{A}^t \vee e_i \in \mathcal{P}^t \vee e_i \in \mathcal{R}^t \}$ (4)	Capacités d'une ressource : $Ca_i = \langle \mathcal{F}, C\mathcal{X}\mathcal{E} \rangle$ (8)

La notion de service est utilisée dans l'espace de services de la manière la plus abstraite et générale possible, comme on peut voir à travers la définition (1) dans le Tableau 10. On voit les services ici à travers leurs fonctionnalités (\mathcal{F}) et les intentions (I) qu'ils permettent de satisfaire, ce qui nous permet de représenter des multiples technologies, aussi bien de services Web plus « traditionnels » (SOAP, REST) que de micro-services comme les composants OSGi³⁰ ou les containers Docker³¹. Quelle que soit la technologie impliquée, on pourra toujours la réduire aux fonctionnalités exposées à travers le service et aux intentions auxquelles le service est censé répondre, ce qui nous permet donc de cacher les détails sur les technologies utilisées derrière cette définition de service.

La deuxième caractéristique correspond à l'omniprésence de la notion de contexte. Celle-ci est associée à chaque entité composant le SIP : service (sv_i), capteur (cp_i) ou ressource (r_i^t). Chaque entité évolue dans un contexte d'exécution ($C\mathcal{X}$ dans le Tableau 10), permettant de la positionner dans un espace de service, et peut demander un contexte requis ($C\mathcal{X}\mathcal{R}$ dans le Tableau 10) pour son bon fonctionnement. Il est important d'observer que la description de contexte utilisée suit le méta-modèle présenté dans la section 3.3, proposé pendant la thèse de Salma Najar [170]. On retrouve donc (cf. définitions (6) et (7) dans le Tableau 10) un ensemble de valeurs (v) d'un élément de contexte (eo) observées pour une entité (sujet sj), associées à des métadonnées (\mathcal{M}).

Il est important d'observer que l'espace de services permet de conceptualiser les composants d'un SIP à travers les rôles qu'ils y jouent. Les composants (on entités e_i suivant la définition (4) au Tableau 10) sont vu à travers trois rôles distincts : les *services*, les *capteurs* et les *ressources*. Les services (sv_i) représentent les entités actives d'un SIP (ensemble \mathcal{A}^t dans le Tableau 10), puisqu'ils fournissent les fonctionnalités nécessaires au bon fonctionnement du système. Les capteurs (cp_i) représentent les entités passives, puisqu'ils permettent l'observation de l'environnement autour de lui. Cet environnement peut être aussi bien l'environnement physique (par exemple, l'observation de la localisation de l'utilisateur, de la température d'une pièce, etc.), mais également logique ou organisationnel (la mémoire disponible d'une ressource, l'état d'exécution d'une tâche, le rôle d'un utilisateur, etc.). Enfin, les ressources (r_i^t), qui ont été ajoutées à la notion d'espace de services en

³⁰ <https://www.osgi.org/>

³¹ <https://www.docker.com>

2019 [131], représentent toute entité capable de réaliser un service pour le compte du SI, participant ainsi au bon fonctionnement du système dans sa totalité.

Tout comme les services, les ressources (r_i) sont définies de manière abstraite, à travers les capacités (Ca_i) offertes (exprimées en termes de fonctionnalités (\mathcal{F}) ou de contexte d'exécution (CxE), suivant la définition (8) au Tableau 10), et les contraintes (Co_i) définies pour leur usage. Cette abstraction nous permet de cacher derrière cette notion aussi bien les ressources informatiques (un serveur, un ordinateur portable, une machine virtuelle...), qu'un collaborateur lequel pourrait être autorisé à réaliser certaines tâches (et donc certains services) au nom du système. En explicitant la notion de ressource dans l'espace de services, on espère pouvoir considérer le placement des services proposés par le SIP sur les ressources, et pouvoir éventuellement le faire de manière dynamique, tenant compte des ressources réellement disponibles à un instant t .

Ces abstractions correspondent ainsi aux rôles joués par les éléments composant un SI, et pas forcément aux éléments eux-mêmes. On conceptualise les éléments à travers les rôles qu'ils jouent auprès d'un SIP, ce qui permet de faire plus facilement abstraction de la vraie nature de ces éléments. Ceux-ci peuvent jouer différents rôles, ils seront alors considérés sous l'angle de leurs rôles. De ce fait, lorsqu'on parle d'un capteur, il ne s'agit pas forcément d'un élément permettant uniquement l'observation de l'environnement (e.g. un simple capteur de température intégré à l'environnement physique), il s'agit surtout d'un élément qui *peut aussi* offrir ces informations (e.g. un tel capteur sur un RaspberryPI). Il n'est pas nécessairement limité à cela, mais il peut aussi jouer ce rôle.

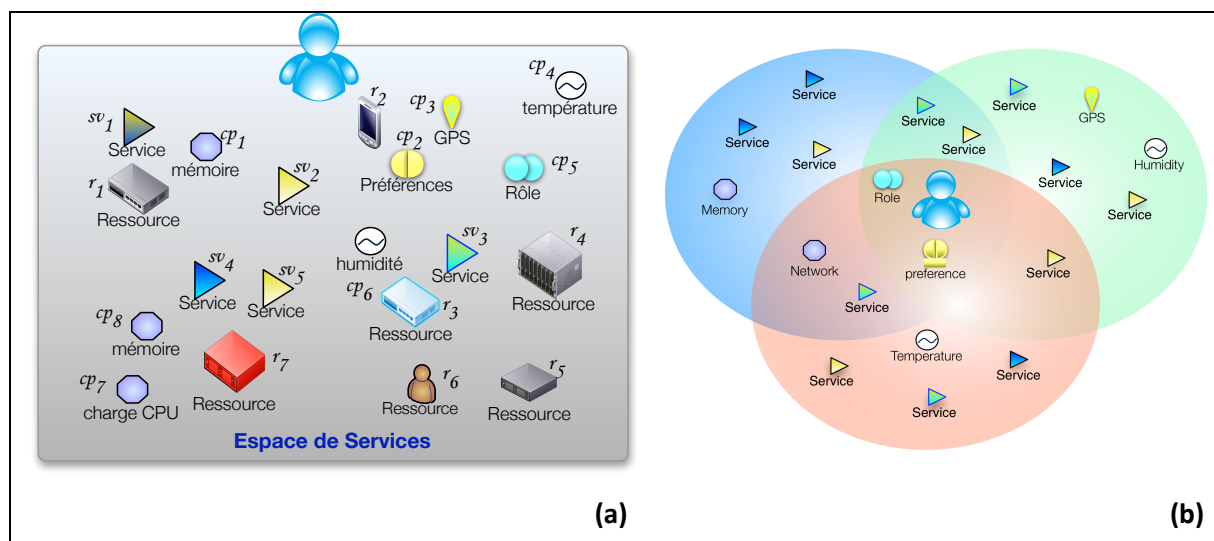


Figure 50. Illustration d'un espace de service (a) et des multiples espaces (b) composant un SIP (d'après [131, 175]).

Ces différentes définitions permettent de conceptualiser un SIP en tant qu'ensemble d'espaces de services, comme l'illustre la Figure 50. Dans chaque espace, l'utilisateur interagit avec un ensemble de services qui peuvent s'exécuter sur des ressources disponibles, en fonction du contexte courant obtenu grâce aux capteurs. Le système dans sa globalité est vu comme un ensemble d'espaces perméables et évolutifs, puisqu'ils peuvent se partager des entités communes (par exemple, un service disponible dans plusieurs espaces, ou une ressource pouvant être utilisée dans plusieurs espaces) et évoluer dans le temps (avec, par exemple, des ressources qui vont et viennent dans un espace ou qui se déplacent d'un espace vers un autre). La notion d'espace de service ne se limite donc pas à des éléments disposés dans une zone géographique précise, mais à une véritable conceptualisation au service du SI. Par exemple, si on considère un aéroport, on peut imaginer chaque hall, salle d'embarquement ou zone de l'aéroport comme un espace de services, mais également conceptualiser des espaces en fonction des secteurs d'activités de l'aéroport (accueil passager, traitement bagages,

maintenance, sécurité...) ou encore du public visé (employé, consommateurs, prestataires...). Ainsi, l'espace de services se présente avant tout comme un cadre de réflexion, permettant une meilleure prise en main des nouveaux SIP de manière transparente, sans tenir particulièrement compte des différences technologiques entre les éléments considérés. Un décideur peut ainsi conceptualiser son SIP et les services proposés, réfléchir aux ressources qui seront mis à disposition et aux informations qu'on pourra observer à partir de ce système, sans pour autant maîtriser les détails concernant tous les aspects techniques impliquant tous ces éléments.

Ce cadre de réflexion fait partie d'une vision centrée utilisateur plus vaste (cf. Figure 51). Il est accompagné par trois autres axes de développement : (i) un axe fonctionnel, composé des mécanismes de découverte et de prédiction de services présentés dans le chapitre 3 ; (ii) un axe de support avec une méthodologie guidant la définition de ces espaces ; et (iii) un axe architectural, avec une plateforme assurant l'exécution de ces espaces et de ses services. Dans la thèse de Salma Najar [170], une première méthodologie a bien été proposée (visible dans [176]), du même pour la plateforme d'exécution, le middleware IPSOM (discuté dans [175]). Néanmoins, ni l'un, ni l'autre prennent en considération la notion de ressource, introduite à *posteriori* [131]. Une nouvelle monture de la méthodologie, tenant compte de la présence de ressources, s'avère encore nécessaire. Du même pour la plateforme d'exécution, laquelle devrait assurer non seulement la découverte et la prédiction de services (comme c'est le cas d'IPSOM [170, 175]), mais également leur exécution et avec elle, le placement des services sur les ressources disponibles. Ce dernier point fait aujourd'hui partie de nos recherches en cours et sera abordé dans la section 4.2.

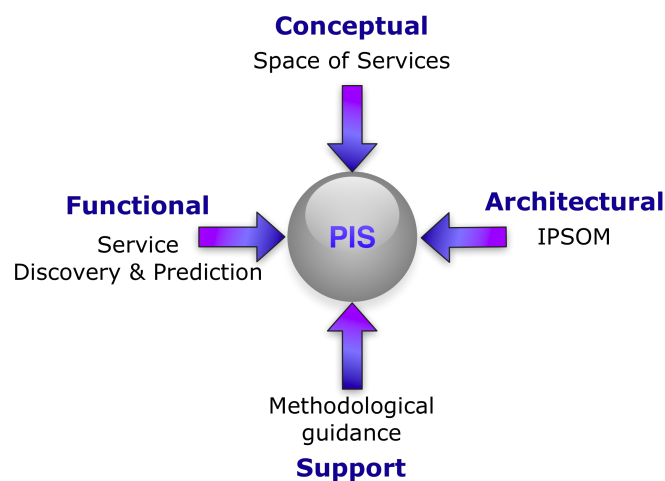


Figure 51. Axes de développement autour d'un SIP [175].

L'espace de service constitue, à ce jour, une des rares initiatives permettant la conception d'un SIP. Des initiatives existent (e.g. [147, 208]), mais souvent elles n'offrent pas une vision systémique des SIP, ou ne considèrent pas la notion de ressources. Par exemple, Rao & Angelov [208] proposent également un cadre conceptuel basé sur la notion de services. Ces auteurs centrent leur analyse sur l'offre de services à travers une vision en 4 éléments ("*infostructure*", *devices*, *interfaces*, *smart spaces*) dont l'objectif serait d'analyser la chaîne de création de la valeur ajoutée dans ces systèmes. Même si on voit apparaître dans le cadre de Rao & Angelov [208] la notion d'infrastructure de communication (à travers l'élément "*infostructure*") et celle de terminal (l'élément *device*), leur analyse est entièrement tournée vers la création de la valeur ajoutée et donc vers le consommateur et son expérience. Seuls les consommateurs (clients finaux) sont considérés et pas les autres types d'utilisateurs possibles dans un SIP (p.ex. employés, collaborateurs, partenaires). C'est donc l'infrastructure et le terminal d'accès aux services par ce consommateur qui sont considérés et pas forcément ceux nécessaires à l'exécution

d'un service. Leur caractère mobile et embarqué est pris en considération, mais pas la disponibilité des ressources, ni leurs capacités à offrir un certain service. Rao & Angelov [208], tout comme Kourouthanassis *et al.* [147], ne considèrent pas le rôle des ressources dans l'offre de services, assumant implicitement que l'exécution des services proposés est garantie, sans considérer les ressources nécessaires pour cela.

Ce travail de réflexion sur les SIP nous a également permis d'établir de nouvelles collaborations, dont notamment celle avec le professeur Massimo Villari, de l'*Università di Messina*. Cette collaboration, qui a abouti, pour l'instant, à la publication d'un article de positionnement [232], vise plus particulièrement le fossé existant entre une vision purement conceptuelle des SI et la vision technique de ceux-ci. Ce fossé est largement reconnu par la littérature [121, 214, 249], et l'évolution des SI vers des SIP n'a fait qu'aggraver ce constat. On remarque souvent dans la littérature une démarche presque exclusivement descendante, avec une vision business projetée sur le niveau technique. La démarche proposée dans [121, 214] en est d'ailleurs un exemple. Or, dans le cas des SIP, il nous paraît essentiel que cette interaction entre le niveau conceptuel et l'environnement technique se fasse dans les deux sens : en remontant des informations et des services présents dans l'environnement au niveau métier, et en projetant les éléments business de leur niveau conceptuel jusqu'à leur exécution sur l'environnement technique réel. Ceci demande une conceptualisation de l'environnement technique au niveau métier, permettant de mieux appréhender celui-ci et ses ressources.

A travers cette collaboration, une réflexion sur les stratégies permettant d'assimiler au niveau conceptuel une vision de l'environnement technique d'un Système d'Information a été menée, aboutissant à la définition du concept de « *micro-business container* ». Les *micro-business containers* (μ BC), agissent comme une abstraction au niveau conceptuel d'un élément business réalisable par le SI à son niveau technique. Ce conteneur business représente ainsi une unité métier, avec ses règles, ses services et ses informations, visibles dans la Figure 52. La mise en œuvre de ces containers est assurée grâce à la notion de microélément, originellement proposée par Carnevale *et al.* [40], laquelle intègre dans une même unité des micro-services et les données manipulées par ceux-ci. Les microéléments sont ainsi intégrés au conteneur business, et à leur tour sont associés à des micro-services du niveau technique (cf. Figure 52b). De cette manière, ces unités business peuvent être aisément projetées sur l'environnement technique, et inversement, les micro-services et les ressources présents dans cet environnement peuvent être progressivement remontés et intégrés au niveau business.

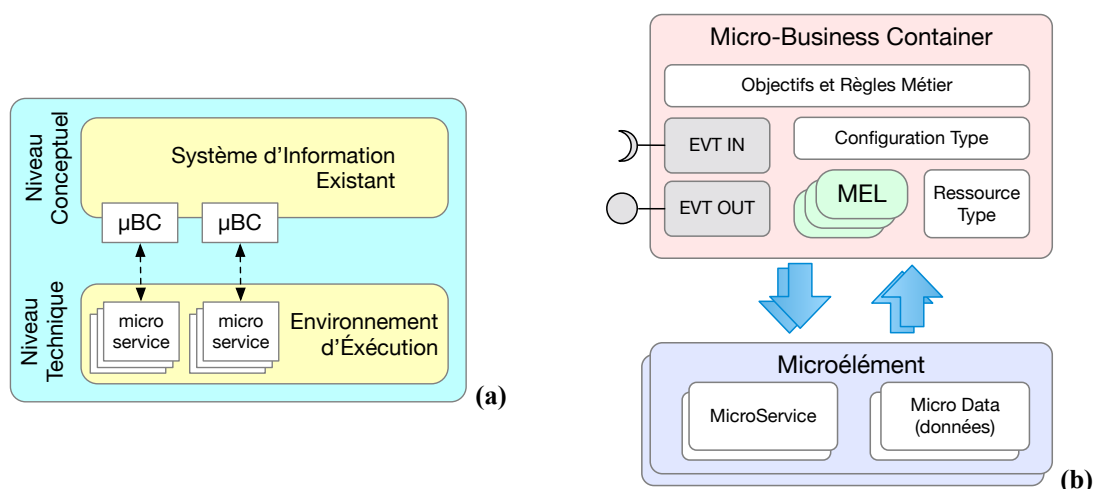


Figure 52. (a) Le micro-business containers en tant qu'élément de liaison entre le niveau technique et le niveau conceptuel, et (b) la composition d'un micro-business container et sa relation avec les microéléments [232].

Ce travail de réflexion avec les *micro-business containers* met alors l'accent sur un volet pour l'instant absent de la théorie de l'espace de services qui sont les processus. L'espace de services permet de conceptualiser les services que devront être offerts, mais pas forcément leur composition (ou décomposition) en processus métier. Or les *micro-business containers* viennent s'imbriquer à ce niveau, permettant d'exprimer au niveau conceptuel les configurations informatiques requises pour exécuter plus facilement les processus métier. Ce travail vient donc en complément à l'espace de services. Il peut s'agir d'une étape supplémentaire vers l'opérationnalisation de l'espace de services, comme l'illustre la Figure 53. Les services et capteurs (en tant que sources d'informations) présents dans les espaces de services pourraient être encapsulés au sein des *micro-business containers* et ainsi organisés dans des multiples processus métiers. L'espace de services trouverait alors un écho, à un niveau « processus », sur les *micro-business containers*, lesquels pourraient se traduire par des containers en exécution sur le plan technique. La communication entre ces différents niveaux, et notamment la remontée des informations à partir du plan technique jusqu'à un niveau conceptuel, nécessaire pour une meilleure prise de décision de la part des concepteurs, pourrait alors se faire à l'aide de la plateforme d'exécution, l'axe architectural mentionné dans la Figure 51. Tout ça reste, bien évidemment, très hypothétique à l'heure actuelle, représentant surtout un axe de travail intéressant pour des travaux à venir. Les collaborations avec l'*Università di Messina* sont invitées à continuer dans les années à venir et avec elles, espérons-le, des nouveaux résultats intéressants surtout sur le plan de l'exécutabilité de l'espace de services.

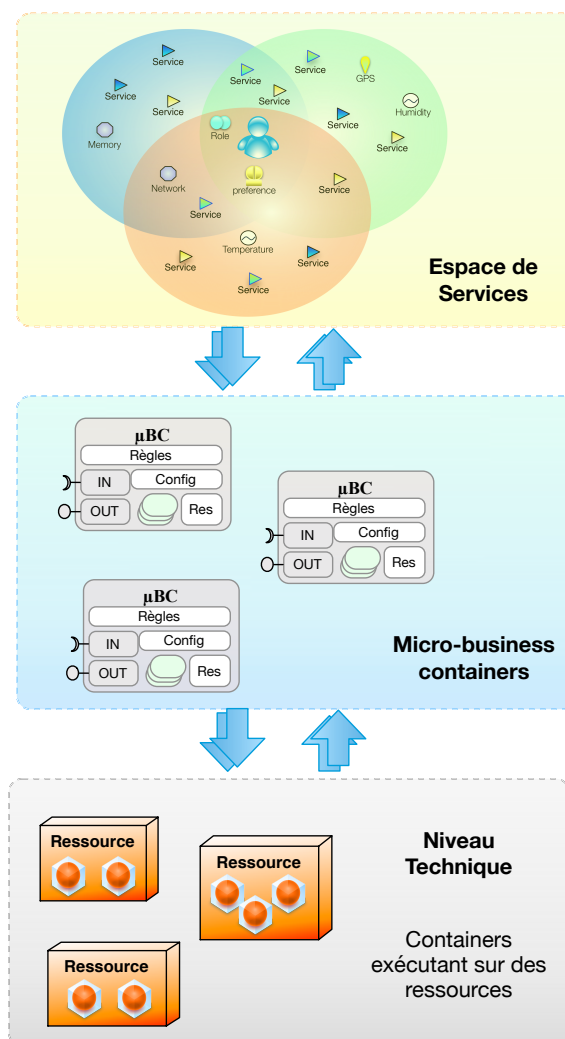


Figure 53. Correspondance entre l'espace de services, les micro-business containers et le niveau technique.

4.1.3 Bibliométrie

Comme pour les chapitres précédents, les travaux décrits dans cette section ont donné lieu à quelques publications. Celles-ci ont été analysées, à l'aide du site scholar.google.com, par rapport au nombre de citations. Le Tableau 6 détaille les données obtenues. S'agissant des publications relativement récentes et ciblant majoritairement la communauté française, le nombre de citations reste très limité. Un autre facteur qui contribue à ce chiffre est l'absence d'une terminologie qui soit largement acceptée en ce qui concerne les Systèmes d'Information Pervasifs. Ce terme n'est pas encore adopté de manière large par l'ensemble des communautés traitant les Systèmes d'Information, d'autres termes sont également utilisés. Par exemple, Hauser *et al.* [102] et Schreiber *et al.* [227], tout comme Kourouthanassis *et al.* [146, 147] et Birnbaum [25] parlent de « Système d'Information Pervasif », Bell [19] et Maass & Varshney [155] parlent de « Système d'Information Ubiquitaire », alors que Neumann *et al.* [184] utilise « *Evolutionary Business Information Systems* ». Les deux premiers termes demeurent les plus utilisés (plus de 900 références pour chacun selon le site scholar.google.com), le dernier étant largement minoritaire (à peine 34 mentions selon le même site). En termes de communauté, le premier, « Système d'Information Pervasif », semble le plus utilisé dans le domaine de l'Informatique (« *Computer Science* »), alors que les autres semblent davantage utilisés par des travaux dans le domaine des Sciences de la Gestion. Il s'agit donc d'un sujet encore récent et relativement restreint dans la communauté SI, pour lequel il n'y a pas encore un réel consensus dans la communauté. Il s'agit d'un sujet qui devra, à notre avis, encore se développer dans les prochaines années.

Tableau 11. Analyse bibliométrique des publications présentant l'espace de services.

	Année	Total	≤ 2016	> 2016	self
INFORSID 2013	2013	1			1
UbiMob 2013	2013	2	1	1	
RCIS 2014	2014	3		2	1
INFORSID 2019	2019				
Atelier 2019	2019				
Total / %		6	16,67 %	50,00 %	33,33 %

- **INFORSID 2013** [129] : Kirsch Pinheiro, M.; Le Grand, B.; Souveyet, C. & Najar, S., « Espace de Services : Vers une formalisation des Systèmes d'Information Pervasifs », *XXXIème Congrès INFORSID 2013 : Informatique des Organisations et Systèmes d'Information et de Décision, 2013*, 215-223
- **UbiMob 2013** [176] : Najar, S.; Kirsch Pinheiro, M.; Le Grand, B. & Souveyet, C., « Systèmes d'Information Pervasifs et Espaces de Services : Définition d'un cadre conceptuel ». *UbiMob 2013 : 9èmes journées francophones Mobilité et Ubiquité*, Jun **2013**, Nancy, France. Disponible sur <https://ubimob2013.sciencesconf.org/19119.html> (Dernière visite : août 2020)
- **RCIS 2014** [175] : Najar, S.; Kirsch Pinheiro, M.; Le Grand, B. & Souveyet, C., "A user-centric vision of service-oriented Pervasive Information Systems", *8th International Conference on Research Challenges in Information Science (RCIS 2014)*, IEEE, **2014**, 359-370
- **INFORSID 2019** [131] : Kirsch-Pinheiro, M. & Souveyet, C., « Le Rôle des Ressources dans l'Evolution des Systèmes d'Information », *Actes du XXXVIIème Congrès INFORSID (INFORSID 2019)*, Paris, France, Juin 11-14 **2019**, 85-97

- **Atelier 2019** [232] : Souveyet, C.; Villari, M.; Steffanel, L. A. & Kirsch-Pinheiro, M., « Une approche basée sur les MicroÉléments pour l'Évolution des Systèmes d'Information », *Atelier Évolution des SI: vers des SI Pervasifs ?*, *INFORSID 2019*, **2019**. Disponible sur https://evolution-si.sciencesconf.org/data/book_evolution_si_fr.pdf (Dernière visite : août 2020)

4.2 Gestion opportuniste de ressources dans les SIP

4.2.1 Rappel de la problématique

Conformément à ce qui a été discuté au début de ce chapitre, les SIP se caractérisent par l'hétérogénéité et le dynamisme de leur environnement. Tels les environnements considérés par le *Fog Computing*, les environnements dans les SIP contiennent des ressources très variées et variables, avec des ressources évoluant entre les serveurs haute performance et machines virtuelles sur le *cloud* aux tablettes et nano-ordinateurs pour l'IoT. Des nombreuses ressources se retrouvent déjà disponibles, intégrées dans cet environnement. Ces ressources ne sont pas toutes dans les *data centers*. Elles sont réparties au sein de l'organisation, et même au-delà, dédiées (ou pas) à différents usages. Ces ressources ne sont pas toujours dédiées exclusivement à certains services ou certaines tâches, et, même si elles ne sont pas toujours très puissantes, elles offrent malgré tout un pouvoir de calcul non négligeable. Malheureusement, en dehors des ressources sur les *data centers* et celles sur le *cloud*, des nombreuses ressources disponibles dans un SIP demeurent souvent peu utilisées, voir sous-utilisées pour certaines.

Nous avons donc des nombreuses ressources qui peuvent être disponibles de manière permanente ou temporaire. Ces ressources sont très hétérogènes, pouvant être mobiles et présenter des caractéristiques très distinctes. Leurs capacités de calcul peuvent également varier dans le temps, puisqu'il ne s'agit pas forcément de ressources dédiées à une tâche précise, pouvant donc exécuter différentes tâches simultanément, ce qui fait varier les capacités disponibles.

Vu la disponibilité de toutes ces ressources, on peut considérer dans le SIP, à l'instar du *Fog Computing*, l'usage de ressources de proximité pour l'exécution de services au nom du SIP. Or cet usage demande une gestion de ressources qui soit opportuniste, puisque guidée par la disponibilité de ces ressources et des capacités disponibles. L'objectif n'est donc pas forcément d'optimiser l'usage d'un pool fixe de ressources, comme bien souvent dans les *data centers* ou le *cloud* [231, 250], mais d'essayer d'utiliser au mieux un pool de ressources hétérogènes, dont la composition et les capacités offertes peuvent varier dans le temps et dont l'usage n'est pas exclusif aux calculs et services considérés. Telle la gestion de ressources pour les plateformes *Fog* [94, 106], la gestion de ressources dans un pool ainsi constitué représente un problème complexe, surtout lorsqu'il est comparé aux plateformes *cloud*.

Dans cette section, nous nous intéressons à cette question, avec l'objectif de permettre un usage opportuniste des ressources dans un SIP, pendant qu'elles soient disponibles et dans la mesure des possibilités offertes par ces ressources à un instant t . Selon notre point de vue, les SIP demanderaient une gestion de ressources qui soit sensible au contexte afin de tenir compte de l'hétérogénéité et du dynamisme de cet environnement. Les travaux présentés dans ce qui suit s'attaquent donc à cette question, et plus particulièrement au placement de services dans les ressources disponibles et à l'architecture nécessaire pour répondre aux besoins d'un SIP. On poursuit ainsi la vision initiée par l'Espace de Services (cf. section 4.1), en nous concentrant davantage sur ce point précis de la vue architectural (Figure 51), nécessaire à l'exécutabilité des espaces de services.

4.2.2 Contribution & Impact

Les travaux que nous avons menés autour du projet PER-MARE (cf. section 2.2), entre 2013 et 2016, nous ont ouvert des nouvelles perspectives de recherche autour de la gestion de ressources dans les environnements de *Fog Computing*. Les SIP étant caractérisés par la même hétérogénéité et le même dynamisme que ces environnements (le *Fog Computing* faisant partie de l'écosystème fomentant l'apparition des SIP), il nous paraissait logique de continuer nos recherches autour de ce sujet au-delà du projet PER-MARE (qui s'est terminé en 2014). Depuis 2016, nous avons poursuivi les travaux autour de la gestion de ressources, notamment avec la plateforme CloudFIT. Ces travaux font l'écho d'un constat, dont la littérature fait aujourd'hui état, concernant les difficultés pour le placement de tâches dans un environnement *Fog Computing* [31,79,106,152,228] pour lesquelles les SIP sont également concernés. Dès 2016, nous avons voulu explorer davantage l'impact de l'hétérogénéité dans l'exécution des tâches dans ce type d'environnement, afin justement de mieux comprendre son effet dans le cas d'un SIP, tout aussi caractérisé par cette hétérogénéité.

Afin de bien explorer cette question, il nous semblait nécessaire de d'abord mieux comprendre d'une part les effets de l'hétérogénéité sur l'exécution des tâches dans ces environnements, et d'autre part, les besoins et les caractéristiques de la gestion de ressources de manière générale et dans les SIP. Pour cela, nous avons premièrement réalisé un ensemble d'expérimentations en environnement réel (pas en simulateur) à l'aide de la plateforme CloudFIT, qui sont venues compléter les résultats du projet PER-MARE (cf. section 2.2). Puis, nous avons réalisé une étude de la littérature sur la gestion de ressources, aussi bien dans les environnements de calcul « traditionnels » (HPC et *cloud*), que dans les environnements de *Fog Computing*. Les résultats de ces deux travaux (résumés dans ce qui suit) ont conduit à la définition d'une architecture conceptuelle pour la gestion de ressources dans un SIP dans le cadre de la thèse de David Beserra (thèse toujours en cours).

Notre premier objectif était ainsi de réaliser plusieurs benchmarks expérimentaux nous permettant de démontrer qu'il était possible d'obtenir des niveaux de performance satisfaisants avec des ressources de proximité. Ceci représente la condition *sine quo non* pour l'usage de ces ressources dans un SIP. En effet, un usage opportuniste des ressources disponibles ne pourrait être envisagé que si cet usage apporte de bénéfices pour l'organisation sans trop dégrader les performances d'exécution.

Une catégorie de ressources nous a particulièrement intéressée, les nano-ordinateurs construits à partir de ce qu'on appelle « systèmes sur puce » (*SoC - System on a chip*). Ces systèmes représentent une rupture avec l'architecture d'une machine traditionnelle car ils encapsulent la CPU, la GPU, la mémoire vive ainsi que autres composants sur une même puce [267]. Ils sont généralement utilisés pour réduire le coût des ordinateurs à carte unique, tels que RaspberryPi, Odroid ou BananaPi. Ceux-ci sont utilisés pour une large gamme d'applications, de l'enseignement de l'informatique [5] à l'Internet des Objets [165]. C'est notamment ce dernier usage qui nous a attiré l'attention. Les coûts particulièrement bas de ces ressources les rend très attractives pour l'implantation d'applications de type IoT dans une organisation, et donc, dans un SIP. Par ailleurs, les nano-ordinateurs de type RaspberryPi illustrent bien les limitations propres aux ressources en « bordure » du réseau. Ces ressources sont, selon Hong & Varghese [106], bien souvent : (i) limitées, *i.e.* disposant des capacités de calcul limitées, avec notamment des processeurs moins puissants ; (ii) hétérogènes, avec des processeurs de différentes architectures ; et (iii) dynamiques, puisque leur charge de travail peut changer, les applications se plaçant en compétition pour ces ressources.

Les résultats que nous avons eu dans le projet PER-MARE (discutés dans la section 2.2) nous ont indiqué qu'il était possible (et même intéressant) de proposer un placement de tâches (qui représentent, à nos yeux, les services dans un SIP) selon les capacités disponibles dans l'environnement. Nous avons donc poursuivi les expériences en nous concentrant sur des ressources dont la puissance serait semblable à des RaspberryPi. Nous avons concentré notre analyse sur deux éléments qui nous semblaient le plus significatifs pour le placement de tâches en fonction du contexte : le contexte d'exécution observée sur la ressource (notamment la charge de la CPU), et la présence sur

la ressource (ou à proximité) des données nécessaires à l'exécution de la tâche. Même si ces deux critères paraissent particulièrement simples, ils permettent de considérer, d'une part un élément de contexte couramment observé par les mécanismes de placement de tâches, et d'autre part, un contexte requis par le service, représenté ici par la présence de données à proximité.

La présence de données à proximité évoque la notion de « *data locality* », tenue comme particulièrement pertinente dans l'usage de ressources de proximité. La localisation des données peut avoir un impact potentiellement important, car non seulement les communications et les transferts de données peuvent s'éteindre du réseau local aux plateformes *cloud*, mais également parce que les transferts de données peuvent dépendre des capacités de chaque nœud (ressource) à traiter certains volumes de données, ce qui nous amène à la question de l'hétérogénéité de ces ressources. D'un côté, le maintien des données en local présenterait non seulement l'avantage de la réduction du trafic d'accès, mais aussi de réduire la dépendance par rapport au *cloud* [110]. De l'autre côté, la capacité de ces ressources à gérer des données va être tout aussi significative que les caractéristiques liées au réseau (latence, débit, etc.), qui, à leur tour, vont aussi avoir un impact. Nous avons d'ailleurs pu constater l'impact de la gestion de données dans les travaux à l'origine de la plateforme CloudFIT dans le projet PER-MARE (cf. section 2.2.2.2). Nous avons donc voulu approfondir nos études à ce sujet.

Dans un premier moment, nous avons donc voulu étudier, à travers des expérimentations, le comportement des ressources de proximité peu puissantes, semblables aux RaspberryPi, sur différents scénarios de lecture et d'écriture des données. Ces expérimentations avaient pour but de mieux identifier la place de ces ressources dans un SIP, à mesure qu'on souhaite les utiliser de manière opportuniste pour l'exécution de certains services. Très souvent, ces ressources dites de proximité vont participer à la production des données. Les nano-ordinateurs comme les RaspberryPi sont souvent intégrés dans un scénario de IoT, se comportant comme une source de données. Pouvoir placer des calculs à proximité des données qu'ils manipulent pourrait permettre en théorie la réduction de la latence et des coûts de transfert vers des structures distantes (*cloud* ou *data center*), mais à condition que la faible puissance de ces dispositifs, notamment en ce qui concerne l'I/O, n'impactent pas (trop) négativement les performances.

Nous avons conduit des multiples expérimentations dans lesquelles on compare les temps obtenus pour la lecture et l'écriture des données sur un RaspberryPi 3³², utilisant différentes configurations (lecture/écriture en mémoire, en carte SD et en disque externe USB), et les temps obtenus en utilisant une plateforme distante³³. Nous avons également considéré l'usage d'un serveur de proximité³⁴, situé sur le même réseau local que le RaspberryPi. Trois scénarios ont été envisagés : (i) opérations sur la même machine ; (ii) sur le même réseau ; et (iii) sur des nœuds situés dans le *cloud*. Le premier cas représente les situations dans lesquelles la localité de données est utilisée pour stocker des données directement à la machine qui les traitera (le RaspberryPi en occurrence). Le deuxième scénario représente le cas intermédiaire où les données sont envoyées vers un équipement à proximité qui les traitera (le serveur de proximité ici). Enfin, le dernier scénario représente les situations où les données sont envoyées vers un nœud distant (représenté ici par la machine virtuelle sur le *cloud*).

La Figure 54 résume les résultats obtenus en termes de performance de lecture et d'écriture à partir d'un RaspberryPi 3. Nous pouvons observer que pour la lecture des données (Figure 54a), les scénarios « même réseau » (LAN) et local présentent des performances similaires pour la lecture de messages de moins de 1 Mo. Cependant, pour les messages plus volumineux, le temps de lecture entre deux nœuds devient beaucoup plus important avec le scénario LAN, passant à des niveaux presque

³² Équipé d'un processeur ARM Cortex-A53 à 4 cœurs, 1GB RAM, et connexion 100Mbps Ethernet.

³³ Machine virtuelle du type *n1-standard-2* (2vCPU, 7,5 Go de RAM) située sur la plateforme Google Cloud (GCP) aux USA.

³⁴ Un serveur AMD Opteron 6164HE à 12 cœurs, 48 Go de RAM, 1Gbps Ethernet.

comparables à ceux d'accès aux nœuds distants, qui pourtant sont les plus pénalisés par la latence des communications. Ce ralentissement met en évidence non seulement les limites du réseau (réseau à 100 Mbits/s dans notre cas), mais également le temps de traitement de la demande que, dans le cas de la plateforme CloudFIT, se fait grâce à une DHT (*Distributed Hash Table*). Pour ce qui est de l'écriture des données sur la DHT, la Figure 54b démontre, sans surprise, que l'option de stockage en mémoire locale est beaucoup plus rapide, car elle ne paie pas la surcharge d'accès au stockage persistant. Nous retrouvons des comportements presque similaires à ceux observés en lecture, notamment dans le cas du scénario « même réseau », dans lequel le temps nécessaire pour le stockage de petits blocs de données (jusqu'à 1MB) affiche des performances similaires au stockage en local, alors que pour des données de plus grande taille le coût tend vers celui d'un stockage sur le *cloud*. Autrement dit, dès que le volume des données devient plus important, les coûts (en termes de performance) pour envoyer les données à une autre machine distante ou à un nœud plus puissant à proximité deviennent semblables.

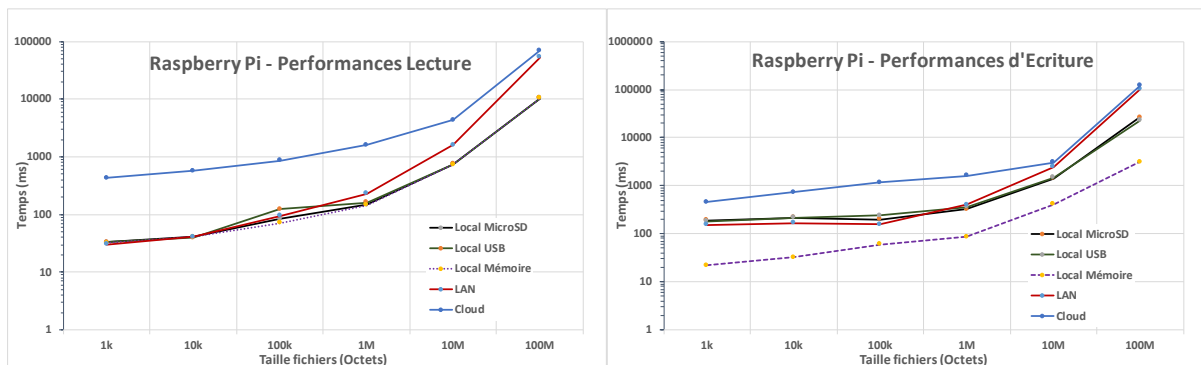


Figure 54. Résultats sur les temps de lecture (a) et d'écriture (b) sur un RaspberryPi 3 [238].

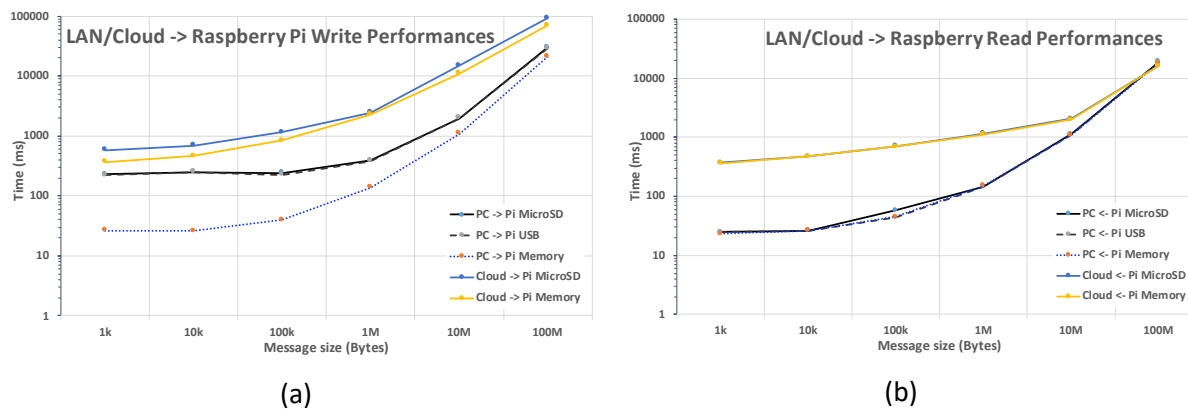


Figure 55. Résultats obtenus en transférant des données distantes vers un équipement de proximité.

Nous avons ensuite étudié le cas du RaspberryPi en tant que simple « réceptacle » ou « passerelle », recevant des données d'une autre machine. Dans ce scénario, dont les résultats sont illustrés dans la Figure 55, une machine distante va lire et écrire des données sur le RaspberryPi. On observe alors un comportement semblable au scénario précédent, même si l'ordre des valeurs impliquées n'est pas tout à fait le même³⁵, attestant d'une certaine asymétrie dans les performances (c.a.d. les performances ne sont pas tout à fait les mêmes en fonction du sens de communication). Cette asymétrie est due notamment au temps de traitement nécessaire à la plateforme CloudFIT pour la manipulation de la DHT.

³⁵ Il convient d'observer que les graphiques sur la Figure 54 et la Figure 55 utilisent une échelle logarithmique.

Tous ces résultats confortent l'idée selon laquelle ces périphériques bas de gamme sont mieux adaptés pour collecter lentement des données à partir de périphériques proches (notamment des capteurs), effectuer de petits calculs et transmettre les données à d'autres nœuds, au lieu de recevoir de gros volumes de données provenant d'ailleurs. Ils confortent également l'intérêt d'observer la présence des données à proximité lors du placement de tâches, puisque les temps de transfert de et vers le dispositif peuvent s'avérer importants en fonction du volume de données concernées.

L'autre critère que nous avons choisi d'explorer dans nos expérimentations a été la charge de la CPU. Celui-ci représente un élément de contexte particulièrement significatif du contexte d'exécution dans le cas de ressources de proximité, comme les RaspberryPi. Ces ressources pouvant être peu puissantes, une surcharge de la CPU peut être facilement atteinte et peut conduire, dans des cas extrêmes, à une paralysie de la ressource. Cette situation a pu être observée, par exemple, lors de nos précédentes expérimentations avec le projet PER-MARE (cf. Figure 32 au chapitre 2), dans lesquelles la surcharge d'un RaspberryPi a conduit à une importante perte de messages de contrôle, et avec elle, l'exécution de tâches déjà finalisées par les autres nœuds. Or l'objectif d'un usage opportuniste n'est nullement de surcharger les ressources disponibles, mais simplement de pouvoir les utiliser lorsque possible.

Nous avons donc procédé à une nouvelle série d'expérimentations autour de ce critère, à l'aide de la plateforme CloudFIT. Celle-ci utilise un mécanisme naïf pour le placement des tâches, basé sur une politique de « *best effort* », dans laquelle chaque instance exécutant CloudFIT va décider de manière indépendante si elle exécute ou non une tâche en fonction de son propre contexte d'exécution et du contexte requis par la tâche. Même si le moniteur de contexte utilisé par la plateforme continue d'observer différents éléments de contexte, nous avons défini pour cette expérimentation un contexte requis basé uniquement sur la charge moyenne de la CPU, qui ne doit pas dépasser son niveau maximale (*i.e.* une moyenne d'usage à 100%). De même, au lieu d'utiliser directement les RaspberryPi pour cette expérimentation, nous avons préféré utiliser des machines virtuelles aux configurations semblables (1 processeur, 3,75Go de mémoire vive) sur la plateforme Google Cloud afin d'éviter que des facteurs extérieurs (*e.g.* température, alimentation, réseau Wifi, etc.) puissent interférer avec l'exécution. Nous avons ainsi exécuté sur ces machines virtuelles un benchmark particulièrement gourmand en CPU (une simulation Monte Carlo, composée de 50 tâches) et observé la distribution de tâches sur les machines selon différentes configurations dans lesquelles certaines machines sont préalablement chargées à l'aide de l'outil *stress-ng*³⁶.

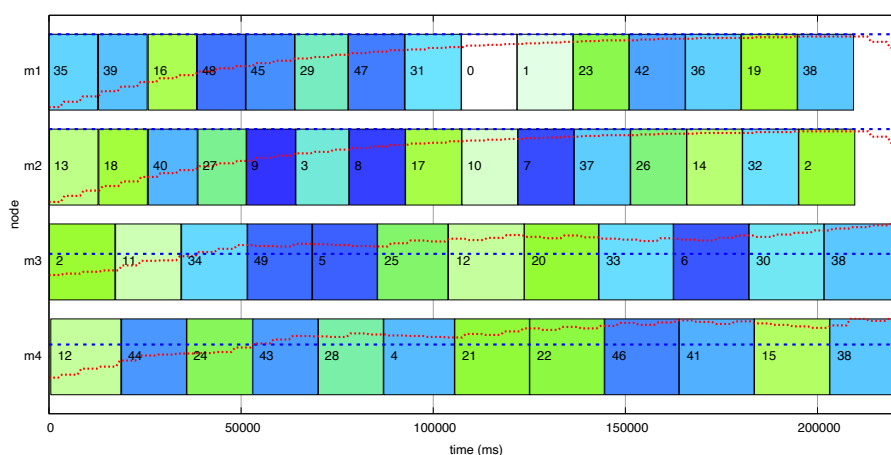


Figure 56. Exécution d'une simulation Monte Carlo sur la plateforme CloudFIT avec surcharge de ressources et sans observation du contexte pour le placement des tâches.

³⁶ <https://kernel.ubuntu.com/~cking/stress-ng/>

Dans le premier scénario, deux machines ($m1$ et $m2$ dans la Figure 56) représentent des ressources de proximité dédiées au calcul, alors que deux autres machines ($m3$ et $m4$ dans la Figure 56) sont supposées subir un usage plus opportuniste. Ces machines sont surchargées à plus de 50% de leur CPU dans Figure 56, à l'aide de l'outil *stress-ng*, mais l'usage de l'information de contexte pour le placement des tâches a été neutralisée. Il en résulte que les machines voient leur charge moyenne, observée par le moniteur de contexte et matérialisée dans la Figure 56 par la ligne rouge, dépasser largement la limite maximale (représentée par la ligne pointillée en noir). L'exécution des tâches dans ces machines est, bien évidemment, ralentie, ce qui est visible par la largeur des rectangles illustrant les tâches en exécution. On peut également s'attendre à d'autres effets découlant de cette surcharge, comme une augmentation de la température et une surconsommation d'énergie, sans parler d'un possible impact sur les autres services exécutant sur la même ressource, voire même la perte de messages que nous avons constaté dans les expériences précédentes (cf. section 2.2.2.2).

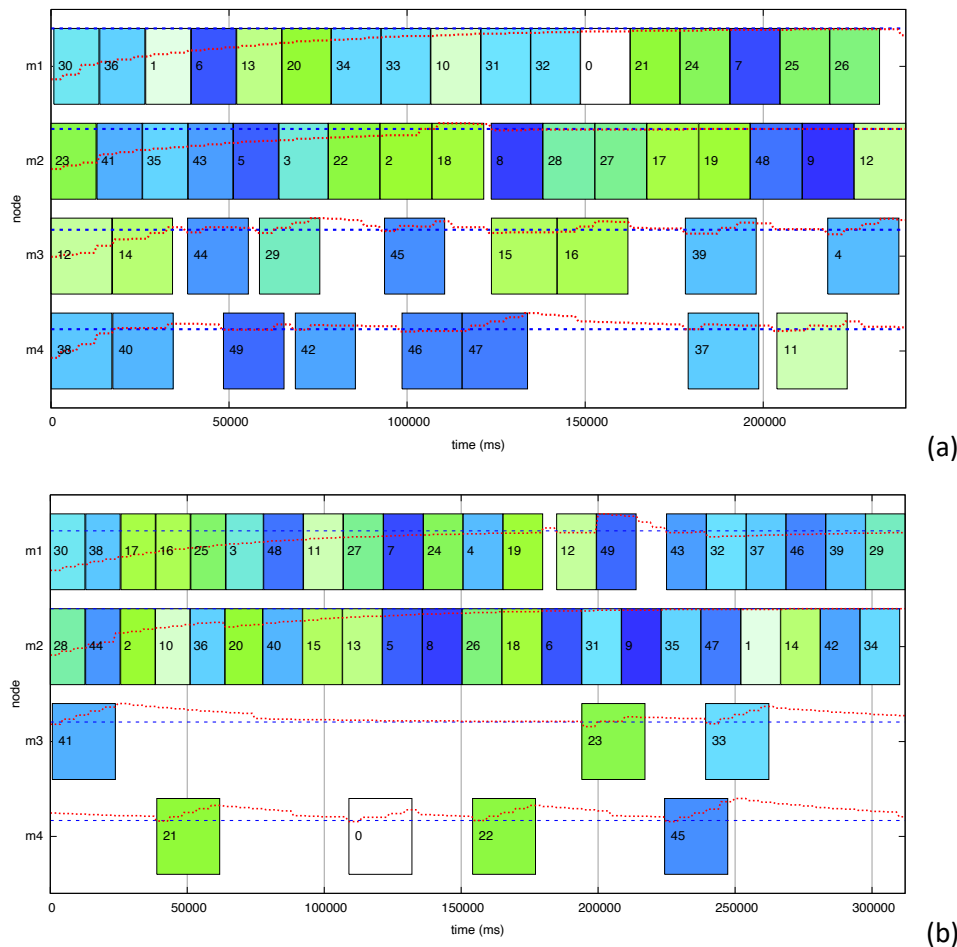


Figure 57. Exécution d'une simulation Monte Carlo dans la plateforme CloudFIT avec surcharge des machines $m3$ et $m4$, et contexte d'exécution pris en considération pour le placement des tâches.

Dans une seconde expérimentation, nous avons rétabli l'usage de l'information de contexte pour le placement des tâches (Figure 57a), puis intensifié la charge à 90% sur les machines $m3$ et $m4$ (Figure 57b). On peut alors observer que les machines concernées arrivent à mieux respecter la limite établie (qui on peut assimiler à une contrainte de la ressource, suivant les définitions de l'Espace de Services vu dans la section 4.1). Elles refusent l'exécution de certaines tâches (illustré par les espaces blancs entre les tâches dans la Figure 57) afin de mieux respecter la limitation posée sur leur contexte d'exécution. On arrive aussi à mieux préserver la CPU de l'ensemble des machines, pour un temps

d'exécution global légèrement supérieur (on passe de 222,4s à 239,2s dans la Figure 57a, avec une charge à 50%). On peut également s'attendre à d'autres bénéfices comme une consommation énergétique moins importante que le cas précédent (Figure 56) et un partage plus équitable des ressources.

Un autre aspect nous a semblé pertinent à analyser était le partitionnement du réseau, dans lequel une ressource se retrouve isolée des autres. Un tel partitionnement peut arriver, par exemple, suite à la mobilité d'une ressource, qui peut se déplacer, perdant ainsi contact avec les autres. La plateforme CloudFIT utilisant une politique de placement totalement répartie, elle devient particulièrement sensible à cette question, comme l'illustre la Figure 58. On observe, dans ce cas, la machine se retrouvant isolée, sans connaissance du travail effectué par les autres (*m2* dans Figure 58), continue l'exécution des tâches, tout en respectant la limite qu'on lui a imposée, afin de garantir la réalisation des services qui lui ont été confiés.

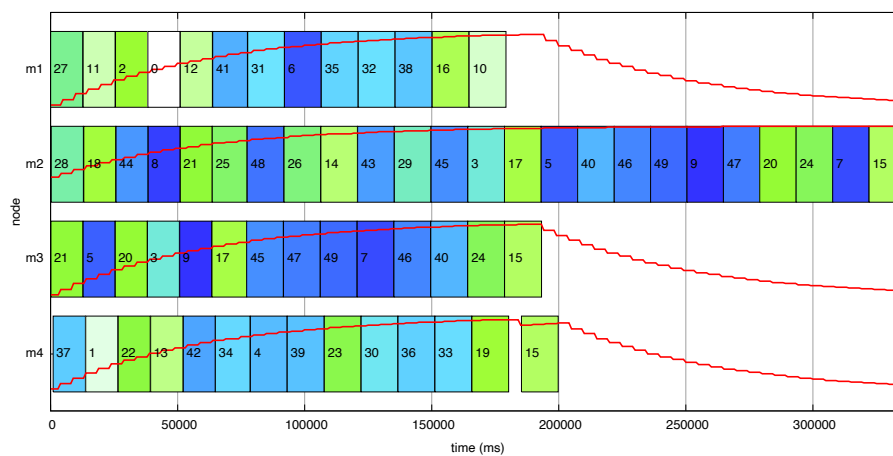


Figure 58. Exécution simulant un partitionnement du réseau, avec la machine *m2* perdant tout contact avec les autres.

Plusieurs enseignements peuvent alors être tirés (ou confirmés) de ces expérimentations. Tout d'abord, il est possible d'obtenir, à l'aide de l'information de contexte, un usage plus rationnel des ressources disponibles, qui éviterait la surconsommation des ressources à faible puissance, tout en tirant quand-même profit de leur pouvoir de calcul. Les expérimentations ont également démontré que le placement des services consommant et/ou produisant des données doit être influencé par le placement de ces données et par le volume des données impliquées, ce qui dépend fortement de l'application. Différentes stratégies pour le placement de services sont ainsi possibles et devraient coexister dans un SIP. Par ailleurs, l'organisation hiérarchique des tâches sur CloudFIT, organisées en *jobs* et *tasks*, s'avère intéressante puisqu'elle permet de gérer l'exécution de services atomiques, considérés comme une simple unité (un *job* composé d'une *task*), ou des compositions (un *job* composé de plusieurs *tasks*), supportant ainsi non seulement la composition de services, mais également un certain parallélisme dans son exécution. Enfin, une architecture totalement répartie, sans aucun élément central, comme celle de CloudFIT, s'est montrée particulièrement intéressante pour gérer le dynamisme de l'environnement et le passage à l'échelle, nécessaire pour couvrir un SIP. Par contre, ce type d'architecture a un prix : elle est certes moins sensible aux pannes et assure un meilleur passage à l'échelle que les architectures dépendant d'un élément central (un serveur, un *broker* ou un *proxy*), comme l'ont souligné Ghobaei-Arani *et al.* [94], mais elle reste sensible au partitionnement du réseau. Dans ces cas, les tâches sont malgré tout réalisées, mais le risque encouru est un certain gaspillage de ressources.

Tout ceci démontre la complexité qui peut entourer une gestion opportuniste de ressources dans un SIP. Si on croit les résultats de nos expérimentations, cette gestion est influencée par des multiples

facteurs impliquant les ressources et leur contexte d'exécution, mais également les applications (services). Des multiples stratégies demeurent possibles, ce qui nous amène à penser qu'un système de politique configurable, tenant compte de différents facteurs, s'avère nécessaire. Ces facteurs peuvent d'ailleurs dépasser le cadre purement technique. Par exemple, même si l'usage de serveurs *cloud* pour le stockage de grands volumes de données peut s'avérer techniquement intéressant, au vu des résultats présentés dans la Figure 54, cet usage peut être déconseillé en fonction du coût lié au propre *cloud* ou en fonction de la confidentialité des données. Inversement, même si l'usage de ressources disponibles localement pour l'exécution d'un service est possible, celle-ci peut s'avérer interdite en fonction, par exemple, de la propriété de la ressource (e.g. ressource appartenant à une autre organisation ou d'usage strictement personnelle). La possibilité de définir des politiques de gestion de ressources de manière flexible et configurable se présente ainsi comme une nécessité pour les SIP. Or de telles politiques ne constituent pas forcément la règle dans la gestion de ressources « traditionnelle » (HPC, *cloud*).

Tableau 12. Mots-clés utilisés pour l'étude de littérature autour de la gestion de ressources.

<i>Resource management</i>	<i>Scheduling</i>	<i>Context-aware resource management</i>
<i>Pervasive System</i>	<i>Information System</i>	<i>Context-aware scheduler</i>
<i>Fog Computing</i>	<i>Edge Computing</i>	<i>Pervasive Information System</i>

Nous avons donc mené une étude de la littérature autour de la gestion de ressources, aussi bien dans la communauté HPC et *Cloud Computing*, que dans le domaine du *Fog Computing*, afin d'en extraire les caractéristiques principales. Cette étude a été menée en 2019, dans le cadre de la thèse de David Beserra, dont le sujet est justement la gestion de ressources dans les SIP. Cette étude s'est concentrée, dans un premier moment, sur un ensemble de 10 articles, majoritairement des *surveys* et des articles récents (après 2018), choisis à partir d'un panel de 85 articles sélectionnés sur *Google Scholar* à partir de la combinaison de mots-clés présentées dans le Tableau 12. Cette revue de littérature, même si elle doit encore être approfondie, a permis à David Beserra d'établir un cadre de réflexion identifiant plusieurs dimensions d'analyse pour la gestion de ressources. Ces dimensions, représentées dans la Figure 59, organisent les nombreuses questions impliquant la gestion de ressources qui ont été identifiées lors de cette étude, comme par exemple, les caractéristiques pouvant qualifier les tâches à exécuter, les ressources impliquées et même les aspects architecturaux pour le gestionnaire de ressources lui-même. La définition de politiques de gestion, ainsi que les informations (dimension *knowledge*) nécessaires à ces politiques ont également été considérées.

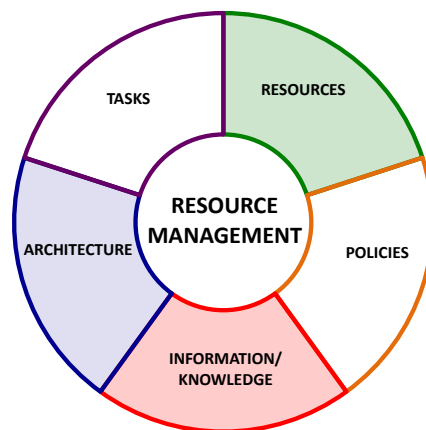


Figure 59. Dimensions de la gestion de ressources identifiées à partir de l'étude.

On remarque que la gestion de ressources va bien au-delà de l'allocation de tâches, même si celle-ci demeure un élément clé, puisqu'il s'agit bien d'affecter des tâches qui doivent être exécutées à des ressources disponibles en fonction d'un ensemble de critères et d'objectifs établis. Des nombreux travaux comme [26,31,43,106,94,250] se sont intéressés à l'allocation de tâches, aux algorithmes utilisés, aux objectifs poursuivis et aux aspects architecturaux. On y observe que la question de l'allocation de tâches est le plus souvent vue comme un problème d'optimisation dans lequel on cherche à optimiser des critères précis (usage des ressources, temps d'exécution, etc.) à l'aide bien souvent d'heuristiques plus ou moins complexes selon les cas [231, 60, 98].

Différents critères apparaissent dans la littérature représentant autant d'informations nécessaires à la gestion de ressources pour la prise de décision. On voit apparaître dans le *Cloud* [26, 231] ou le *Fog Computing* [94, 106] des critères comme la consommation d'énergie, le temps de vie d'une batterie, ou encore les coûts financiers (« *monetary cost* » [26]) et la localisation (« *target location* » [26]), ce qui représente un certain élargissement par rapport à ce qui était traditionnellement pris en considération (par exemple, le temps de réponse moyen des tâches, taux d'occupation d'un *cluster* [242], la capacité du processeur ou la latence réseau [14]). Néanmoins, les critères restent souvent préétablis, observés quelle que soit l'application, correspondant bien souvent aux critères d'optimisation recherchés.

Des travaux comme [250] se sont intéressés aux caractéristiques des tâches et du pool de ressources. On observe que les tâches ont souvent des contraintes, mais également des données (produites ou consommées) qui leur sont associées. Ces contraintes peuvent concerner des informations qu'on pourrait associer à des éléments de contexte (p.ex. nombre de CPUs, bande passante, latence réseau, etc. [94, 55, 151]). Cependant, le traitement de ces informations, vues comme des métriques, se fait bien souvent de manière *ad-hoc*. On considère chaque information de manière isolée, dans son acquisition et son traitement, sans réellement tenir compte d'éléments comme le caractère incertain et les erreurs d'acquisition pourtant assez courants dans l'observation d'une information de contexte. L'information collectée est toujours considérée comme fiable, des éléments d'interprétation qu'on pourrait associer à la notion de QoC (*Quality of Context*) sont le plus souvent absents. Même dans les travaux qui assument l'usage d'une information de contexte [79, 149], on observe souvent un traitement *ad-hoc*, focalisant uniquement certains éléments de contexte bien précis, sans forcément tenir compte des particularités de ce type d'information.

En outre, il est intéressant d'observer ici une certaine distinction entre la communauté SOA (*Service Oriented Approach*) et la communauté HPC (*High Performance Computing*). Dans la première, le plus souvent on ne considère pas qu'un service puisse avoir de contraintes pour son exécution, ces sont souvent les clients qui eux vont avoir des conditions pour l'invocation du service, alors que dans la deuxième, il est normal qu'une tâche (élément exécutable) puisse présenter des conditions ou des contraintes pour son exécution. Cette idée d'une condition pour l'exécution d'une tâche s'approche de notre notion de « contexte requis » qu'on associe à un service (cf. section 4.1 ou encore chapitre 3). On s'éloigne ainsi un peu de la vision dominante dans la communauté SOA pour s'approcher davantage de celle de la communauté HPC.

Concernant les ressources elles-mêmes, elles sont considérées sous deux niveaux : celui de la ressource individuelle et celui de l'ensemble de ressources dans un pool. Il y a une forte distinction en ce qui concerne la composition et la stabilité du pool de ressources entre les travaux en *Cloud Computing* et HPC [26,231,43,250] et ceux en *Fog Computing* [31,79,94,106,228]. Dans les premiers, les pools considérés sont bien plus stables et de nature plutôt homogène (c.a.d. peu de variation entre les ressources et des ressources qui ne changent pas ou peu dans le temps), alors que dans les deuxièmes, le pool se présente comme plus dynamique, notamment avec la présence de ressources mobiles, et plus hétérogène (voir, par exemple, [1, 94, 251]).

Concernant maintenant plus particulièrement les travaux dans le *Fog Computing*, plusieurs approches proposant le déploiement de services et d'applications ont pu être observés, dont certaines basées sur

la virtualisation [222], sur les micro-clouds [81], sur les micro-services [261], ou encore sur les workflow [101]. On observe également des travaux se présentant comme sensibles au contexte, dont [79,31]. Dans [79], un *broker* collecte les informations de contexte issues des ressources et estime leur stabilité et leur fiabilité, afin d'y placer des tâches en conséquence. L'objectif est clairement d'assurer l'exécution des tâches en évitant les ressources moins fiables. De manière semblable, Breitbach *et al.* [31] considèrent la stabilité des ressources dans l'environnement, mais cette fois pour le placement des données. Ils proposent ainsi d'estimer le nombre de réplicas pour chaque donnée et d'ensuite affecter l'exécution des tâches en fonction de la présence de ces réplicas et d'un indicateur de performance. A nouveau, on peut observer la présence d'un *broker* qui décide le placement des tâches et des données, ce qui risque d'affecter les possibilités de passage à l'échelle d'un tel système.

On observe également que plusieurs travaux, même dans le cas du *Fog Computing*, assument la disponibilité au préalable des certaines informations, comme le nombre et les caractéristiques des ressources disponibles, l'ensemble de tâches à exécuter, ou encore le profilage de ces tâches [149,251]. Or comme l'a souligné [94], il n'est pas toujours possible de disposer de toutes les informations nécessaires avant le *scheduling* dans des environnements aussi dynamiques. Selon ces auteurs, les tâches dans ces environnements ont un « temps d'arrivage » inconnu (on ne connaît pas à l'avance quand les tâches seront soumises à exécution) et leur placement doit se faire au fur et à mesure qu'elles sont proposées à exécution. Ce type de *scheduling*, que Ghobaei-Arani *et al.* [94] qualifie de « dynamique », représente un défi pour l'optimisation de l'usage des ressources, objectif souvent recherché par ces mécanismes.

Un autre aspect, souligné par [106], est la question de la propriété des ressources (*i.e.* à qui appartient les ressources). Selon Hong & Varghese [106], différentes organisations peuvent se partager des ressources afin d'optimiser leur usage ou même pour des raisons d'efficacité énergétique. Dans le cas des SIP, on peut facilement envisager cette situation, avec des organisations partenaires (et même des collaborateurs) qui partagent leurs ressources dans le cadre d'un SIP. Prenons l'exemple d'un aéroport, entre la société maintenant l'aéroport, les compagnies aériennes et tous les prestataires de services, les possibilités de partage de ressources sont significatives. On se retrouve ainsi dans un environnement multi-application, multi-utilisateur, selon la classification proposée par [106].

Au vu de cette étude, on peut considérer que la gestion de ressources dans un SIP comme un cas particulier de gestion de ressources dans lequel nous avons :

- i. Un pool de ressources hétérogène et dynamique ;
- ii. Un pool pouvant appartenir à différents propriétaires ;
- iii. Des services dont l'exécution doit être programmée en fonction de la demande (et donc pas forcément anticipée) ;

Et pour laquelle nous avons également besoin de :

- iv. Une politique de placement des services qui ne vise pas forcément l'optimisation des ressources, mais plutôt un usage opportuniste de ces ressources ;
- v. Critères (ou métriques) à observer qui peuvent varier selon les services demandés ; du même pour les critères de prise de décision qui peuvent varier selon les ressources ;
- vi. Une évolutivité et une possibilité de passage à l'échelle importants, au vu des dimensions que peut prendre un SIP.

Les deux premiers points (*i* et *ii*) peuvent être vus comme une conséquence même de la manière dont les SIP se constituent : par l'introduction des différentes technologies et des nouvelles pratiques (voir des partenariats), qui s'ajoutent à celles et ceux déjà présentes et qui complexifient l'environnement qui en résulte, tout en apportant à ces systèmes des nouvelles opportunités (*e.g.* des nouveaux services, des nouveaux processus métiers, etc.). Le 3^{ème} point correspond à la nécessité d'offrir davantage de souplesse à ces systèmes, avec une offre de services qui pourrait alors être adaptée à la

demande. Le 4^{ème} point évoqué découle du dynamisme de l'environnement, plus propice à un usage opportuniste des ressources disponibles que d'un usage dit « optimale » de celles-ci. L'objectif n'est donc pas forcément l'optimisation de ressources (par exemple, un équilibrage de la charge parmi les ressources, ou encore minimiser le nombre de ressources utilisées), ni des performances brutes (e.g. minimiser le temps d'exécution, ou maximiser le respect des SLA), mais surtout de pouvoir utiliser les ressources lorsqu'elles sont disponibles et éviter ainsi un certain « gaspillage » des ressources parfois sous-utilisées dans l'organisation. Pour cela, différentes informations peuvent être observées et utilisées pour la prise de décision, puisque différentes stratégies peuvent être considérées, en fonction du service demandé et de la ressource envisagée pour son exécution. Cette variabilité dans les informations nécessaires conduit à penser à la nécessité de définir des *politiques configurables*, à la fois pour les *services* et pour les *ressources*, et de prendre en considération les informations de contexte en tant que telles, avec une vraie gestion de contexte (et notamment de la QoC) associée.

Enfin, le dernier point considère le besoin d'évolutivité et de passage à l'échelle qui peuvent caractériser un SIP. Lorsqu'on parle d'un SI, il est possible déjà d'envisager un nombre très important de ressources et des services. Le caractère dynamique d'un SIP ne fait qu'aggraver ce constat et souligner l'importance de la prise en compte de l'évolutivité : un SIP doit pouvoir évoluer en réaction aux changements dans son environnement, mais également en fonction des changements de stratégie qui peuvent survenir au niveau business. Même si on organise les ressources d'un SIP en de multiples espaces de services, l'échelle reste un facteur important. Du même que l'évolutivité : des critères qui sont aujourd'hui importants dans la gestion de certaines ressources (ou services) peuvent évoluer et céder leur place à des nouveaux critères à l'avenir, au gré de l'évolution de l'organisation elle-même.

Tous ces considérations font dégager quatre caractéristiques majeures pour une gestion opportuniste des ressources dans un SIP : le *dynamisme*, la *sensibilité au contexte*, la *souplesse* et le *passage à l'échelle*. Or, peu de travaux étudiés dans la littérature considèrent ces aspects. Tous ces points, mis bout-à-bout, distinguent de manière considérable la gestion de ressources dans un SIP de celle dans les plateformes de *Cloud Computing*, ou même de *Fog Computing*.

Ces points ont inspiré la définition d'une architecture conceptuelle pour une gestion opportuniste des ressources dans un SIP. A l'instar de CloudFIT, l'architecture proposée, dont les éléments principaux sont illustrés dans la Figure 60, se présente comme une architecture décentralisée, dans laquelle chaque ressource décide de manière individuelle de l'exécution ou non d'un service qui lui est soumis. Une attention particulière a été donnée à la modularité dans l'architecture : chaque composant a été défini à l'aide d'interfaces d'entrée et de sortie bien établies, de manière à ce qu'il puisse évoluer avec un impact sur le fonctionnement des autres composants. Chaque composant assure donc une responsabilité bien précise dans l'architecture, offrant et consommant des services à la direction des autres composants. On retrouve ainsi (cf. Figure 60) un composant pour la gestion de la communication sur un réseau pair-à-pair (« *communication manager* ») et un « *launcher* » assurant l'exécution d'un service soumis à la file d'exécution (« *priority queue* »). On retrouve également un composant dédié à la gestion de contexte, permettant une prise en compte complète de ce type d'information (de son acquisition et sa modélisation, à son interprétation). On remarquera que plusieurs éléments de cette architecture ont été inspirés de l'architecture développée pour CloudFIT (cf. chapitre 2.2, Figure 30). On se sert ainsi de l'expérience acquise sur PER-MARE pour la définition de ce travail sur ce SIP.

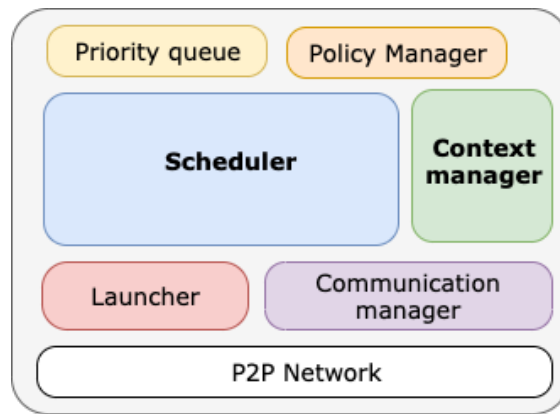


Figure 60. Représentation simplifiée de l'architecture conceptuelle pour la gestion de ressources d'un SIP.

Cependant les deux éléments centraux de cette architecture sont le « *policy manager* », qui assure la gestion et l'interprétation des politiques, aussi bien celles de services soumis que celles applicables à la ressource elle-même ; et le « *scheduler* » qui a la responsabilité sur l'exécution ou non d'un service sur la ressource, en fonction des politiques définies. Ces politiques se divisent donc en différentes catégories : celles applicables aux services et celles applicables aux ressources, mais également, les politiques définies pour une ressource en particulier (appelées politiques « locales ») et celles définies pour toute une organisation (dites politiques « globales »). Toutes ces politiques peuvent se baser sur différentes informations, et notamment des informations de contexte. On s'approche donc des définitions issues de l'espace de services des contraintes et des capacités (cf. Tableau 10), avec les contraintes qui s'appliquent sur les capacités d'une ressource.

La définition de ces composants est accompagnée par la définition d'un comportement attendu de chaque composant et de leurs interactions. Ce comportement établit ainsi une dynamique dans le fonctionnement de l'architecture afin d'assurer, entre autres, une prise de décision répartie entre les différentes ressources. Le défi ici est de garantir l'exécution de tout service sollicité. L'arrivée d'une demande auprès de la file d'attente devra déclencher une analyse de la part du *scheduler*. Celui-ci va pouvoir solliciter les politiques applicables au gestionnaire de politiques (« *policy manager* ») et obtenir les informations de contexte nécessaires à la résolution de ces politiques auprès du gestionnaire de contexte (« *context manager* »). A partir de ces informations, trois décisions peuvent être prises : l'exécution immédiate du service, sa délégation à une ressource voisine ou sa mise en attente. A chaque fois qu'un service est délégué ou mis en attente, sa priorité est revue à la hausse, de telle manière à garantir qu'il sera exécuté. L'idée serait de pouvoir exécuter un service considéré comme prioritaire, même si certaines politiques ne sont pas toutes respectées.

La définition de cette architecture, laquelle fait partie intégrante de la thèse de David Beserra, est toujours en cours. Nous travaillons en ce moment sur la définition de politiques et d'un formalisme permettant leur expression. Il est important que ces politiques puissent exprimer des contraintes relatives à l'usage des ressources, mais aussi par rapport aux services, suivant les mêmes principes établis par les définitions données dans la théorie des espaces de services (cf. section 4.1). On retrouve donc les notions de contexte requis, de contraintes et des propriétés tels que définies par l'espace de services (cf. Tableau 10). Le rôle du *scheduler* devient ainsi celui résoudre ces définitions de manière à garantir que la ressource restera dans un état considéré comme souhaitable, tout en exécutant des services au nom du SIP.

Enfin, si on positionne ce travail par rapport à la plateforme IPSOM proposée dans la thèse de Salma Najjar [170], celle-ci va procéder à la sélection de services à exécuter, à travers les mécanismes de découverte et de prédiction qui ont été proposés (cf. chapitre 3). Cette plateforme pourra alors soumettre le service choisi au gestionnaire de ressources (représenté par l'architecture proposée ci-dessus) appartenant à la ressource la plus proche dans l'espace de service. Celle-ci pourra ensuite

décider d'exécuter le service ou de le déléguer à d'autres ressources situées dans l'environnement. Ces ressources peuvent être organisées dynamiquement en « communautés » (espaces), comme le propose CloudFIT ou encore en groupes selon le contexte, comme nous avons proposés à travers le mécanisme de *context grouping* (cf. chapitre 2). L'architecture discutée dans cette section représente ainsi une première étape nécessaire vers la construction d'une plateforme d'exécution de type *middleware* permettant la réalisation des espaces de services.

4.2.3 Bibliométrie

S'agissant d'un travail toujours en cours, cette contribution a été l'objet de peu de publications. Certains résultats, jugés encore préliminaires, n'ont pas pu être soumis à publication. On ne peut considérer ici que les publications impliquant les expérimentations réalisées avec la plateforme CloudFIT. Comme pour l'espace de services (cf. section 4.1), ces publications ont visé en priorité la communauté française, afin de recueillir des retours de cette communauté qui nous semblaient nécessaires au développement de la proposition. Ces publications sont listées ci-dessous et leur impact, par rapport au nombre de citations, est résumé dans le Tableau 13. Comme attendu, un nombre particulièrement réduit de citations a pu être identifié, ne préjugant en rien l'impact que ces travaux pourront avoir sur le domaine.

- **UbiMob 2016** [241] : Steffanel, L.A. & Kirsch-Pinheiro, M., « Stratégies Multi-Échelle pour les Environnements Pervasifs et l'Internet des Objets ». *11èmes Journées Francophones Mobilité et Ubiquité (UbiMob 2016)*, 5 juillet 2016, Lorient, France. Paper n°6. Disponible sur https://ubimob2016.telecom-sudparis.eu/files/2016/07/Ubimob_2016_paper_6.pdf (Dernière visite: aout 2020)
- **EUSPN 2018** [239] : Steffanel, L.A. & Kirsch-Pinheiro, M., "Improving Data Locality in P2P-based Fog Computing Platforms", *9th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2018)*, Leuven, Belgium, November 5-8, 2018. DOI: doi:10.1016/j.procs.2018.10.151
- **IJITSA 2018** [240] : Steffanel, L.A.; Kirsch-Pinheiro, M.; Vaz Peres, L. & Kirsch Pinheiro, D. "Strategies to implement Edge Computing in a P2P Pervasive Grid", *International Journal of Information Technologies and Systems Approach (IJITSA)*, IGI Global, 11(1), 2018, 1-15. DOI: doi:10.4018/IJITSA/2018010101
- **COMPASS 2019** [238] : Steffanel, L.A. & Kirsch-Pinheiro, M., « Accès aux Données dans le Fog Computing : le cas des dispositifs de proximité », *Conférence d'informatique en Parallélisme, Architecture et Système (CompAS'19)*, 25-28 July 2019, Anglet, France. Disponible sur <https://hal.univ-reims.fr/hal-02174708> (Dernière visite : aout 2020)

Tableau 13. Analyse bibliométrique des publications liées la gestion de ressources dans les SIP.

	Année	Total	≤ 2018	> 2018	self
UbiMob 2016	2016				
EUSPN 2018	2018	2		2	
IJITSA 2018	2018	2		2	
COMPASS 2019	2019				
Total / %		4	0 %	100 %	0 %

4.3 Bilan

Contrairement aux chapitres précédents, les contributions présentées ici sont toujours en cours de développement. La notion d'espace de services, même si elle compte déjà quelques publications, se trouve toujours en cours d'évolution, plusieurs aspects demandant encore de l'attention, notamment la méthodologie de conception, qui doit être étendue pour la prise en charge des ressources.

La prise en charge de ces espaces dans une plateforme d'exécution reste également à faire. Les travaux sur la gestion opportuniste de ressources représentent un premier pas dans cette direction. Ces travaux ont démarré par une phase empirique, avec les expérimentations sur CloudFIT, et suivie d'une étude de littérature. Ces premières phases nous ont permis d'acquérir les connaissances nécessaires pour avancer vers la définition d'une architecture conceptuelle pour la gestion opportuniste de ressources, dont la première version a été discutée dans ce chapitre. Cette architecture est toujours en cours de construction dans le cadre de thèse de David Beserra. A terme, elle représentera un socle de base pour la construction d'une véritable plateforme d'exécution pour les SIP.

Des nombreuses lacunes restent encore à combler pour la réalisation de cette vision d'espaces de services qui puissent être pris en considération en temps d'exécution par une plateforme. Non seulement la gestion des ressources dans ces environnements dynamiques doit être considérée, mais également les processus métiers qui devront s'exécuter dans ces espaces à l'aide des services proposés. La collaboration démarrée avec *l'Università di Messina* s'inscrit dans ce contexte. L'utilisation de containers pour encapsuler et organiser des services et des capteurs dans des segments de processus représente une première piste de réflexion envisageable, qui doit être développée davantage à l'avenir.

Le caractère préliminaire de ces travaux ne nous permet pas de tirer un bilan sur l'impact qu'ils auront notamment dans la communauté SI. Ils représentent cependant une première étape importante vers la prise en compte des SIP. A travers les challenges considérés dans ce chapitre, nous avons voulu souligner l'importance d'arriver à une vision conceptuelle de ces systèmes, mais également d'une meilleure prise en compte des aspects techniques nécessaires à la réalisation de notre vision sur les SIP. Il en ressort de ces travaux le caractère profondément pluridisciplinaire des SIP. Le développement de ces systèmes amène avec eux des nombreux défis pour lesquels des compétences issues des différentes communautés de l'Informatique (et peut-être même au-delà de l'Informatique) seront nécessaires. Nous avons pu le voir avec les travaux sur la gestion opportuniste des ressources, pour lesquels des compétences issues des communautés HPC et *Cloud Computing*, et de manière plus générale, des Systèmes Répartis, ont dû être mobilisées.

Cette multidisciplinarité est peut-être la caractéristique la plus marquante dans l'étude des Systèmes d'Information Pervasifs. Les défis que l'apparition de ces systèmes nous imposent ne pourront être traités que par une démarche globale et pluridisciplinaire.

IV Conclusion & Perspectives

Tout au long de ce document, différentes contributions ont été présentées. Celles-ci s'étendent de 2002, date de début de ma thèse, jusqu'à présent. Elles ont pour fil conducteur la notion de contexte, laquelle a été appliquée à différentes communautés de l'Informatique.

J'ai débuté mes recherches à la communauté CSCW (*Computer Supported Cooperative Work*) par l'application de la notion de contexte à l'adaptation de l'information de conscience de groupe (chapitre 1). Il en est ressorti un modèle de contexte orienté objets et un processus de filtrage dont les principes ont inspiré d'autres travaux bien plus tard. Il y a eu ensuite mes travaux autour des environnements pervasifs, résolument ancrés dans la communauté de l'Informatique Ubiquitaire (chapitre 2). Là encore, la notion de contexte a été mise à contribution, d'abord dans un mécanisme de distribution pair-à-pair, puis dans la gestion de ressources au sein du projet PER-MARE. Ce projet était parmi les précurseurs à explorer l'usage du *Fog Computing* pour les applications de type *Big Data*. Ces travaux ont ouvert la perspective d'un usage opportuniste des ressources disponibles à partir de l'observation du contexte d'exécution de ces ressources.

Dans le chapitre 3, des travaux intégrant la communauté SOC (*Service Oriented Computing*) ont été présentés. Ceux-ci sont majoritairement issus de mon intégration au Centre de Recherche en Informatique (CRI), à l'Université Paris 1 Panthéon Sorbonne. Ils apportent la combinaison de notion d'intention, issue de l'Ingénierie des Exigences, à la notion de contexte dans l'orientation service. Ce triplet « *intention, contexte, service* » a servi de base notamment pour mes contributions sur la découverte et la prédiction de services. Ces travaux m'ont permis de porter un nouveau regard sur la question des habitudes de travail de chacun et de soulever la question de la pertinence d'une information de contexte : comment savoir si un élément de contexte caractérise ou non le choix d'une application ou d'un service ? Mes travaux autour du *context mining* apportent quelques éléments de réponse, avec notamment l'usage des techniques d'ACF (Analyse de Concepts Formels) dans une démarche d'amélioration continue. Ces travaux ont également ouvert la porte à une réflexion plus profonde sur l'applicabilité de techniques de *Machine Learning* à des données de contexte.

Toutes ces contributions convergent vers la notion de Systèmes d'Information Pervasifs (SIP), qui fait l'objet de mes dernières recherches, présentées dans le chapitre 4. Ces systèmes représentent la nouvelle génération de Système d'Information (SI). Il ne s'agit nullement des nouveaux systèmes à créer à partir de rien, mais bel et bien de l'évolution de systèmes existants, qui sont aujourd'hui bouleversés par l'arrivée des nouvelles technologies et de nouvelles pratiques. Toutes ces évolutions (*Fog Computing, IoT, Big Data, Machine Learning, etc.*) conduisent ces systèmes bien au-delà des frontières traditionnellement acceptées par les SI encore en place aujourd'hui. Les Systèmes d'Information Pervasifs s'invitent bien au-delà des limites de l'organisation, par l'intégration de l'environnement physique, des technologies mobiles, ou encore du *Cloud Computing*. Même si on parle beaucoup, notamment grâce à l'*IoT* et au *Big Data*, des données disponibles partout, il ne s'agit pas uniquement de l'information, mais véritablement d'un système qui se déploierait partout, disponible tout le temps. En somme, on parle bien de la vision de Weiser [264] sur l'Informatique Ubiquitaire qui se déploie (et bouleverse) les Systèmes d'Information.

A la base de tous ces bouleversements à l'origine des SIP se trouve un environnement devenu éminemment dynamique qu'il faut savoir maîtriser. Ce dynamisme apporte avec lui la promesse d'un Système d'Information plus souple, capable de s'adapter facilement (ou plus facilement) aux changements. La notion de contexte peut ainsi jouer un rôle clé dans cette transformation des SI dits « traditionnels » vers les Systèmes d'Information Pervasifs. Il s'agit alors d'apporter de l'adaptabilité à ces systèmes à tous les niveaux, de l'infrastructure au pilotage, en passant par tout le support aux métiers. On considère ici l'information de contexte au sens large, issue de l'observation des utilisateurs, de l'environnement physique, mais aussi de l'organisation elle-même, tel que proposé dans mes travaux de thèse (chapitre 1). Chaque niveau d'un SIP peut ainsi bénéficier de l'information de contexte pour son adaptation, comme les différentes contributions présentées dans ce document

ont pu l'illustrer. Chacune à son niveau, elles montrent qu'on peut apporter plus de réactivité aux infrastructures, aux services et aux applications, grâce à la prise en compte de l'information de contexte.

Cependant, le vrai challenge ne se retrouve pas dans l'adaptation de chaque niveau de manière isolée, mais c'est de créer une véritable synergie entre les niveaux. Chaque niveau devrait pouvoir s'adapter en fonction de ses propres conditions et objectifs, mais également en fonction des informations et des changements opérés sur les niveaux voisins. C'est toute une dynamique entre les différents niveaux d'un Système d'Information qui peut être alimentée par une prise en charge globale de la notion de contexte.

Même si l'information de contexte est vue ici essentiellement comme un levier pour l'adaptation, il ne s'agit pas de tout automatiser dans un SIP. Un SIP est un Système d'Information qui évolue, et la nature même de ces systèmes leur impose d'être prévisibles et maitrisables. On doit pouvoir maitriser un SI, ses applications, ses processus, ses services, son infrastructure, etc., quelle que soit la situation. On doit être capable de gérer un SIP malgré son hétérogénéité et son dynamisme. L'adaptation au sein d'un SIP pourra se faire certes de manière automatique, mais elle pourra aussi venir d'un pilotage actif. Les réflexions qui ont été menées sur la conscience de groupe (chapitre 1) et sur le *context mining* (chapitre 3) laissent entrevoir le potentiel de l'information de contexte pour la prise de décision. L'information de contexte peut alors devenir la pierre angulaire dans une démarche d'amélioration continue qui sera, sans nul doute, essentielle à l'apparition et à la survie des futurs SIP.

Tout ceci conduit de manière presque inéluctable à penser à la généralisation du support à la notion de contexte à tout le Système d'Information. Il s'agit alors de considérer ce support au contexte comme une « facilité ». Cette idée, introduite dans [23] (chapitre 3), représente à mon humble avis, la clé de voûte des Systèmes d'Information Pervasifs. Si on voit un SIP comme une ville, la gestion de contexte devrait s'apparenter à une « facilité » intégrée dans la ville, tel l'eau ou l'électricité, un service disponible à tous les membres de la communauté. Penser la gestion de contexte comme une « facilité » (une « *context facility* ») implique de généraliser cette notion à tout le système. Tout devient alors observable. Chaque élément dans un SIP pourrait ainsi être observé, devenir source d'information de contexte, et en même temps, consommateur de ce type d'information pour différents usages, de l'adaptation à la prise de décision.

Cette vision d'une « *context facility* », disponible pour tout le SIP, apporte avec elle des nombreux challenges, notamment liés au passage à l'échelle qu'implique cette vision : ce n'est plus une application ou un service en particulier qui bénéficie d'une telle plateforme, mais l'ensemble des éléments d'un SIP, quel que soit leur niveau. On peut illustrer ces propos par l'information de conscience de groupe. Celle-ci devient alors contexte organisationnel pour être mieux exploitée dans les collecticiels, qui n'auront plus à s'occuper de la gestion de cette information, mais également devenir contexte requis pour l'usage de certains services ou ressources. On ne capture plus une information de contexte pour un usage précis, mais de manière générale, pour différents usages, même futurs. Ceci pose la question de la modélisation de cette information, de son stockage, mais également de son traitement à très large échelle.

A travers cette idée de « *context facility* », tout devient source possible d'information de contexte. Cette information peut ainsi être remontée à tous les niveaux d'un SIP et entraîner avec elle des réactions, de l'adaptation à la prise de décision, des réactions qui peuvent à leur tour déclencher de nouveaux changements. C'est à travers la notion de contexte vue comme une « facilité » que la synergie entre les niveaux d'un SIP pourra se créer. C'est cette synergie que la Figure 61 illustre. On y voit, dans la partie inférieure de l'image, différents éléments agissant comme source de contexte et alimentant cette « facilité », qui, à son tour, rend cette information disponible à tous les niveaux du système. A chaque niveau, cette information pourra déclencher des changements, qui à leur tour alimentent cette base et la dynamique sur les autres niveaux.

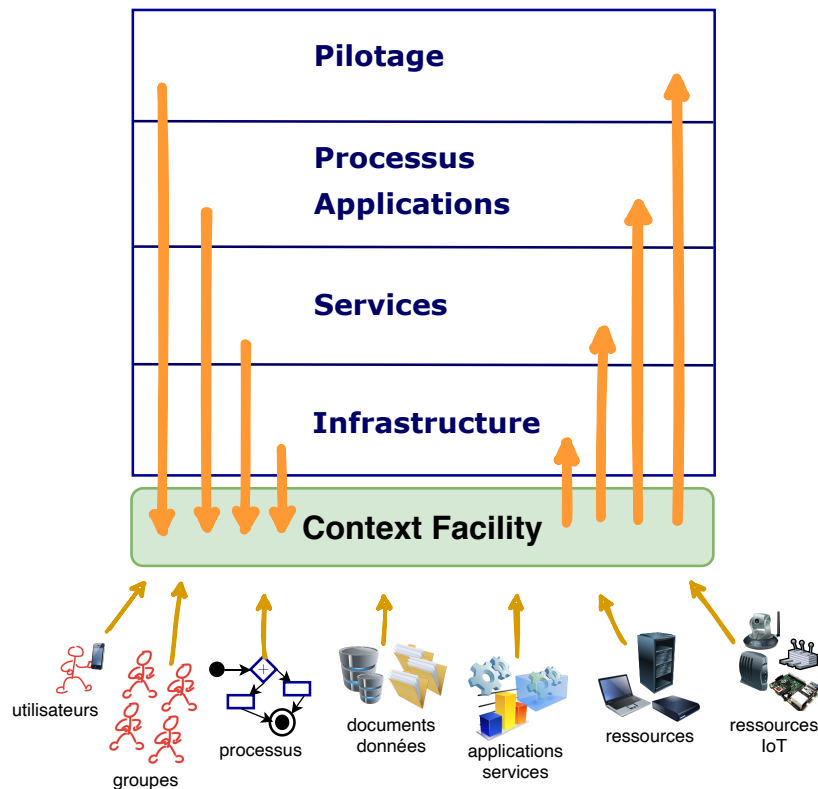


Figure 61. Context facility au service d'un SIP.

Cette synergie pourrait apporter aux Systèmes d'Information Pervasifs la souplesse dont ils auront besoin pour mieux tirer profit des opportunités offertes par les nouvelles technologies à venir et de la propre dynamique de l'environnement, qu'il soit physique, logique ou organisationnel. Cependant, les défis pour que cette vision puisse un jour devenir réalité sont nombreux et particulièrement importants. Tout d'abord, il y a les défis qu'on peut qualifier de techniques, et notamment la mise en place de plateformes et des méthodes permettant la mise en œuvre de cette vision. La construction d'une plateforme pour l'exécution des espaces de services, mentionnée dans le chapitre 4, ou encore le passage à l'échelle des techniques de *Machine Learning*, discutée dans le chapitre 3, font partie de ces défis techniques. Néanmoins, ces défis mentionnés dans ce document sont loin d'être les seuls. Les questions sur la sécurité des données et des infrastructures, ou encore la robustesse de ces environnements très dynamiques commencent à se faire sentir dès à présent.

Ces défis techniques sont accompagnés par des défis d'ordre méthodologique, et notamment le développement des modèles et des méthodologies pour la gestion et le pilotage des SIP. La définition de la théorie des espaces de service (chapitre 4) constitue pour moi une première piste de réflexion dans ce sens, piste qui doit encore être approfondie dans les années à venir. Des très nombreuses questions restent, en effet, en suspens, d'un point de vue méthodologique. Par exemple, les questions liées à la sécurité (aussi bien des données et des infrastructures que des applications et des processus), la confidentialité et même la confiance qu'un utilisateur peut avoir sur le système ou que le système peut avoir sur ses composants, sont autant de questions qui doivent être soulevées d'un point de vue méthodologique aussi bien que technique. Cette synergie entre les niveaux, devenue de plus en plus nécessaire, ouvre la porte à plus de coopération entre les différents acteurs engagés, et à l'intégration de ces différents points de vue dans les méthodologies à venir.

Enfin, les SIP et cette vision d'une « *context facility* » posent également des questions sur le plan social et humain. Sommes-nous prêts à accepter un tel niveau d'observation de notre quotidien en entreprise ? Saurons-nous accepter l'automatisation grandissante de notre environnement de travail ?

Comme toute nouvelle technologie, comme tout changement, tous ces bouleversements apportent leur lot d'espoir et de crainte, que nous, en tant que société, devrions apprendre à équilibrer. Ces derniers défis échappent largement à mes domaines de compétences, mais je suis curieuse de savoir où l'avenir nous mènera.

V Références

1. Aburukba, R. O.; AliKarrar, M.; Landolsi, T. & El-Fakih, K., "Scheduling Internet of Things requests to minimize latency in hybrid Fog-Cloud computing", *Future Generation Computer Systems*, 111, **2020**, 539 - 551. <https://doi.org/10.1016/j.future.2019.09.039>
2. Adomavicius G. & Tuzhilin A., "Context-Aware Recommender Systems". In: Ricci F., Rokach L., Shapira B., Kantor P. (eds), *Recommender Systems Handbook*. Springer, **2011**, 217-253
3. Alarcon, R. & Fuller, D., "Intelligent awareness in support of collaborative virtual work groups", Haake, J. & Pino, J. (Eds.), *8th International Workshop on Groupware: Design, Implementation and Use (CRIWG 2002)*, LNCS 2440, Springer-Verlag, **2002**, 168-188
4. Alarcon, R.; Collazos, C. & Guerrero, L., "Distributed shared contexts", Karmouch, A.; Korba, L. & Madeira, E. (Eds.), *1st International Workshop on Mobility Aware Technologies and Applications (MATA 2004)*, LNCS 3284, Springer-Verlag, **2004**, 27-36
5. Ali, M.; Vlaskamp, J.H.A.; Eddin, N.N.; Falconer, B. & Oram, C., "Technical development and socioeconomic implications of the Raspberry Pi as a learning tool in developing countries", *Computer Science and Electronic Engineering Conf. (CEECE)*, IEEE, **2013**, 103-108
6. Aljoumaa, K.; Assar, S. & Souveyet, C., "Reformulating User's Queries for Intentional Services Discovery Using an Ontology-Based Approach", *4th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*, **2011**, 1-4
7. Alonso, G.; Casati, F.; Kuno, H. & Machiraju, V., "Web Services: Concepts, Architectures and Applications". Springer, **2004**, p. 354
8. Alrawais, A.; Alhothaily, A.; Hu, C. & Cheng, X. "Fog Computing for the Internet of Things: Security and Privacy Issues". *IEEE Internet Computing*, 21(2), **2017**, 34-42. <https://doi.org/10.1109/MIC.2017.37>
9. Ameyed, D.; Miraoui, M. & Tadj, C., "A survey of prediction approach in pervasive computing", *International Journal of Scientific & Engineering Research*, 6(5), **2015**, 306-316
10. Andriole, S.J. & Bojanova, I., "Optimizing operational and strategic IT", *IEEE IT Professional*, 16(5), September/October **2014**, 12-15
11. Ausiello, G., « Cooperative data stream analysis and processing », In : Ferscha, A. (Ed.), *Pervasive Adaptation: Next generation pervasive computing research agenda*, **2011**, 15. Disponible sur <https://www.pervasive.jku.at/rab/pdf/PerAda%20Research%20Agenda.pdf> (dernière visite : Sept. 2019)
12. Baaziz, A. & Quoniam, L. "How to use Big Data technologies to optimize operations in Upstream Petroleum Industry". *21st World Petroleum Congress*, Jun **2014**, Moscow, Russia. 21. Disponible sur <https://hal.archives-ouvertes.fr/hal-01015137/> (dernière visite : novembre 2019)
13. Baldauf, M.; Dustdar, S. & Rosenberg, F., "A survey on context-aware systems", *International Journal of Ad Hoc and Ubiquitous Computing*, 2 (4), **2007**, 263-277
14. Barbosa, J. G. & Moreira, B., "Dynamic scheduling of a batch of parallel task jobs on heterogeneous clusters", *Parallel Computing*, 37(8), **2011**, 428 - 438. <https://doi.org/10.1016/j.parco.2010.12.004>
15. Bardram, J.E., "The Java Context Awareness Framework (JCAF) – a service infrastructure and programming framework for context-aware applications", In : Gellersen, H.W, Want, R., Schmidt, A. (Eds), *LNCS 3468 - Third International Conference in Pervasive Computing (Pervasive'2005)*, Munich, Germany, May 8-13, **2005**, Springer- Verlag, 98-115
16. Bauer, C. & Dey, A., "Considering context in the design of intelligent systems: Current practices and suggestions for improvement", *Journal of Systems and Software*, 112, **2016**, Elsevier, 26-47
17. Bauer, J. S.; Newman, M. W. & Kientz, J. A., "Thinking About Context: Design Practices for Information Architecture with Context-Aware Systems", *iConference 2014 Proceedings*, **2014**, 398-411. DOI: 10.9776/14116.
18. Becker, T. & Intoyoad, W., "Context Aware Process Mining in Logistics", *Procedia CIRP*, 63, **2017**, 557 - 562
19. Bell, D., "Ubiquitous Information Systems (UBIS): A design research study of intelligent middleware and architecture", *UK Academy for Information Systems Conference Proceedings 2009 (UKAIS*

- 2009). Article n° 12, **2009**. Disponible sur : <https://aisel.aisnet.org/ukais2009/12> (Dernière visite : août 2020)
20. Bell, G. & Dourish, P., "Yesterday's tomorrows: notes on ubiquitous computing's dominant vision", *Personal and Ubiquitous Computing*, 11 (2), Jan. **2007**, 133-143
 21. Bellavista, P.; Corradi, A.; Fanelli, M. & Foschini, L. "A survey of context data distribution for mobile ubiquitous systems", *ACM Computing Survey*, **2013**, 45(1), 1-49
 22. Ben Mokhtar, S.; Preuveneers, D.; Georgantas, N.; Issarny, V. & Berbers, Y., "EASY: Efficient semAntic Service discoverY in pervasive computing environments with QoS and context support". *Journal of System and Software*, vol. 81, issue 5, **2008**, 785–808
 23. Ben Rabah, N.; Kirsch Pinheiro, M.; Le Grand, B.; Jaffal, A. & Souveyet, C., "Machine Learning for a Context Mining Facility", *16th Workshop on Context and Activity Modeling and Recognition, 2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, **2020**, 678-684
 24. Bettini, C.; Brdiczka, O.; Henriksen, K.; Indulska, J.; Nicklas, D.; Ranganathan, A. & Riboni, D., "A survey of context modelling and reasoning techniques", *Pervasive and Mobile Computing*, 6(2), Apr **2010**, 161-180, **2010**.
 25. Birnbaum, J., "Pervasive information systems", *Communications of the ACM*, 40(2), Feb. **1997**, 40–41. DOI: 10.1145/253671.253695
 26. Bittencourt, L. F.; Goldman, A.; Madeira, E. R.; da Fonseca, N. L. & Sakellariou, R., "Scheduling in distributed systems: A cloud computing perspective", *Computer Science Review*, 30, **2018**, Elsevier, 31 - 54
 27. Blichmann, G. & Meißner, K., "Customizing workspace awareness by non-programmers". *ACM SIGCHI Symp. on Engineering Interactive Comp. Systems*, **2017**, 123-128
 28. Bonomi, F.; Milito, R.; Zhu, J. & Addepalli, S. "Fog computing and its role in the internet of things". *Proceedings of the 1st MCC Workshop on Mobile Cloud Computing (MCC '12)*, ACM, **2012**, 13–16
 29. Borges, M. R.; Brézillon, P.; Pino, J. A. & Pomerol, J.-C., "Groupware System Design and the Context Concept", Shen, W.; Lin, Z.; Barthes, J.-P. A. & Li, T. (Eds.), *8th International Conference Computer Supported Cooperative Work in Design - Revised Selected Papers (CSCWD 2004)*, LNCS 3168, Springer-Verlag., **2004**, 45-54
 30. Bouthier, C., « Mise en contexte de la conscience de groupe : adaptation et visualisation », Thèse de Doctorat, Institut National Polytechnique de Lorraine, Nancy, France, **2004**.
 31. Breitbach, M.; Schäfer, D.; Edinger, J. & Becker, C., "Context-Aware Data and Task Placement in Edge Computing Environments", *2019 IEEE International Conference on Pervasive Computing and Communications (PerCom 2019)*, **2019**, 1-10
 32. Brézillon, J. & Brézillon, P., "Context Modeling: Context as a Dressing of a Focus". In: Kokinov, B.; Richardson, D.; Roth-Berghofer, T. & Vieu, L. (Eds.), *Modeling and Using Context*, Springer, LNCS 4635, p. 136-149, 2007.
 33. Brézillon, P., "Context-Based Development of Experience Bases". In: Brézillon, P.; Blackburn, P. & Dapoigny, R. (Eds.), *8th Int. and Interdisciplinary Conf. on Modeling and Using Context (CONTEXT)*, p. 87-100, 2013.
 34. Brézillon, P., "Contextual Modeling of Group Activity", In: Brézillon, P.; Turner, R. & Penco, C. (Eds.), *Modeling and Using Context (Context 2017)*, LNCS 10257, Springer International Publishing, **2017**, 113-126
 35. Brézillon, P., "Expliciter le contexte dans les objets communicants". In: Kintzig, C., Poulain, G., Privat, G., Favennec, P.-N. (Eds.), *Objets Communicants*, Hermes Science Publications, p. 293–303, 2002.
 36. Brown, P.; Bovey, J. & Chen, X., "Context-aware applications: from the laboratory to the marketplace", *IEEE Personal Communications*, vol. 4 issue 5, p. 58-64, 1997.
 37. Brusilovsky, P. & Maybury, M. T., "From adaptive hypermedia to the adaptive Web", *Communication of ACM*, ACM, 45 (5), May **2002**, 30-33

38. Bucur, O. ; Beaume, P. & Boissier, O., « Définition et représentation du contexte pour des agents sensibles au contexte », In: Coutaz, J., Lecomte, S. (Eds.), *Actes des deuxièmes journées francophones: Mobilité et Ubiquité 2005 (UbiMob'05)*, 31 mai – 3 juin **2005**, Grenoble, 13-16
39. Burkhart, T.; Weis, B.; Werth, D. & Loos, P., "Towards Process-Oriented Recommender Capabilities in Flexible Process Environments--State of the Art", *45th Hawaii International Conference on System Sciences*, **2012**, 4386-4395
40. Carnevale, L. ; Celesti, A. ; Galletta, A. ; Dustdar, S. & Villari, M., "From the Cloud to Edge and IoT: a Smart Orchestration Architecture for Enabling Osmotic Computing", *32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, Krakow, **2018**, 419-424
41. Carrillo Ramos, A. C., "Ubiquitous agents for an adapted access to information systems: the framework PUMAS", PhD Thesis, Université Joseph-Fourier - Grenoble I, **2007**
42. Carrillo-Ramos, A.; Kirsch Pinheiro, M.; Villanova-Oliver, M.; Gensel, J. & Berbers, Y., "Collaborating agents for adaptation to mobile users", In: Chevalier, M.; Soule-Dupuy, C. & Julien, C. (Eds.), *Collaborative and Social Information Retrieval and Access: Techniques for Improved User Modeling*, IGI Global, **2009**, 250-276
43. Casavant, T. L. & Kuhl, J. G., "A taxonomy of scheduling in general-purpose distributed computing systems", *IEEE Transactions on Software Engineering*, **1988**, 14 (2) , 141-154
44. Cassales, G.W.; Charao, A.; Kirsch-Pinheiro, M.; Souveyet, C. & Steffemel, L.A., "Context-Aware Scheduling for Apache Hadoop over Pervasive Environments", *The 6th International Conference on Ambient Systems, Networks and Technologies (ANT 2015)*, London, UK, June 2 - 5, 2015. *Procedia Computer Science*, vol. 52, Jun **2015**, Elsevier, 202–209. doi: 10.1016/j.procs.2015.05.058.
45. Cassales, G.W.; Charão, A.S.; Kirsch-Pinheiro, M.; Souveyet, C. & Steffemel, L.A. "Bringing Context to Apache Hadoop", In: Jaime Lloret Mauri, Christoph Steup & Sönke Knoch (Eds.), *8th International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2014)*, August 24 - 28, **2014**, Rome, Italy, ISBN: 978-1-61208-353-7, IARIA, 252-258.
46. Cassales, G.W.; Charão, A.S.; Kirsch-Pinheiro, M.; Souveyet, C. & Steffemel, L.-A. "Improving the performance of Apache Hadoop on pervasive environments through context-aware scheduling", *Journal of Ambient Intelligence and Humanized Computing*, 7(3), **2016**, 333-345.
47. Castro-Leon, E., "Consumerization in the IT service ecosystem", *IEEE IT Professional*, 16(5), September/October **2014**, 20–27
48. Cavallo, M.; Cusma, L.; Modica, G.D.; Polito, C. & Tomarchio, O. "A scheduling strategy to run Hadoop jobs on geodistributed data", *3rd Workshop on Cloud for IoT (CLIoT 2015), in conjunction with the European Conference on Service-Oriented and Cloud Computing (ESOCC 2015)*, **2015**
49. Chaari, T.; Laforest, F. & Celentano, A., "Adaptation in context-aware pervasive information systems: the SECAS project", *Journal of Pervasive Computing and Communications*, vol. 3-4, 2007.
50. Chaari, T.; Dejene, E.; Laforest, F.; Scuturici, V.-M. "Modeling and Using Context in Adapting Applications to Pervasive Environments". *IEEE Int. Conf. on Pervasive Services (ICPS'06)*, p. 111-120, 2006.
51. Chabridon, S.; Conan, D.; Abid, Z.; Taconet, C. "Building ubiquitous QoC-aware applications through model-driven software engineering". *Sci. Comput. Program.*, vol. 78, p. 1912–1929, 2013.
52. Chalmers, D.; Dulay, N. & Sloman, M. "Towards Reasoning About Context in the Presence of Uncertainty", *1st Int. workshop on advanced context modelling, reasoning and management*. Nottingham, UK, September **2004**. Disponible sur : <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.58.7038&rep=rep1&type=pdf> (Dernière visite Mai 2020).
53. Chang, J.M.; Ho, P.-C. & Chang, T.-C., "Securing BYOD", *IEEE IT Professional*, 16(5), September/October **2014**, 9-11
54. Chapela-Campa, D.; Mucientes, M. & Lama, M., "Mining frequent patterns in process models", *Information Sciences*, 472, Jan. **2019**, Elsevier, 235 – 257
55. Chen, J.; Wang, S. & Thiele, L., "Proactive Speed Scheduling for Real-Time Tasks under Thermal Constraints", *2009 15th IEEE Real-Time and Embedded Technology and Applications Symposium*, **2009**, 141-150. DOI: [10.1109/RTAS.2009.30](https://doi.org/10.1109/RTAS.2009.30)

56. Chen, N.; Yang, Y.; Zhang, T.; Zhou, M.; Luo, X. & Zao, J. K. "Fog as a Service Technology", *IEEE Communications Magazine*, 56(11), November **2018**, 95-101
57. Chen, Q. ; Zhang, D. ; Guo, M. ; Deng, Q. & Guo, S. "SAMR: a self-adaptive MapReduce scheduling algorithm in heterogeneous environment", *Proceedings of the 2010 10th IEEE International Conference on Computer and Information Technology. (CIT '10)*, IEEE Computer Society, Washington, DC, USA, ISBN : 978-0-7695-4108-2, **2010**, 2736–2743.
58. Chen, S.; Zhang, T. & Shi, W., "Fog computing", *IEEE Internet Computing*, 21(2), March **2017**, 4 – 6
59. Cheverest, K.; Mitchell, K. & Davies, N., "The role of adaptive hypermedia in a context-aware tourist guide", *Communication of ACM*, 45, **2002**, 47–51.
60. Chhabra, A.; Singh, G. & Kahlon, K. S., "Multi-criteria HPC task scheduling on IaaS cloud infrastructures using meta-heuristics", *Cluster Computing*, **2020**, <https://doi.org/10.1007/s10586-020-03168-1>
61. Colman, A.; Hussein, M.; Han J. & Kapuruge, M., "Context Aware and Adaptive Systems". In: Brézillon, P. & Gonzalez, A. J. (Eds.), *Context in Computing*, Springer, **2014**, 63-82.
62. Coronato, A. & Pietro, G. D., "MiPeG: A middleware infrastructure for pervasive grids", *Future Generation Computer Systems*, 24 (1), **2008**, 17-29
63. Coutaz, J.; Crowley, J.; Dobson, S. & Garlan, D., "Context is the key", *Communications of the ACM*, 48 (3), **2005**, 49-53.
64. Da, K. « Plateforme d'adaptation autonome contextuelle à base de connaissances », PhD thesis, Université de Pau et des Pays de l'Adour, Septembre **2014**
65. Da, K.; Dalmau, M. & Roose, « Kalimucho : Plateforme de supervision d'applications sensibles au contexte », Exposito, E. & Zouari, M. (Eds.), *6^{ème} Conférence francophone sur les Architectures Logicielles (CAL 2012)*, *Revue des Nouvelles Technologies de l'Information RNTI L-7*, **2012**.
66. Da, K.; Roose, P.; Dalmau, M.; Nevado, J. & Karchoud, R. "Kali2Much: a context middleware for autonomic adaptation-driven platform", *Proceedings of the 1st Workshop on Middleware for Context-Aware Applications in the IoT (M4IoT@Middleware 2014)*, p. 25–30, 2014.
67. Decouchant, D.; Mendoza, S.; Sanchez, G. & Rodrigues, J., "Adapting groupware systems to changes in the collaborator's context of use". *Expert Syst. with App.*, 40, **2013**, 4446-4462
68. Degas, A.; Arcangeli, J.-P.; Trouilhet, S.; Calvary, G.; Coutaz, J.; Lavirotte, S. & Tigli, J.-Y. "Opportunistic Composition of Human-Computer Interactions in Ambient Spaces", *Workshop on Smart and Sustainable City, The Smart World Congress*, **2016**.
69. Deneckère R.; Hug C.; Khodabandelou G. & Salinesi C., "Intentional Process Mining: Discovering and Modeling the Goals Behind Processes using Supervised Learning", *International Journal of Information System Modeling and Design*, 5(4), **2014**, IGI Global, 22 - 47
70. Devlic, A. "SIP-based Context Distribution: Does Aggregation Pay off?", *ACM SIGCOMM Computer Communication Review*, 40(5), **2010**, 35-46.
71. Devlic, A., "Context-addressed communication dispatch", KTH Information and Communication Technology, **2009**. Disponible sur : <http://www.diva-portal.org/smash/get/diva2:213988/FULLTEXT03.pdf> (dernière visite : Juin 2019).
72. Devlic, A. & Klintskog, E., "Context retrieval and distribution in a mobile distributed environment". In *Proceedings of the Third Workshop on Context Awareness for Proactive Systems (CAPS 2007)*. Guildford, UK. **2007**.
73. Devlic, A.; Reichle, R.; Wagner, M.; Kirsch Pinheiro, M.; Vanrompay, Y.; Berbers, Y. & Valla, M., "Context inference of users' social relationships and distributed policy management", *6th IEEE Workshop on Context Modeling and Reasoning (CoMoRea)*, *7th IEEE International Conference on Pervasive Computing and Communication (PerCom'09)*, Galveston, Texas, 13 March **2009**. DOI : 10.1109/PERCOM.2009.4912890
74. Dey, A. K., "Understanding and using context", *Personal and Ubiquitous Computing*, 5(1), **2001**, 4-7.
75. Dey, A.K., "Intelligibility in ubiquitous computing systems". In : Ferscha, A., *Pervasive Adaptation: Next generation pervasive computing research agenda*, 68-69, **2011**. Disponible sur <https://www.pervasive.jku.at/Conferences/fet11/RAB.pdf> (dernière visite: août 2020).

76. Dorn, C.; Burkhart, T.; Werth, D. & Dustdar S., "Self-adjusting Recommendations for People-Driven Ad-Hoc Processes". In: Hull, R., Mendling, J. & Tai, S. (eds), *Business Process Management. BPM 2010*. Lecture Notes in Computer Science, vol 6336, **2010**, Springer, 327-342
77. Dourish, P. & Bellotti, V., "Awareness and Coordination in Shared Workspaces", *Proceedings of ACM Conference on Computer-Supported Cooperative Work (CSCW 92)*, ACM Press, **1992**, 107-114
78. Earley, S.; Harmon, R.; Lee, M.R. & Mithas, S., "From BYOD to BYOA, Phishing, and Botnets". *IEEE IT Professional*, 16(5), September/October **2014**, 16–18
79. Edinger, J.; Schäfer, D.; Krupitzer, C.; Raychoudhury, V. & Becker, C., "Fault-avoidance strategies for context-aware schedulers in pervasive computing systems", *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, **2017**, 79-88
80. Elazhary, H. "Internet of Things (IoT), mobile cloud, cloudlet, mobile IoT, IoT cloud, fog, mobile edge, and edge emerging computing paradigms: Disambiguation and research directions", *Journal of Network and Computer Applications*, 128, Elsevier, **2019**, 105-140.
81. Elkhatib, Y.; Porter, B.; Ribeiro, H. B.; Zhani, M. F. ; Qadir, J. & Rivière, E., "On using micro-clouds to deliver the fog", *IEEE Internet Computing*, 21 (2), **2017**, 8–15
82. Engel, T.A.; Charao, A.; Kirsch-Pinheiro, M. & Steffanel, L.A. "Performance Improvement of Data Mining in Weka through Multi-core and GPU Acceleration: opportunities and pitfalls", *Journal of Ambient Intelligence and Humanized Computing*, Springer, June **2015**. doi:10.1007/s12652-015-0292-9.
83. Engel, T.A.; Charao, A.; Kirsch-Pinheiro, M. & Steffanel, L.A., "Performance Improvement of Data Mining in Weka through GPU Acceleration", *5th International Conference on Ambient Systems, Networks and Technologies (ANT 2014)*, Hasselt, Belgium, June 2 - 5, 2014. *Procedia Computer Science*, 32, **2014**, Elsevier, 93–100.
84. Fensel, D.; Facca, F.M.; Simperl, E. & Toma, I., "*Semantic Web Services*", Springer, **2011**, p. 350
85. Ferguson-Boucher, K., "Cloud Computing: A Records and Information Management Perspective", *IEEE Security & Privacy*, 9(6), Nov.-Dec. **2011**, 63-66. doi: 10.1109/MSP.2011.159
86. Flauzac, O.; Krajecki, M. & Steffanel, L.A. "CONFIIT: a middleware for peer-to-peer computing". *Journal of Supercomputing*, 53(1), July **2010**, Springer, 86-102.
87. Floch, J.; Frà, C.; Fricke, R.; Geihs, K.; Wagner, M.; Lorenzo, J.; Soladana, E.; Mehlhase, S.; Paspallis, N.; Rahnama, H.; Ruiz, P.A. & Scholz, U., "Playing MUSIC: building context-aware and self-adaptive mobile applications". *Softw.: Pract. Exp.*, 43, **2013**, 359-388.
88. Garcia Lopez, P.; Montresor, A.; Epema, D.; Datta, A.; Higashino, T.; Iamnitchi, A.; Barcellos, M.; Felber, P. & Riviere, E. "Edge-centric Computing: Vision and Challenges", *ACM SIGCOMM Computer Communication Review*, 45 (5), **2015**, 37-42.
89. García, K., "Ascertaining the Availability of Shared Resources in Ubiquitous Collaborative Environments", PhD Thesis, Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, **2013**.
90. García, K.; Kirsch-Pinheiro, M.; Mendoza, S. & Decouchant, D. "Ontology-Based Resource Discovery in Pervasive Collaborative Environments". In: Antunes, P., Gerosa, M.A., Sylvester, A., Vassileva, J., & de Vreede, G.-J. (eds.), *19th Int. Conf. on Collaboration and Technology (CRIWG 2013)*, LNCS 8224. Springer, **2013**, 233–240
91. García, S.; Luengo, J. & Herrera, F., "Tutorial on practical tips of the most influential data preprocessing algorithms in data mining", *Knowledge-Based Systems*, 98, 15 April **2016**, 1-29
92. Geihs, K.; Reichle, R.; Wagner, M. & Khan, M.U., "Modeling of Context-Aware Self-Adaptive Applications in Ubiquitous and Service-Oriented Environments", In: Cheng, B.H.C., de Lemos, R., Giese, H., Inverardi, P. & Magee, J. (Eds.), *Software Engineering for Self-Adaptive Systems, Lecture Notes in Computer Science*, 5525, **2009**, 146-163
93. Gensel, J. ; Villanova-Oliver, M. & Kirsch-Pinheiro, M., « Modèles de contexte pour l'adaptation à l'utilisateur dans des Systèmes d'Information Web collaboratifs », *8èmes Journées Francophones d'Extraction et Gestion des Connaissances (EGC'08)*, *Atelier sur la Modélisation Utilisateur et Personnalisation d'Interfaces Web*, **2008**, 5-15

94. Ghobaei-Arani, M. ; Soury, A. & Rahmanian, A. A., "Resource Management Approaches in Fog Computing: a Comprehensive Review", *Journal of Grid Computing*, vol. 18, **2019**, Springer, 1-42. DOI 10.1007/s10723-019-09491-1
95. Gradecki, J.D., "*Mastering JXTA: Building Java Peer-to-Peer Applications*", John Wiley & Sons, ISBN: 0-471-25084-8, **2002**, p. 552.
96. Greenberg, S., "Context as a dynamic construct", *Human-Computing Interaction*, vol. 16, n° 2-4, **2001**, 257-268
97. Grudin, J., "Desituating action: digital representation of context", *Human-Computing Interaction*, vol. 16, n° 2-4, **2001**, 269-286
98. Gupta, A.; Faraboschi, P.; Gioachin, F.; Kale, L. V.; Kaufmann, R.; Lee, B.; March, V.; Milojevic, D. & Suen, C. H., "Evaluating and Improving the Performance and Scheduling of HPC Applications in Cloud", *IEEE Transactions on Cloud Computing*, 4(3), **2016**, 307-321
99. Gutwin, C. & Greenberg, S., "A Descriptive Framework of Workspace Awareness for Real-Time Groupware", *Computer Supported Cooperative Work (CSCW)*, 11(3-4), sept. **2002**, Kluwer Academic Publishers, 411 – 446
100. Hagrais, H. "Intelligent pervasive adaptation in shared spaces", In : Ferscha, A. (Ed.), *Pervasive Adaptation: Next generation pervasive computing research agenda*, **2011**, 16. Disponible sur <https://www.pervasive.jku.at/rab/pdf/PerAda%20Research%20Agenda.pdf> (dernière visite : Sept. 2019).
101. Hao, Z.; Novak, E. ; Yi, S. & Li, Q., "Challenges and software architecture for fog computing", *IEEE Internet Computing*, 21(2), **2017**, 44–53. DOI:doi.ieeecomputersociety.org/10.1109/MIC.2017.26
102. Hauser, M.; Günther, S. A.; Flath, C. M. & Thiesse, F., "Designing Pervasive Information Systems: A Fashion Retail Case Study", *Proceedings of the 38th International Conference on Information Systems (ICIS 2017)*, Seoul, South Korea, Dec. **2017**. Disponible sur https://www.researchgate.net/publication/319998755_Designing_Pervasive_Information_Systems_A_Fashion_Retail_Case_Study (Dernière visite : août 2020)
103. Henricksen, K.; Indulska, J. & Rakotonirainy, A., "Modeling context information in pervasive computing systems". In Mattern, F. & Naghshineh, M., *1st Int. Conference in Pervasive Computing (Pervasive 2002), Lecture Notes in Computer Science, 2414*, August 26–28 **2002**, 167-180
104. Hermosillo, G.; Seinturier, L. & Duchien, L., "Using Complex Event Processing for Dynamic Business Process Adaptation", *7th IEEE 2010 International Conference on Services Computing (SCC 2010)*, Jul 2010, Miami, Florida, United States. IEEE Computer Society, **2010**, 466-473
105. Hofmann, P.; Woods, D., "Cloud computing: The limits of public clouds for business applications", *IEEE Internet Computing*, 14 (6), Nov. **2010**, 90–93.
106. Hong, C.-H. & Varghese, B., "Resource Management in Fog/Edge Computing: A Survey on Architectures, Infrastructure, and Algorithms", *ACM Computing Survey*, 52(5), **2019**, article n° 97. DOI:10.1145/3326066
107. Hong, J.; Suh, E.-H.; Kim, J. & Kim, S., "Context-aware system for proactive personalized service based on context history", *Expert Systems with Applications*, 36(4), **2009**, 7448 – 7457
108. Hoyos, J.R.; Preuveneers, D. & García-Molina, J.J., "Quality Parameters as Modeling Language Abstractions for Context-Aware Applications: An AAL Case Study", In: Brézillon P., Turner R. & Penco C. (Eds), *Modeling and Using Context. CONTEXT 2017. Lecture Notes in Computer Science, vol. 10257*, p. 569-581, **2017**
109. Hu, X.; Ding, Y.; Paspallis, N.; Bratskas, P.; Papadopoulos, G.A.; Vanrompay, Y.; Kirsch Pinheiro, M. & Berbers, Y., "A Hybrid Peer-to-Peer Solution for Context Distribution in Mobile and Ubiquitous Environments", In: Papadopoulos G., Wojtkowski W., Wojtkowski G., Wrycza S., Zupancic J. (eds), *17th International Conference on Information Systems Development (ISD2008), Information Systems Development: Towards a Service Provision Society*, **2008**, Springer, 501-510. DOI : 10.1007/b137171_52
110. Huang, D. & Wu, H. "Mobile Cloud Computing: foundations and service models". Morgan Kaufmann, **2017**, p. 307. ISBN: 978-0-12-809641-3

111. Hughes, D.; Greenwood, P.; Blair, G.; Coulson, G.; Grace, P.; Pappenberger, F.; Smith, P. & Beven, K., "An Experiment with Reflective Middleware to Support Grid-based Flood Monitoring", *Concurr. Comput.: Pract. Exper.*, John Wiley and Sons Ltd., 20(11), **2008**, 1303-1316.
112. Hutter, F.; Kotthoff, L. & Vanschoren, J., "Automated Machine Learning: Methods, Systems, Challenges", Springer, **2019**
113. IBM, "What is edge computing?". Disponible sur : <https://www.ibm.com/cloud/what-is-edge-computing> (dernière visite : avril **2019**).
114. Issarny, V.; Caporuscio, M. & Georgantas, N., "A Perspective on the Future of Middleware-based Software Engineering". *Future of Software Engineerin (FOSE)*, **2007**, 244–258
115. Jackson, M., "Software requirements specifications: a lexicon of practice, principles and prejudices", ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, **1995**, p. 228
116. Jaffal, A. ; Grand, B. L. & Kirsch-Pinheiro, M., "Refinement Strategies for Correlating Context and User Behavior in Pervasive Information Systems", *1st Workshop on Big Data and Data Mining Challenges on IoT and Pervasive (Big2DM)*, *6th International Conference on Ambient Systems, Networks and Technologies (ANT-2015)*, *Procedia Computer Science*, vol. 52, Jun **2015**, 1040-1046. DOI : <http://dx.doi.org/10.1016/j.procs.2015.05.103>
117. Jaffal, A. ; Kirsch-Pinheiro, M. & Le Grand, B., "Unified and Conceptual Context Analysis in Ubiquitous Environments", In : Jaime Lloret Mauri, Christoph Steup & Sönke Knoch (Eds.), *8th International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2014)*, August 24 - 28, 2014, ISBN 978-1-61208-353-7, IARIA, **2014**, 48-55
118. Jaffal, A., « Aide à l'utilisation et à l'exploitation de l'Analyse de Concepts Formels pour des non-spécialistes de l'analyse des données », Thèse de doctorat, Université Paris 1 Panthéon Sorbonne, **2019**. Disponible sur : <https://tel.archives-ouvertes.fr/tel-02526323/> (Dernière visite : juillet 2020)
119. Jaffal, A., « Analyse formelle de concepts pour la gestion du contexte », Dissertation de master, Master Recherche Systèmes d'Information et de Décision, Université Paris 1 Panthéon Sorbonne, **2013**. p. 47
120. Jaffal, A.; Grand, B. L. & Kirsch-Pinheiro, M., « Extraction de connaissances dans les Systèmes d'Information Pervasifs par l'Analyse Formelle de Concepts », *Extraction et Gestion des Connaissances (EGC 2016)*, *Revue des Nouvelles Technologies de l'Information*, RNTI-E-30, **2016**, 291-296
121. Kaabi, R.S. & Souveyet, C., "Capturing Intentional services with Business Process Maps". *1st Int. Conference on Research Challenges in Information Science (RCIS)*, **2007**, 309- 318
122. Khambatti, M.; Dong Ryu, K. & Dasgupta, P., "Structuring peer-to-peer networks using interest-based communities", In: Aberer, K., Kalogeraki, V., Koubarakis, M. (eds.), *VLDB 2003, Lecture Notes in Computer Science*, 2944, **2003**, 48–63.
123. Khan, M.U., "Unanticipated Dynamic Adaptation of Mobile Applications", PhD thesis (Doktor der Ingenieurwissenschaften), University of Kassel, German, 31 March **2010**
124. Kirsch Pinheiro, M. ; Carrillo-Ramos, A. ; Villanova-Oliver, M. ; Gensel, J. & Berbers, Y., "Context-Aware Adaptation in Web-based Groupware Systems", In: JingTao Yao (Ed.), *Web-based Support Systems, Series: Advanced Information and Knowledge Processing*, Springer, mars **2010**, 3-31
125. Kirsch Pinheiro, M. « Adaptation Contextuelle et Personnalisée de l'Information de Conscience de Groupe au sein des Systèmes d'Information Coopératifs ». Thèse de Doctorat, Université Joseph-Fourier - Grenoble I, **2006**. p. 267
126. Kirsch-Pinheiro, M.; de Lima, J.V. & Borges, M.R.S., "A framework for awareness support in groupware systems", *Computers in Industry*, 52(1), **2003**, 47-57
127. Kirsch Pinheiro, M. & Souveyet, C., "Is Group-Awareness Context-Awareness?", In: Rodrigues, A.; Fonseca, B. & Preguiça, N. (Eds.), *24th International Conference on Collaboration and Technology (CRIWG 2018)*, *LNCS 11001*, Springer, **2018**, 198-206
128. Kirsch Pinheiro, M. & Souveyet, C., "Is Group-Awareness Context-Awareness?: Converging Context-Awareness and Group Awareness Support", *International Journal of e-Collaboration*

- (IJeC), Special Issue of Revised and Extended Papers from the 24th International Conference on Collaboration and Technology, 15(3), **2019**, p. 19. DOI: 10.4018/IJeC.2019070101
129. Kirsch Pinheiro, M.; Le Grand, B.; Souveyet, C. & Najar, S., « Espace de Services : Vers une formalisation des Systèmes d'Information Pervasifs », *XXXIème Congrès INFORSID 2013 : Informatique des Organisations et Systèmes d'Information et de Décision*, **2013**, 215-223
 130. Kirsch-Pinheiro, M. & Souveyet, C. "Supporting context on software applications: a survey on context engineering" (« Le support applicatif à la notion de contexte : revue de la littérature en ingénierie de contexte »), *Modélisation et utilisation du contexte*, 2(1), **2018**, ISTE OpenScience. Disponible sur : <https://www.openscience.fr/Le-support-applicatif-a-la-notion-de-contexte-revue-de-la-litterature-en/> (Dernière visite : août 2020)
 131. Kirsch-Pinheiro, M. & Souveyet, C., « Le Rôle des Ressources dans l'Évolution des Systèmes d'Information », *Actes du XXXVIIème Congrès INFORSID (INFORSID 2019)*, Paris, France, Juin 11-14 **2019**, 85-97
 132. Kirsch-Pinheiro, M., Mazo, R., Souveyet, C. & Sprovieri, D., "Requirements Analysis for Context-oriented Systems", *7th Int. Conf. on Ambient Systems, Networks and Technologies (ANT 2016)*, *Procedia Computer Science*, 83, **2016**, 253-261
 133. Kirsch-Pinheiro, M. & Souveyet, C., "Quality on Context Engineering". In: Brézillon P., Turner R., Penco C. (Eds.) *Modeling and Using Context. CONTEXT 2017. Lecture Notes in Computer Science*, 10257, **2017**, 432-439
 134. Kirsch-Pinheiro, M.; Vanrompay, Y.; Victor, K.; Berbers, Y.; Valla, M.; Frà, C.; Mamelli, A.; Barone, P.; Hu, X.; Devlic, A. & Panagiotou, G., "Context Grouping Mechanism for Context Distribution in Ubiquitous Environments", In: Robert Meersman, Zahir Tari et al. (Eds.), *10th International Symposium on Distributed Objects, Middleware, and Applications (DOA'08), OTM 2008 Conferences, Lecture Notes in Computer Science*, 5331, **2008**, 571-588
 135. Kirsch-Pinheiro, M.; Vanrompay, Y. & Berbers, Y., "Context-aware service selection using graph matching". In: Paoli, F. D.; Toma, I.; Maurino, A.; Tilly, M. & Dobson, G. (Eds.), *2nd Non Functional Properties and Service Level Agreements in Service Oriented Computing Workshop (NFPSLA-SOC'08), at ECOWS 2008*, CEUR Workshop proceedings, 411, **2008**. Disponible sur : <http://ceur-ws.org/Vol-411/> (Dernière visite : août 2020)
 136. Kirsch-Pinheiro, M. & Rychkova, I., "Dynamic Context Modeling for Agile Case Management", In: Y.T. Demey & H. Panetto (Eds.), *OTM 2013 Workshops, Lecture Notes in Computer Science*, 8186, Springer-Verlag, **2013**, 144–154
 137. Kirsch-Pinheiro, M.; Gensel, J. & Martin, "Representing Context for an Adaptative Awareness Mechanism". In: H. Vreede, G.-J.; Guerrero, L. & Marin Raventós, G. (Eds.), *Groupware: Design, Implementation, and Use, X International Workshop on Groupware (CRIWG 2004), Lecture Notes in Computer Science*, 3198, Springer Berlin Heidelberg, **2004**, 339-348
 138. Kirsch-Pinheiro, M.; Gensel, J. & Martin, H., "Awareness on Mobile Groupware Systems". In: Ahmed Karmouch, Larry Korba, Edmundo Roberto Mauro Madeira (Eds.), *1st International Workshop on Mobility Aware Technologies and Applications (MATA 2004), Lecture Notes in Computer Science*, 3284, **2004**, 78-87
 139. Kirsch-Pinheiro, M.; Villanova-Oliver, M.; Gensel, J. & Martin, H., "Context-aware filtering for collaborative web systems: adapting the awareness information to the user's context". In: Hisham Haddad, Lorie M. Liebrock, Andrea Omicini, Roger L. Wainwright (Eds.), *Proceedings of the 2005 ACM Symposium on Applied Computing (SAC 2005)*, **2005**, 1668-1673
 140. Kirsch-Pinheiro, M.; Villanova-Oliver, M.; Gensel, J. & Martin, H., "BW-M: a framework for awareness support in Web-based groupware systems". In: Weiming Shen, Anne E. James, Kuo-Ming Chao, Muhammad Younas, Zongkai Lin, Jean-Paul A. Barthès (Eds.), *Proceedings of the 9th International Conference on Computer Supported Cooperative Work in Design (CSCWD 2005)*, Volume 1, **2005**, 240-246
 141. Kirsch-Pinheiro, M.; Villanova-Oliver, M.; Gensel, J. & Martin, H., "A Personalized and Context-Aware Adaptation Process for Web-Based Groupware Systems". In: Moira C. Norrie, Schahram Dustdar, Harald Gall (Eds.), *Proceedings of the CAISE 06, Workshop on Ubiquitous Mobile*

- Information and Collaboration Systems* (UMICS 2006), **2006**, 884-898. Disponible sur <http://ceur-ws.org/Vol-242/> (dernière visite : août 2020).
142. Kirsch Pinheiro, M.; Villanova-Oliver, M.; Gensel, J.; Berbers, Y. & Martin, H., "Personalizing Web-Based Information Systems through Context-Aware User Profiles", *International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (Ubicomm 2008)*, **2008**, 231-238. DOI: 10.1109/UBICOMM.2008.18
 143. Knoll, S.W. & Lukosch, S.G., "Context and Collaborative Work: A Context-Sensitive Intervention Approach for Collaboration in Dynamic Environment. In: Brézillon, P., Gonzalez, A.J. (Eds.), *Context in Computing*, Chapter 21, **2014**, 327-341
 144. König, I.; Voigtmann, C.; Klein, B. & David, K., "Enhancing Alignment Based Context Prediction by Using Multiple Context Sources: Experiment and Analysis". In: Beigl, M.; Christiansen, H.; Roth-Berghofer, T.; Kofod-Petersen, A.; Coventry, K. & Schmidtke, H. (Eds.), *Modeling and Using Context*, LNCS 6967, **2011**, 159-172
 145. Kornysheva, E.; Deneckère, R. & Claudepierre, B., "Towards Method Component Contextualization", *IJISMD*, **2**, **2011**, 49–81.
 146. Kourouthanassis, P. E. & Giaglis, G. M., "A Design Theory for Pervasive Information Systems". *3rd Int. Workshop on Ubiquitous Computing (IWUC 2006), in conjunction with ICEIS 2006*, May **2006**, Paphos, Cyprus, 62-70. DOI: 10.5220/0002503700620070. Disponible sur : <https://www.scitepress.org/Papers/2006/25037/25037.pdf> (Dernière visite : Aout 2020)
 147. Kourouthanassis, P. E.; Giaglis, G. M. & Karaiskos, D. C., "Delineating 'pervasiveness' in pervasive information systems: a taxonomical framework and design implications". *Journal of Information Technology*, **25**(3), **2010**, 273-287
 148. Krogstie, J.; Lyytinen, K.; Opdahl, A.L.; Pernici, B., Siau, K. & Smolander, K., "Research areas and challenges for mobile information systems". *International Journal of Mobile Communications*, **2**(3), September **2004**, 220–234. DOI: 10.1504/IJMC.2004.005161
 149. Kumar, K.A. ; Konishetty, V.K. ; Voruganti, K. & Rao, G.V.P. "CASH: context aware scheduler for Hadoop", *Proceedings of the International Conference on Advances in Computing, Communications and Informatics (ICACCI '12)*, New York, NY, USA, ISBN : 978-1-4503-1196-0, **2012**, 52–61.
 150. Lemlouma T. & Layaida N., "Context-Aware Adaptation for Mobile Devices", *IEEE International Conference on Mobile Data Management*, **2004**, IEEE Computer Society, 106-111
 151. Liu, Z.; Yang, X.; Yang, Y.; Wang, K. & Mao, G., "DATS: Dispersive Stable Task Scheduling in Heterogeneous Fog Networks", *IEEE Internet of Things Journal*, **6**(2), **2019**, 3423-3436. DOI: [10.1109/JIOT.2018.2884720](https://doi.org/10.1109/JIOT.2018.2884720)
 152. Lopez Peña, M. A. & Muñoz Fernández, I., "SAT-IoT: An Architectural Model for a High-Performance Fog/Edge/Cloud IoT Platform", *IEEE 5th World Forum on Internet of Things (WF-IoT)*, **2019**, 633-638
 153. Lork, C.; Rajasekhar, B.; Yuen, C. & Pindoriya, N. M., "How many watts: A data driven approach to aggregated residential air-conditioning load forecasting", *13th Workshop on Context and Activity Modeling and Recognition (CoMoRea 2017), IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, **2017**, 285-290
 154. Lu, Y., "Industry 4.0: A survey on technologies, applications and open research issues", *Journal of Industrial Information Integration*, **6**, June **2017**, 1 – 10
 155. Maass, W. & Varshney, U., "Design and evaluation of Ubiquitous Information Systems and use in healthcare", *Decision Support Systems*, **54**(1), **2012**, 597-609. DOI: 10.1016/j.dss.2012.08.007
 156. Machine Learning Mastery. Disponible sur : <https://machinelearningmastery.com/start-here/> (Dernière visite : 31 juillet 2018).
 157. MacKenzie, C.M. ; Laskey, K. ; McCabe, F. ; Brown, P.F. & Metz, J., "Reference Model for Service Oriented Architecture 1.0". OASIS Standard, **2006**. Disponible sur <http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.html> (Dernière visite : juillet 2020).
 158. Marie, P.; Desprats, T.; Chabridon, S. & Sibilla, M., "The QoCIM Framework: Concepts and Tools for Quality of Context Management". In: Brézillon, P. & Gonzalez, A. J. (Eds.), *Context in Computing: A Cross-Disciplinary Approach for Modeling the Real World*, Springer New York, p.155-172, **2014**.

159. Martin, D. (Ed.), "OWL-S: Semantic Markup for Web Services", W3C Member Submission 22. November **2004**, <http://www.w3.org/Submission/OWL-S> (Dernière visite : juillet 2020).
160. Martinez-Carreras, M. A.; Muñoz, A. & Botia, J., "Building and evaluating context-aware collaborative working environments", *Information Sciences*, **235**, **2013**, 224 - 241
161. Mayrhofer, R.; Harald, R. & Alois, F., "Recognizing and predicting context by learning from user behavior". In: W. Schreiner, G. Kotsis, A. Ferscha, & K. Ibrahim (Eds.), *Int. Conf. on Advances in Mobile Multimedia (MoMM2003)*, **2003**, 25–35
162. Mendonça, S.; Pina e Cunha, M.; Kaivo-Oja, J. & Ruff, F., "Wild Cards, Weak Signals and Organizational Improvisation". FEUNL Working Paper No. 432, 2003. Available at SSRN: <https://ssrn.com/abstract=882123>
163. Miorandi, D.; Sicari, S.; Pellegrini, F. D. & Chlamtac, I., "Internet of things: Vision, applications and research challenges", *Ad Hoc Networks*, vol. 10 issue 7, p. 1497-1516, 2012.
164. Mirbel, I. & Crescenzo, P., "From End-User's Requirements to Web Services Retrieval: A Semantic and Intention-Driven Approach". In: J.-H. Morin, J. Ralyté & M. Snene (eds.), *Exploring Services Science*, Springer Berlin Heidelberg, **2010**, 30–44
165. Molano, J.I.R.; Betancourt, D. & Gómez, G., "Internet of things: A prototype architecture using a Raspberry Pi", In: Uden, L.; Heričko, M. & Ting, I.H. (Eds.), *Knowledge Management in Organizations (KMO 2015)*, *Lecture Notes in Business Information Processing*, 224, Springer, **2015**, 618–631
166. Montero, R. S.; Rojas, E.; Carrillo, A. A. & Llorente, I. M. "Extending the Cloud to the Network Edge", *IEEE Computer*, 50(4), April **2017**, 91-95.
167. Moran T. & Dourish, P., "Introduction to this special issue on context-aware computing", *Human-Computer Interaction*, 16 (2-3), **2001**, 87–95.
168. Mulfari, D.; Fazio, M.; Celesti, A.; Villari, M. & Puliafito, A., "Design of an IoT cloud system for container virtualization on smart objects". *3rd Workshop on Cloud for IoT (CLIoT 2015)*, *European Conference on Service-Oriented and Cloud Computing (ESOCC 2015)*, *Communications in Computer and Information Science (CCIS)*, vol. 567, **2015**, Springer, 33-47
169. Najar, S.; Kirsch-Pinheiro, M. & Souveyet, C., "Enriched Semantic Service Description for Service Discovery: Bringing Context to Intentional Services", *International Journal On Advances in Intelligent Systems*, volume 5, numbers 1 & 2, June **2012**, 159-174, IARIA Journals / ThinkMind, ISSN: 1942-2679
170. Najar, S., « Adaptation des services sensibles au contexte selon une approche intentionnelle », Phd Thesis, Université Paris 1 Panthéon Sorbonne, Paris, France, **2014**
171. Najar, S.; Saidani, O.; Kirsch-Pinheiro, M.; Souveyet, C. & Nurcan, S., "Semantic representation of context models: a framework for analyzing and understanding" In: J. M. Gomez-Perez, P. Haase, M. Tilly, and P. Warren (Eds), *Proceedings of the 1st Workshop on Context, information and ontologies (CIAO 09)*, *European Semantic Web Conference (ESWC'2009)*, **2009**, 1-10
172. Najar, S.; Kirsch Pinheiro, M. & Souveyet, C., "The influence of context on intentional service". *5th Int. IEEE Workshop on Requirements Engineerings for Services (REFS'11)*, *IEEE Conference on Computers, Software, and Applications (COMPSAC'11)*, Munich, Germany, 470-475, **2011**
173. Najar, S.; Kirsch Pinheiro, M. & Souveyet, C., "Bringing context to intentional services". *3rd Int. Conference on Advanced Service Computing, Service Computation'11*, Rome, Italy, **2011**, 118-123. Best Paper Awards
174. Najar, S.; Kirsch Pinheiro, M. & Souveyet, C., "Towards Semantic Modeling of intentional pervasive System", *6th International Workshop on Enhanced Web Service Technologies (WEWST'11)*, *European Conference on Web Services (ECOWS'11)*, Lugano, Switzerland, **2011**, 30-34
175. Najar, S.; Kirsch Pinheiro, M.; Le Grand, B. & Souveyet, C., "A user-centric vision of service-oriented Pervasive Information Systems", *8th International Conference on Research Challenges in Information Science (RCIS 2014)*, IEEE, **2014**, 359-370
176. Najar, S.; Kirsch Pinheiro, M.; Le Grand, B. & Souveyet, C., « Systèmes d'Information Pervasifs et Espaces de Services : Définition d'un cadre conceptuel ». *UbiMob 2013 : 9èmes journées*

francophones Mobilité et Ubiquité, Jun **2013**, Nancy, France. Disponible sur <https://ubimob2013.sciencesconf.org/19119.html> (Dernière visite : août 2020)

177. Najar, S.; Kirsch-Pinheiro, M. & Souveyet, C. "Service discovery and prediction on Pervasive Information System", *Journal of Ambient Intelligence and Humanized Computing*, vol. 6, issue 4, June **2015**, Springer, 407-423. ISSN: 1868-5137. DOI: [10.1007/s12652-015-0288-5](https://doi.org/10.1007/s12652-015-0288-5)
178. Najar, S.; Kirsch-Pinheiro, M. & Souveyet, C., "A context-aware intentional service prediction mechanism in PIS", In: David De Roure, Bhavani Thuraisingham & Jia Zhang (Eds.), *IEEE 21st International Conference on Web Services (ICWS 2014)*, 27 June - 2 July **2014**, Anchorage, Alaska, USA, IEEE CS, 662-669. DOI : 10.1109/ICWS.2014.97
179. Najar, S. ; Kirsch-Pinheiro, M. & Souveyet, C., "A new approach for service discovery and prediction on Pervasive Information System", *5th International Conference on Ambient Systems, Networks and Technologies (ANT-2014)*, *Procedia Computer Science*, v32, **2014**, Elsevier, 421–428. DOI : <http://dx.doi.org/10.1016/j.procs.2014.05.443>
180. Najar, S.; Kirsch Pinheiro, M.; Vanrompay, Y.; Steffemel, L.A. & Souveyet, C., "Intention Prediction Mechanism in an Intentional Pervasive Information System", In : Kolomvatsos, K., Anagnostopoulos, C., Hadjiefthymiades, S. (Eds.), *Intelligent Technologies and Techniques for Pervasive Computing*, IGI Global, **2013**, pp. 251-275. DOI: 10.4018/978-1-4666-4038-2.ch014, ISBN : 978-1-4666-4040-5
181. Najar, S.; Kirsch-Pinheiro, M.; Souveyet, C. & Steffemel, L. A., "Service Discovery Mechanisms for an Intentional Pervasive Information System". *Proceedings of 19th IEEE International Conference on Web Services (ICWS 2012)*, Honolulu, Hawaii, 24-29 June **2012**, 520-527
182. Najar, S. ; Kirsch-Pinheiro, M. ; Steffemel, L. A. & Souveyet, C., « Analyse des mécanismes de découverte de services avec prise en charge du contexte et de l'intention ». In : Philippe Roose & Nadine Rouillon-Couture (dir.), *8èmes Journées Francophones Mobilité et Ubiquité (Ubimob 2012)*, June 4-6, **2012**, Anglet, France. Cépaduès Editions, 210-221. ISBN 978.2.36493.018.6
183. Najar, S. ; Kirsch-Pinheiro, M. & Souveyet, C., « Mécanisme de prédiction dans un système d'information pervasif et intentionnel », In : Philippe Roose & Nadine Rouillon-Couture (dir.), *8èmes Journées Francophones Mobilité et Ubiquité (Ubimob 2012)*, June 4-6, **2012**, Anglet, France. Cépaduès Editions, pp. 146-157. ISBN: 978.2.36493.018.6
184. Neumann, G.; Sobernig, S. & Aram, M., "Evolutionary Business Information Systems", *Business & Information Systems Engineering*, 6, **2014**, 33-38. DOI: 10.1007/s12599-013-0305-1
185. Nunes, V.T.; Santoro, F.M. & Borges, M., "Capturing Context about Group Design Processes", *Proceedings of the 11th Int. Conf. on Computer Supported Cooperative Work in Design, CSCWD 2007*, April 26-28, Melbourne, Australia, **2007**
186. O'Leary, D. E., "Embedding AI and Crowdsourcing in the Big Data Lake", *IEEE Intelligent Systems*, 29(5), Sept.-Oct. **2014**, 70-73. DOI: 10.1109/MIS.2014.82
187. Olaniyan, R.; Fadahunsi, O.; Maheswaran, M. & Zhani, M.F. "Opportunistic edge computing: Concepts, opportunities and research challenges", *Future Generation Computer Systems*, 89, **2018**, 633 – 645.
188. Paolucci, M.; Kawamura, T.; Payne, T.R. & Sycara, K., "Semantic Matching of Web Services Capabilities", In: I. Horrocks & J. Hendler (eds.), *The Semantic Web – ISWC 2002*, Lecture Notes in Computer Science, vol. 2342, Springer, **2002**, 333–347
189. Papazoglou, M.P. & Georgakopoulos, D., "Introduction: Service-oriented computing". *Communication of the ACM*, 46(10), **2003**, 24–28
190. Papazoglou, M.P., "Service-oriented computing: concepts, characteristics and directions". In *Proceedings of the Fourth International Conference on Web Information Systems Engineering (WISE 2003)*, **2003**, 3–12
191. Papazoglou, M.P.; Traverso, P.; Dustdar, S. & Leymann, F., "Service-Oriented Computing: A Research Roadmap". *Int. Journal of Cooperative Information Systems*, 17(2), **2008**, 223-255
192. Parashar, M. & Pierson, J.-M., "Pervasive grids: Challenges and opportunities", In: K. Li, C. Hsu, L. Yang, J. Dongarra & H. Zima (Eds.), *Handbook of Research on Scalable Computing Technologies*, IGI Global, **2010**, 14–30.

193. Paridel, K.; Yasar, A.; Vanrompay, Y.; Preuveneers, D. & Berbers, Y., "Teamwork on the road: Efficient collaboration in VANETs with context-based grouping", *2nd International Conference on Ambient Systems, Networks and Technologies (ANT-2011)*, *Procedia Computer Science*, 5, **2011**, Elsevier, 48-57.
194. Paspallis, N.; Rouvoy, R.; Barone, P.; Papadopoulos, G.A.; Eliassen, F. & Mamelli, A., "A Pluggable and Reconfigurable Architecture for a Context-Aware Enabling Middleware System". In: Meersman, R. & Tari, Z. (Eds.), *On the Move to Meaningful Internet Systems: Confederated International Conferences, CoopIS, DOA, GADA, IS, and ODBASE 2008*, *Lecture Notes in Computer Science*, 5331, **2008**, Springer, 553–570
195. Perera, C.; Zaslavsky, A.B.; Christen, P. & Georgakopoulos, D., "Context Aware Computing for The Internet of Things: A Survey". *IEEE Communications Surveys & Tutorials*, 16 (1), **2014**, 414-454
196. Petit, M.; Ray, C. & Claramunt, C., « Algorithme de recommandation adaptable pour la personnalisation d'un système mobile », *6^{èmes} Journées Francophones Ubiquité et Mobilité (UBIMOB 2010)*, **2010**, 59-62
197. Ploesser, K.; Recker, J. & Rosemann, M., "Supporting Context-Aware Process Design: Learnings from a Design Science Study", In: zur Muehlen, M. & Su, J. (Eds.), *Business Process Management Workshops (BPM 2010)*, *Lecture Notes in Business Information Processing*, 66., Springer, **2010**, 97-104
198. Prat, N., "Goal Formalisation and Classification for Requirements Engineering", *Proceedings of the third International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ)*, **1997**, 145-156
199. Preuveneers, D.; Naqvi, N.Z.; Ramakrishnan, A.; Berbers, Y. & Joosen, W., "Adaptive Dissemination for Mobile Electronic Health Record Applications with Proactive Situational Awareness". *49th Hawaii Int. Conf. on System Sciences (HICSS 2016)*, **2016**, 3229-3238
200. Preuveneers, D. & Berbers, Y., "Context-driven migration and diffusion of pervasive services on the OSGi framework". *Int. J. Auton. Adapt. Commun. Syst.*, 3, **2010**, 3–22.
201. Preuveneers, D.; Bergh, J.V.D.; Wagelaar, D.; Georges, A.; Rigole, P.; Clerckx, T.; Berbers, E.; Coninx, K. & Bosschere, K.D., "Towards an extensible context ontology for ambient intelligence". In: Markopoulos, P.; Eggen, B.; Aarts, E. & Crowley, J. (Eds.), *Proceedings of the Second European Symposium on Ambient Intelligence (EUSAI 2004)*, *LNCS 3295*, Springer-Verlag, **2004**, 148–159
202. Preuveneers, D.; Victor, K.; Vanrompay, Y.; Rigole, P.; Kirsch Pinheiro, M. & Berbers, Y. "Context-Aware Adaptation in an Ecology of Applications". In: Dragan Stojanovic (Ed.), *Context-Aware Mobile and Ubiquitous Computing for Enhanced Usability: Adaptive Technologies and Applications*, IGI Global, **2009**.
203. Priss, U., "Formal Concept Analysis in Information Science". In: Blaise, C. (eds.), *Annual Review of Information Science and Technology*, ASIST, vol. 40, **2006**, 521-543
204. Puglisi, S.; Moreira, Á. T.; Torregrosa, G. M.; Igartua, M. A. & Forné, J., "MobilitApp: Analysing Mobility Data of Citizens in the Metropolitan Area of Barcelona", In: Mandler, B.; Marquez-Barja, J.; Mitre Campista, M. E.; Cagáñová, D.; Chaouchi, H.; Zeadally, S.; Badra, M.; Giordano, S.; Fazio, M.; Somov, A. & Vieriu, R.-L. (Eds.), *Internet of Things: IoT Infrastructures, 2nd Int. Summit, IoT 360^o 2015, Part I*, Springer, **2016**, 245-250
205. Ramakrishnan, A.; Preuveneers, D. & Berbers, Y., "Enabling self-learning in dynamic and open IoT environments", In: Shakshuki, E. & Yasar, A. (Eds.), *The 5th International Conference on Ambient Systems, Networks and Technologies (ANT-2014)*, *Procedia Computer Science*, 32, **2014**, Elsevier, 207-214
206. Ramakrishnan, A.K., "Support for Data-driven Context Awareness in Smart Mobile and IoT Applications: Resource Efficient Probabilistic Models and a Quality-aware Middleware Architecture" (Ondersteuning voor data-gedreven context-bewustzijn in intelligente mobiele en IoT applicaties: Hulpbronnenefficiënte probabilistische modellen en een kwaliteit-aware middleware architectuur), PhD thesis, Katholieke Universiteit Leuven, Belgium **2016**

207. Ramírez-Gallego, S.; Krawczyk, B.; García, S.; Woźniak M. & Herrera, F., "A survey on data preprocessing for data stream mining: Current status and future directions", *Neurocomputing*, 239, 24 May **2017**, 39-57
208. Rao, B. & Angelov, B., "Pervasive Information Systems Value Chain: A Services Perspective", *International Journal of Innovation and Technology Management*, 06(01), **2009**, 17-40
209. Reichle, R. ; Wagner, M. ; Khan, M.U. ; Geihs, K.; Lorenzo, J. ; Valla, M. ; Fra, C. ; Paspallis, N. & Papadopoulos, G.A., "A Comprehensive Context Modeling Framework for Pervasive Computing Systems". In: R. Meier, and S. Terzis (Eds.), *Distributed Applications and Interoperable Systems (DAIS 2008)*, LNCS 5053, **2008**, 281–295
210. Reichle, R.; Wagner, M.; Khan, M.U.; Geihs, K.; Valla, M.; Frà, C.; Paspallis, N. & Papadopoulos, G.A., "A Context Query Language for Pervasive Computing Environments", *6th Annual IEEE Int. Conference on Pervasive Computing and Communications (PerCom)*, **2008**, 434-440
211. Rychkova, I. ; Kirsch Pinheiro M. & Le Grand B. "Context-Aware Agile Business Process Engine: Foundations and Architecture", In : Nurcan, S., Proper, H., Soffer, P., Krogstie, J., Schmidt, R., Halpin, T. & Bider, I. (Eds.), *Enterprise, Business-Process and Information Systems Modeling, Proceedings of the 14th Working Conference on Business Process Modeling, Development, and Support (BPMDS 2013)*, Lecture Notes in Business Information Processing, 146, Valence, Spain, **2013**, 32-47
212. Rychkova I. ; Kirsch-Pinheiro M. & Le Grand B., "Automated Guidance for Case Management: Science or Fiction?", In : Ficher, L. (Ed.), *Empowering Knowledge Workers: New Ways to Leverage Case Management*, Series BPM and Workflow Handbook Series, Future Strategies Inc., **2014**, 67-78. ISBN : 978-0-984976478
213. Rolland, C., "Capturing System Intentionality with Maps". In: J. Krogstie, A.L. Opdahl & S. Brinkkemper (Eds.), *Conceptual Modelling in Information Systems Engineering*, Springer Berlin Heidelberg, **2007**, 141–158
214. Rolland, C.; Kirsch-Pinheiro, M. & Souveyet, C., "An Intentional Approach to Service Engineering", *IEEE Transactions on Services Computing*, 3(4), Oct.-Dec. **2010**, 292-305
215. Römer, K. & Mattern, F., "Adaptation without anticipation", In : Ferscha, A., *Pervasive Adaptation: Next generation pervasive computing research agenda*, 28-29, **2011**. Disponible sur <https://www.pervasive.jku.at/Conferences/fet11/RAB.pdf> (dernière visite: août 2020)
216. Rosa, M. G. P.; Borges, M. R. S. & Santoro, F. M., "Evaluation of Contextual Information Influence on Group Interaction", Shen, W.; Luo, J.; Lin, Z.; Barthès, J.-P. A. & Hao, Q. (Eds.), *Computer Supported Cooperative Work in Design III*, LNCS 4402, Springer, **2007**, 13-22
217. Rottenberg, S.; Leriche, S.; Taconet, C.; Lecocq, C. & Desprats, T. "MuSCa: A multiscale characterization framework for complex distributed systems", *2014 Federated Conference on Computer Science and Information Systems*, **2014**, 1657-1665
218. Rouvoy, R.; Barone, P.; Ding, Y.; Eliassen, F.; Hallsteinsen, S.; Lorenzo, J.; Mamelli, A. & Scholz, U., "MUSIC: Middleware Support for Self-Adaptation in Ubiquitous and Service-Oriented Environments". In: Cheng, B.; de Lemos, R.; Giese, H.; Inverardi, P. & Magee, J. (Eds.), *Software Engineering for Self-Adaptive Systems*, Lecture Notes in Computer Science, 5525, Springer, **2009**, 164-182
219. Saidani, O.; Rolland, C. & Nurcan, S., "Towards a Generic Context Model for BPM". *48th Annual Hawaii International Conference on System Sciences (HICSS'2015)*, Jan **2015**
220. Salman, O.; Elhadj, I.; Kayssi, A. & Chehab, A. "Edge computing enabling the Internet of Things," *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, Milan, **2015**, 603-608.
221. Santos, L.O.B. da S.; Guizzardi, G.; Pires, L.F. & Sinderen, M., "From User Goals to Service Discovery and Composition". In: Heuser C.A. & Pernul G. (Eds), *Advances in Conceptual Modeling - Challenging Perspectives (ER 2009)*, Lecture Notes in Computer Science, 5833, **2009**, 265-274
222. Satyanarayanan, M.; Bahl, P.; Caceres, R. & Davies, N. "The case for VM-based cloudlets in mobile computing". *IEEE Pervasive Computing*, 8(4), **2009**, 14–23.
223. Sawyer, P., Mazo, R., Diaz, D., Salinesi, C., Hughes, D., "Using Constraint Programming to Manage Configurations in Self-Adaptive Systems". *Computer*, 45(10), **2012**, 56-63

224. Schadt, E.E.; Linderman, M.D.; Sorenson, J.; Lee, L. & Nolan, G.P., "Computational solutions to large-scale data management and analysis". *Nature Reviews Genetics*, 11(9), Sept. **2010**, 647–657
225. Schilit, B.N. & Theimer, M.M., "Disseminating Active Map Information to Mobile Hosts", *Network*, 8, IEEE, **1994**, 22-32
226. Schmidt, K., "The problem with 'awareness': introductory remarks on 'Awareness in CSCW'", *Computer Supported Cooperative Work*, 11(3-4), sept. **2002**, Kluwer Academic Publishers, 285-298
227. Schreiber, F. A.; Camplani, R.; Fortunato, M.; Marelli, M. & Rota, G., "PerLa: A Language and Middleware Architecture for Data Management and Integration in Pervasive Information Systems", *IEEE Transactions on Software Engineering*, 38 (2), March-April **2012**, 478-496, DOI: 10.1109/TSE.2011.25
228. Shekhar, S. & Gokhale, A., "Dynamic Resource Management Across Cloud-Edge Resources for Performance-Sensitive Applications", *17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*, **2017**, 707-710
229. Shoaib, M.; Bosch, S.; Incel, O.D.; Scholten, H. & Havinga, P. J.M. "A survey of online activity recognition using mobile phones". *Sensors*, 15(1), **2015**, 2059–2085
230. Sigg, S.; Haseloff, S. & David, K. "An Alignment Approach for Context Prediction Tasks in UbiComp Environments". *IEEE Pervasive Computing*, vol. 9, issue 4, **2010**, 90–97
231. Singh, S. & Chana, I., "A Survey on Resource Scheduling in Cloud Computing: Issues and Challenges", *Journal of Grid Computing*, 14(2), **2016**, 217-264
232. Souveyet, C.; Villari, M.; Steffanel, L. A. & Kirsch-Pinheiro, M., « Une approche basée sur les MicroÉléments pour l'Évolution des Systèmes d'Information », *Atelier Évolution des SI : vers des SI Pervasifs ?*, *INFORSID 2019*, **2019**. Disponible sur https://evolution-si.sciencesconf.org/data/book_evolution_si_fr.pdf (Dernière visite : août 2020)
233. Steffanel, L. & Kirsch-Pinheiro, M. "CloudFIT, a PaaS platform for IoT applications over Pervasive Networks", In: Celesti A., Leitner P. (Eds). *3rd Workshop on Cloud for IoT (CLIoT 2015)*. Advances in Service-Oriented and Cloud Computing (ESOCC 2015). Communications in Computer and Information Science, 567, **2015**, 20-32.
234. Steffanel, L. A.; Flauzac, O.; Charao, A. S.; Barcelos, P.; Stein, B.; Cassales, G.; Nesmachnow, S.; Rey, J.; Cogorno, M.; Kirsch-Pinheiro, M. & Souveyet, C., "Mapreduce challenges on pervasive grids", *Journal of Computer Science*, 10(11), July **2014**, 2194-2210.
235. Steffanel, L. A.; Flauzac, O.; Charao, A. S.; Barcelos, P. P.; Stein, B.; Nesmachnow, S.; Kirsch Pinheiro, M. & Diaz, D., "PER-MARE: Adaptive Deployment of MapReduce over Pervasive Grids", *8th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC'13)*, **2013**, 17-24.
236. Steffanel, L.A. & Kirsch Pinheiro, M. "When the cloud goes pervasive: approaches for IoT PaaS on a ubiquitous world". In: Mandler B. et al. (Eds), *EAI International Conference on Cloud, Networking for IoT systems (CN4IoT 2015)*, *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (LNICST)*, 169, **2015**, 347–356.
237. Steffanel, L.A. & Kirsch-Pinheiro, M., "Leveraging Data Intensive Applications on a Pervasive Computing Platform: the case of MapReduce", *1st Workshop on Big Data and Data Mining Challenges on IoT and Pervasive (Big2DM)*, London, UK, June 2 - 5, 2015. *Procedia Computer Science*, 52, Jun **2015**, Elsevier, 1034–1039. doi: 10.1016/j.procs.2015.05.102.
238. Steffanel, L.A. & Kirsch-Pinheiro, M., « Accès aux Données dans le Fog Computing : le cas des dispositifs de proximité », *Conférence d'informatique en Parallélisme, Architecture et Système (CompAS'19)*, 25-28 July **2019**, Anglet, France. Disponible sur <https://hal.univ-reims.fr/hal-02174708> (Dernière visite : août 2020)
239. Steffanel, L.A. & Kirsch-Pinheiro, M., "Improving Data Locality in P2P-based Fog Computing Platforms", *9th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2018)*, *Procedia Computer Science*, 141, Elsevier, **2018**, 72-79. DOI : 10.1016/j.procs.2018.10.151

240. Steffemel, L.A.; Kirsch-Pinheiro, M.; Vaz Peres, L. & Kirsch Pinheiro, D. "Strategies to implement Edge Computing in a P2P Pervasive Grid", *International Journal of Information Technologies and Systems Approach (IJITSA)*, IGI Global, 11(1), **2018**, 1-15. DOI: doi:10.4018/IJITSA/2018010101
241. Steffemel, L.A. & Kirsch-Pinheiro, M., « Stratégies Multi-Échelle pour les Environnements Pervasifs et l'Internet des Objets ». *11èmes Journées Francophones Mobilité et Ubiquité (Ubimob 2016)*, 5 juillet **2016**, Lorient, France. Paper n°6. Disponible sur https://ubimob2016.telecom-sudparis.eu/files/2016/07/Ubimob_2016_paper_6.pdf (Dernière visite: aout 2020)
242. Streit, A., "On Job Scheduling for HPC-Clusters and the dynP Scheduler", In: Monien, B.; Prasanna, V.K. & Vajapeyam, S. (Eds.), *High Performance Computing - HiPC 2001, Lecture Notes in Computer Science*, 2228, **2001**, Springer, 58-67
243. Sundmaeker, H.; Guillemin, P.; Friess, P. & Woelfflé, S., "Vision and Challenges for realizing the internet of things". *Cluster of European Research projects on the Internet of Things (CERP-IoT)*, **2010**. DOI: 10. 2759/26127. Disponible sur : http://www.theinternetofthings.eu/sites/default/files/Rob%20van%20Kranenburg/Clusterbook%202009_0.pdf (Dernière visite : novembre 2014)
244. Suraci, V.; Mignanti, S. & Aiuto, A., "Context-aware Semantic Service Discovery", *16th IST Mobile and Wireless Communications Summit*, **2007**, 1-5
245. Sutterer, M.; Droegehorn, O. & David, K., "UPOS: User Profile Ontology with Situation-Dependent Preferences Support", *First International Conference on Advances in Computer-Human Interaction - ACHI 2008*, IEEE, **2008**, 230-235
246. Tax, N.; Sidorova, N.; Haakma, R. & van der Aalst, W. M., "Mining local process models", *Journal of Innovation in Digital Ecosystems*, 3(2), **2016**, Elsevier, 183 – 196
247. TechTarget, "Cutting edge: IT's guide to edge data centers", SearchDataCenter, June **2016**. Disponible sur : <https://searchdatacenter.techtarget.com/guide/Cutting-edge-ITs-guide-to-edge-data-centers> (dernière visite : avril 2019).
248. Toninelli, A.; Corradi, A. & Montanari, R., "Semantic-based discovery to support mobile context-aware service access", *Computer Communications*, vol. 31, issue 5, march **2008**, 935-949
249. Trabelsi, L. (2014). Les Systèmes d'Information Urbanisés : étude et analyse de leur performance. Thèse de doctorat en Sciences de Gestion. Université Nice Sophia Antipolis, **2014**
250. Tyagi, R. & Gupta, S.K., "A survey on scheduling algorithms for parallel and distributed systems". In: Mishra, A.; Basu, A. & Tyagi V. (eds), *Silicon Photonics & High Performance Computing, Advances in Intelligent Systems and Computing*, vol. 718, **2018**, 51 – 64. DOI: 10.1007/978-981-10-7656-5_7
251. Tychalas, D. & Karatza, H., "A Scheduling Algorithm for a Fog Computing System with Bag-of-Tasks Jobs: Simulation and Performance Evaluation", *Simulation Modelling Practice and Theory*, 98, **2020**, 101982. <https://doi.org/10.1016/j.simpat.2019.101982>
252. Van Lamsweerde A., "Requirements Engineering in the Year 00: A Research Perspective". *22nd International Conference on Software Engineering (ICSE'2000)*, Limerick, Ireland, **2000**, 5-19
253. Vanrompay, Y., "Efficient Prediction of Future Context for Proactive Smart Systems", PhD thesis, Katholieke Universiteit Leuven, Belgium, **2011**.
254. Vanrompay, Y.; Kirsch Pinheiro, M.; Ben Mustapha, N.; Aufaure, M.-A., "Context-Based Grouping and Recommendation in MANETs", In: Kolomvatsos, K., Anagnostopoulos, C., Hadjiefthymiades, S. (Eds.), *Intelligent Technologies and Techniques for Pervasive Computing*, IGI Global, **2013**, 157-178.
255. Vanrompay, Y.; Kirsch-Pinheiro, M. & Berbers, Y., "Context-Aware Service Selection with Uncertain Context Information", *Context-Aware Adaptation Mechanism for Pervasive and Ubiquitous Services 2009 (CAMPUS 2009)*, *Electronic Communications of the EASST*, vol. 19, **2009**
256. Vanrompay, Y.; Kirsch-Pinheiro, M. & Berbers, Y., "Service Selection with Uncertain Context Information", In: Stephan Reiff-Marganiec and Marcel Tilly (Eds.), *Handbook of Research on Service-Oriented Systems and Non-Functional Properties: Future Directions*, IGI Global, 192-215, **2011**

257. Vanrompay, Y.; Mehlhase, S. & Berbers, Y. "An effective quality measure for prediction of context information", *8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, **2010**, 13 -17
258. Velasquez, K. ; Abreu, D.P. ; Assis, M. *et al.* "Fog orchestration for the Internet of Everything: state-of-the-art and research challenges". *Journal of Internet Services and Applications*, 9, article 14, **2018**. DOI: <https://doi.org/10.1186/s13174-018-0086-3>
259. Verbert, K.; Manouselis, N.; Ochoa, X.; Wolpers, M.; Drachler, H.; Bosnic, I. & Duval, E., "Context-Aware Recommender Systems for Learning: A Survey and Future Challenges", *IEEE Transactions on Learning Technologies*, **2012**, 5(4), 318-335
260. Villalba, I.; Prez, J.L.; Carrera, D.; Pedrinaci, C. & Panziera, L., "Servioticity and iserve: A scalable platform for mining the IoT". *6th Int. Conf. on Ambient Systems, Networks and Technologies (ANT-2015)*, Procedia Computer Science, 52, **2015**, 1022-1027
261. Villari, M.; Fazio, M.; Dustdar, S.; Rana, O. & Ranjan R., "Osmotic Computing: A New Paradigm for Edge/Cloud Integration", *IEEE Cloud Computing*, 3(6), **2016**, 76-83
262. Wagner, M.; Reichle, R.; Khan, M.U.; Geihs, K.; Lorenzo, J.; Valla, M.; Frà, C.; Paspallis, N. & Papadopoulos, G.A., "A Comprehensive Context Modeling Framework for Pervasive Computing Systems", *8th IFIP WG 6.1 International Conference on Distributed Applications and Interoperable Systems (DAIS 2008)*, Lecture Notes on Computer Science, vol. 5053, **2008**, 281-295
263. Wang, W. & Reani, M., "The rise of mobile computing for Group Decision Support Systems: A comparative evaluation of mobile and desktop", *International Journal of Human-Computer Studies*, 104, **2017**, 16-35
264. Weiser, M., "The computer for the 21st century", *Scientific American*, vol. 265, No 3, September **1991**, 94-104
265. White, Tom, "Hadoop: The definitive guide", 2nd edition, O'Reilly, **2010**, ISBN 978-1-449-38973-4, p. 626.
266. Wille, R., "Formal Concept Analysis as Mathematical Theory of Concepts and Concept Hierarchies". In: Ganter, B. *et al.* (Eds.), *Formal Concept Analysis*, Springer-Verlag, **2005**, 1-33
267. Wolf, W.; Jerraya, A. A. & Martin, G., "Multiprocessor system-on-chip (MPSoC) technology", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, **2008**, 27(10), 1701-1713
268. Xia, C. & Tingjie, L., "Intelligent Business Prediction in Context-awareness Services Based on Hidden Markov Model (HMM)", *2nd Pacific-Asia Conference on Web Mining and Web-based Application*, **2009**, 116-119
269. Yu, E.S.K. & Mylopoulos, J., "Why Goal-Oriented Requirements Engineering". *Int. Working Conference on Requirement Engineering: Foundation for Software Quality*, **1998**, 15-22
270. Zaharia, M. ; Konwinski, A. ; Joseph, A.D. ; Katz, R. & Stoica, I., "Improving MapReduce performance in heterogeneous environments", *Proceedings of the 8th USENIX conference on Operating systems design and implementation (OSDI'08)*, USENIX Association, Berkeley, CA, USA, **2008**, 29-42.
271. Zaidenberg, S., « Apprentissage par renforcement de modèles de contexte pour l'informatique ambiante ». PhD Thesis, Institut National Polytechnique de Grenoble - INPG, **2009**
272. Zappatore, M.; Longo, A.; Bochicchio, M. A.; Zappatore, D.; Morrone, A. A. & De Mitri, G., "Towards Urban Mobile Sensing as a Service: An Experience from Southern Italy", In: Mandler, B.; Marquez-Barja, J.; Mitre Campista, M. E.; Cagáňová, D.; Chaouchi, H.; Zeadally, S.; Badra, M.; Giordano, S.; Fazio, M.; Somov, A. & Vieriu, R.-L. (Eds.), *Internet of Things: IoT Infrastructures*, *2nd Int. Summit, IoT 360° 2015, Part I*, Springer, p. 377-387, 2016.

Annexes

Annexe I - *Curriculum Vitae*

CURRICULUM VITAE

Identification

Manuele Kirsch Pinheiro

Maitre de Conférences en Informatique

Centre de Recherches en Informatique (CRI)

École de Management de la Sorbonne

Université Paris 1 – Panthéon Sorbonne



État civil

Nom :	Kirsch Pinheiro
Épouse :	Steffenel
Prénom :	Manuele
Date de Naissance :	18 novembre 1976
Adresse Professionnelle	Université Paris 1 – Panthéon Sorbonne Centre de Recherche en Informatique 90 rue Tolbiac – Bureau C1405 75013 Paris
Téléphone :	+33 (0) 6 62 27 78 83
Email :	Manuele.KirschPinheiro@univ-paris1.fr
Page Web :	http://www.kirschpm.fr

Parcours professionnel

Sept. 2008 – à présent	Maître de conférences à l'Université Paris 1 Panthéon-Sorbonne, Paris, France. Intégrée au <i>Centre de Recherche en Informatique</i> .
Sept. 2007 - Août 2008	Post-doctorante à Katholiek Universiteit Leuven (K.U.Leuven), en Belgique. Intégrée à l'équipe <i>Embedded Systems</i> du groupe DISTRINET (<i>Distributed Systems and Computer Networks</i>), dans le cadre du projet européen MUSIC.
Sept. 2006 - Août 2007	ATER à l' Institut Universitaire Technologique Charlemagne (IUT Nancy-Charlemagne), de l' Université Nancy 2 , Nancy. Service à temps complet effectué au Département Information-Communication (InfoCom). Intégration recherche à l'équipe ECOO, LORIA .

Février – Mai 2005	Vacataire à l' Institut de Géographie Alpine - UJF , Grenoble, France.
Juin 2001 – Juin 2002	Visiteur (stage de recherche) à l' INRIA Rhône-Alpes , à Montbonnot Saint-Martin, France, sous la responsabilité <i>Vincent Quint</i> (Directeur de Recherches INRIA), dans le cadre du projet de coopération franco-brésilien <i>Projet CEMT</i> .
2000	Intervenante (salariée) à Interop Informática , Porto Alegre, Brésil. Activités : support technique et formation HP-UX et Linux.
Sept. – Déc. 1999	Mission d'Études (stage de recherche) au Centre d'Informatique et d'Electronique (NCE) de l' Université Fédérale du Rio de Janeiro (UFRJ) , Rio de Janeiro, Brésil, sous la responsabilité du <i>Marcos R.S. Borges</i> (Professeur UFRJ).
Avril 1998 - Juillet 1999	Enseignante (salariée) au Centre Universitaire Franciscain (UNIFRA), Santa Maria, Brésil. Activités : enseignement au niveau bac+4 (Modélisation Orientée objets, Logique de Première Ordre), encadrements de travaux de fin d'études.

Formations

2002 – 2006	Doctorat (PhD) en Informatique. Thèse de l' Université Joseph Fourier – Grenoble I (UJF) , Grenoble, préparée au sein du Laboratoire Logiciels, Systèmes, Réseaux (LSR) de l' IMAG (Informatique et Mathématiques Appliquées de Grenoble) , sous la direction de Hervé Martin (Professeur, Université Joseph-Fourrier, Grenoble I) et de Jérôme Gensel (Maître de Conférences, Université Pierre Mendès France, Grenoble II).
1998 – 2001	Master (Msc.) en Informatique. Dissertation de l' Université Fédérale du Rio Grande do Sul (UFRGS) , Porto Alegre, Brésil, sous la direction de José Valdeni de Lima (Professeur, UFRGS).
1994 – 1998	Diplômée (Bsc.) en Informatique (équivalent Bac+4), Université Fédérale de Santa Maria (UFSM) , Santa Maria, Brésil.

Encadrements

- David Beserra, **Gestion de ressources sensible au contexte dans les Systèmes d'Information Pervasifs**. Directeur de thèse : Carine SOUVEYET (Centre de Recherche en Informatique, Université Paris 1 Panthéon Sorbonne), co-encadrant : Manuele Kirsch Pinheiro. Thèse démarrée en Septembre 2016. Prévission de soutenance : fin 2020.
 - Taux d'encadrement scientifique estimé : **75%**
 - Travail à la croisée de plusieurs évolutions technologiques ayant fortement impactées les Systèmes d'Information : l'Internet of Things (IoT), l'Edge Computing et le Cloud Computing. A terme, ce travail permettra de proposer des mécanismes pour une gestion opportuniste des ressources dans les Systèmes d'Information Pervasifs.

2. Ali Jaffal, **Aide à l'utilisation et à l'exploitation de l'Analyse de Concepts Formels pour des non-spécialistes de l'analyse des données**. Directeur de thèse : Bénédicte Le Grand (Centre de Recherche en Informatique, Université Paris 1 Panthéon Sorbonne), co-encadrant : Manuele Kirsch Pinheiro. Thèse soutenue le 3 octobre 2019.
 - Taux d'encadrement scientifique estimé : **15%**
 - Travail proposant différentes approches pour la prise en main de l'Analyse de Concepts Formels par des non-experts sur cette méthode. La thèse propose notamment une étude de cas démontrant l'application de cette méthode à des données contextuelles. Ce travail de thèse a donné lieu à 8 publications, dont 1 revue. A la suite de sa thèse, M Jaffal a intégré l'école ITESCIA en tant qu'enseignant coordinateur pédagogique.

3. Salma Najar, **Adaptation des services sensibles au contexte selon une approche intentionnelle**. Directeur de thèse : Carine SOUVEYET (Centre de Recherche en Informatique, Université Paris 1 Panthéon Sorbonne), co-encadrant : Manuele Kirsch Pinheiro. Thèse soutenue le 11 avril 2014.
 - Taux d'encadrement scientifique estimé : **75%**
 - Travail proposant une vision centrée utilisateur des Systèmes d'Information Pervasifs, dans laquelle des mécanismes de découverte et de prédiction de services permettent l'adaptation des services offerts par ces systèmes au contexte d'usage et à l'intention des utilisateurs. Ce travail de thèse a donné lieu à 13 publications, dont 1 chapitre de livre, 1 revue internationale, 6 conférences internationales internationaux (dont un *Best Paper Award* à Service Computation 2011), et 5 conférences nationales. A la suite de sa thèse, Mme Najar a choisi d'intégrer l'industrie et poursuit aujourd'hui sa carrière en tant que consultante en intégration de données.

4. Assia Ait Ali Slimane, **Méthode de sélection intentionnelle des services, en fonction des exigences non fonctionnelles des clients**. Directeur de thèse : Carine SOUVEYET (Centre de Recherche en Informatique, Université Paris 1 Panthéon Sorbonne), co-encadrant : Manuele Kirsch Pinheiro. Thèse soutenue le 20 décembre 2012.
 - Taux d'encadrement scientifique estimé : **15%**
 - Travail proposant une approche méthodologique pour la prise en compte de la qualité de services dans le cas de services dits intentionnels. Il a été proposé un modèle pour la prise en compte de la qualité, exprimée sous la forme d'exigences non-fonctionnels, dans le cadre de services intentionnels, ainsi qu'un processus de configuration et de sélection multicritères. Ce travail de thèse a donné lieu à 6 publications, dont un chapitre de livre. A la suite de sa thèse, Mme Ait Ali Slimane a choisi d'intégrer l'industrie et poursuit aujourd'hui sa carrière en tant que consultante BI.

Animation scientifique

Participation à des jurys de thèse

2 juillet 2019

Membre jury de thèse de Mahamat Charfadine SALIM, « Gestion dynamique et évolutive de règles de sécurité pour l'Internet des Objets » à l'**Université de Reims Champagne Ardennes**. Thèse dirigée par Olivier FLAUZAC et Florent NOLOT.

15 août 2016	Membre jury de thèse de Eráclito DE SOUZA ARGOLO, « <i>Trajatórias conceituais intencionais de ensino e aprendizagem: investigação em fluxo temporal em espaços e contextos nos processos educacionais em EAD</i> » à l' Universidade Federal do Rio Grande do Sul (Porto Alegre, Brésil). Thèse dirigée par José Valdeni de LIMA et codirigée par Lucinéia Heloísa THOM.
19 avril 2016	Membre jury de thèse de Arun Kishore RAMAKRISHNAN, « <i>Support for data-driven context awareness in smart mobile and IoT applications: Resource efficient probabilistic models and a quality-aware middleware architecture</i> » à la Katholieke Universiteit Leuven (Leuven, Belgique). Thèse dirigée par Yolande BERBERS.
16 octobre 2014	Membre jury de thèse de Kelling DA, « <i>Kalimucho-A : autonomic knowledge-based context-driven adaptation platform</i> » à l' Université de Pau et des Pays de l'Adour . Thèse dirigée par Marc DALMAU et codirigée par Philippe ROOSE
11 décembre 2013	Membre jury de thèse de Elvia Kimberly GARCIA GARCIA, « <i>Ascertaining the Availability of Shared Resources in Ubiquitous Collaborative Environments</i> » au Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional , Ciudad de México, Mexique, sous la direction de Sonia Guadalupe MENDOZA CHAPA.

Participation à des comités de sélection

- Comité de sélection pour le poste CNU 27 MCF 1170 à l'Université Lille (campagne de recrutement 2012)
- Comité de sélection pour le poste CNU 27 MCF 0744 à l'Université Paris 1 (campagne de recrutement 2012)
- Comité de sélection pour le poste CNU 27 MCF 343 à l'Université Paris 1 (campagne de recrutement 2010)

Participation à projets

PER-MARE 2013 – 2014	<p>L'objectif du projet PERMARE (STIC-AmSud 13-STIC-07) était celui d'étudier la mise en place et le déploiement d'applications de type MapReduce sur des Grilles Pervasives.</p> <ul style="list-style-type: none"> • <u>Mots clé</u> : Grilles Pervasives, <i>Big Data</i>, Informatique sensible au contexte, Internet des Objets
IST-MUSIC 2007-2008	<p>Le projet européen FP6 IST-MUSIC avait pour objectif la création d'une plateforme ouverte pour le développement d'applications mobiles auto-adaptatives. Cette plateforme inclut des méthodologies pour la conception et le développement de ce type d'application, ainsi qu'un <i>middleware</i> réparti pour garantir l'exécution de ces applications.</p> <ul style="list-style-type: none"> • <u>Mots clé</u> : Informatique Ubiquitaire, Systèmes auto-adaptatifs, Informatique sensible au contexte, Architectures orientées services.

Xwiki Concerto 2007	Ce projet français (financé par la RNTL/ANR) avait pour objectif le développement d'un éditeur collaboratif de type Wiki basé sur un réseau pair-à-pair supportant des terminaux mobiles tels que les téléphones portables et les PDAs.
VULNERALP 2003	Projet français financé par la région Rhône-Alpes visant l'étude et l'estimation des vulnérabilités liées aux risques sismiques dans le bassin grenoblois.
STIMULI 2003-2006	Projet franco-brésilien financé par CAPES-COFECUB, avec pour objectif l'étude et le développement de systèmes informatiques multimédia répartis.
CEMT 2001-2002	Projet franco-brésilien financé par CNPq-INRIA, dont objectif était de développer un environnement d'édition collaborative, ouvert et basé sur le Web, pour l'éducation. Le processus d'édition proposé se caractérise par l'usage un système de workflow et par le support à l'information de conscience de groupe.

Expertises

- Expert évaluation projet dans le cadre d'un AAP « ECOS Nord pour l'automatique et l'informatique » en 2014
- Expert évaluation projet dans le cadre d'un AAP « Futur & Ruptures » de l'Institut Mines-Télécom en 2014
- External reviewer pour le *Natural Sciences and Engineering Research Council of Canada* (NSERC) en 2016

Conférences invitées

- Exposé "O Ensino Superior em Computação na França" (« L'enseignement supérieur en Informatique en France »), réalisée le 30 avril 2013, en compagnie de Dr. Luiz Angelo Steffenele à l'Universidade Federal de Santa Maria (UFSM), Santa Maria, RS, Brésil, lors de la rentrée solennelle du cours en Informatique à l'UFSM.
- Exposé "IST-MUSIC Project Overview" réalisée le 29 juillet 2008 aux 4èmes journées Francophones Mobilité et Ubiquité (UbiMob 2008), à Saint-Malo.
- Exposé "Panorama des outils et des ressources : Blogs, RSS et Wikis" à la journée "Blogs pour les entreprises et les professionnels de l'information", organisée par l'ADBS Lorraine le 25 janvier 2007 à Metz.
- Exposé "Un Mécanisme d'Adaptation Guidé par le Contexte en Utilisant une Représentation par Objets" réalisée à Paris, le 5 septembre 2006 à l'atelier "Etude du contexte", dans le cadre de la 3ème Journées Francophones Mobilité et Ubiquité (UbiMob'06).
- Exposé "Un mécanisme d'adaptation guidé par le contexte en utilisant une représentation par objets" réalisée à Lille, le 1er décembre 2005 à la réunion du groupe de travail "Mobilité et Ubiquité" (GT 4.2 du GdR I3). Disponible sur <http://iihm.imag.fr/nigay/GTMOB/Dec2005/>

Participation comités d'organisation

- **Demo chair**, en compagnie de Pascal Hirmir, 19th International Conference on Pervasive Computing and Communications (PerCom 2021).
- **Guest editor** pour l'édition spéciale "P2P and Cloud computing technologies" du journal Applied Sciences Journal, MDPI, parution prévue début 2021.
- **Co-organisatrice** de l'atelier « Évolution des SI : vers des SI Pervasifs ? » à la conférence INFORSID 2019, édition des proceedings.
- **Publicity chair** pour la conférence 23rd IEEE Enterprise Computing Conference (EDOC 2019)
- **Guest editor** pour l'édition spéciale "Computing Challenges on IoT and Pervasive Systems" de l'International Journal of Information Technologies and Systems Approach (IJITSA), volume 10, issue 2, 2017, IGI Global.
- **Organisateur** et **PC chair** du 2nd Workshop on Big Data and Data Mining Challenges on IoT and Pervasive Systems, organisé conjointement à la 7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016)
- **Organisateur** et **PC chair** du 1st Workshop on Big Data and Data Mining Challenges on IoT and Pervasive Systems, organisé conjointement à la 6th International Conference on Ambient Systems, Networks and Technologies (ANT 2015).
- Membre du **comité d'organisation** des conférences INFORSID 2013 et RCIS 2013.

Participation comités de lecture

Pour un souci d'espace et de clarté, uniquement les comités pour des journaux ou des événements à partir de 2013 ont été pris en considération.

- Relecteur pour le journal *Future Generation Computer Systems* (FGTS), Elsevier
- Relecteur pour le journal *Computer Networks* (COMNET), Elsevier
- Relecteur pour le *Journal of Zhejiang University Science C (Computers & Electronics)*, publié par Springer sous le nouveau nom *Frontiers of Information Technology & Electronic Engineering*

2020 :

- Membre comité de programme pour le *Doctoral Consortium Annual IEEE/ACM International Symposium in Cluster, Cloud, and Grid Computing* (CCGrid 2020)
- Membre comité de programme pour le *13th International Conference on Complex, Intelligent, and Software Intensive Systems* (CISIS-2019), Track 5: Semantic Web, Web Services and Data Integration
- Membre comité de programme pour l'*International Conference Exploring Service Science* (IESS 2.0)
- Membre comité de programme pour l'*International Conference on Ambient Systems, Networks and Technologies* (ANT 2020), track Context awareness & Multimodal interfaces
- Membre comité de programme pour la conférence *INformatique des ORganisations et Systèmes d'Information et de Décision* (INFORSID 2020)

2019 :

- Membre comité de programme pour l'*IEEE International Congress on Internet of Things* (ICIOT 2019)
- Membre comité de programme pour le *13th IEEE International Conference on Research Challenges in Information Science* (RCIS 2019)
- Membre comité de programme pour le *Doctoral Consortium Annual IEEE/ACM International Symposium in Cluster, Cloud, and Grid Computing* (CCGrid 2019)

2018 :

- Membre comité de programme pour le 9th *International Conference on Ambient Systems, Networks and Technologies* (ANT 2018)
- Membre comité de programme pour l'*IEEE 6th International Conference on Future Internet of Things and Cloud* (FiCloud 2018)
- Membre comité de programme pour le 17th *International Conference on Perspectives in Business Informatics Research* (BIR 2018)
- Membre comité de programme pour le 6th *International Conference on Serviceology for Services* (ICServ 2018).
- Membre comité de programme pour le 2018 *World Congress on Services* (ServiceCongress 2018)
- Membre comité de programme pour le 9th *International Conference Exploring Service Science* (IESS 2018), Smart Service Processes track
- Membre comité de programme pour le 15th *IEEE International Conference on Advanced and Trusted Computing* (ATC 2018)
- Membre comité de programme pour le 5th *Workshop on Middleware for Context-Aware Applications in the IoT* (M4IOT 2018)

2017 :

- Membre comité de programme pour le *workshop Quality Awareness in Modeling and Using Context in Applications (QAMUCA)*, intégré à la conférence 20th *International and Interdisciplinary Conference on Modeling and Using Context* (CONTEXT-17)
- Membre comité de programme pour le 5th *International Conference on Serviceology for Services* (ICServ), July 2017.
- Membre comité de programme pour le 11th *IEEE International Conference on Research Challenges in Information Science* (RCIS 2017), Poster & Demo Track
- Membre comité de programme pour le 14th *IEEE International Conference on Advanced and Trusted Computing* (ATC 2017)
- Membre comité de programme pour le 8th *International Conference on Exploring Service Science*, IESS 1.7, 2017
- Membre comité de programme pour le 4th *Workshop on Middleware for Context-Aware Applications in the IoT* (M4IOT 2017), intégré à la ACM/IFIP/USENIX ACM International Middleware Conference 2017 (Middleware 2017)

2016 :

- Membre comité de programme pour le 3rd *Workshop on Middleware for Context-Aware Applications in the IoT* (M4IOT 2016), intégré à la ACM/IFIP/USENIX ACM International Middleware Conference 2016 (Middleware 2016)
- Membre comité de programme pour le 7th *International Conference on Exploring Service Science 1.6* (IESS1.6)
- Membre comité de programme pour le 4th *Workshop on Cloud for IoT* (CLIoT 2016), intégré à la 5th *European Conference on Service-Oriented and Cloud Computing* (ESOCC 2016)
- Membre comité de programme pour le 13th *IEEE International Conference on Advanced and Trusted Computing* (ATC 2016), Track 4 *Middleware for the Internet of Things*
- Membre comité de programme pour le 10th *IEEE International Conference on Research Challenges in Information Science* (RCIS 2016) - Poster & Demo Track
- Membre comité de programme pour le 7th *International Conference on Ambient Systems, Networks and Technologies* (ANT 2016)
- Membre comité de programme pour le 2nd *Workshop on Big Data and Data Mining Challenges on IoT and Pervasive Systems* (BigD2M 2016), intégré à 7th *International Conference on Ambient Systems, Networks and Technologies* (ANT 2016)

- Membre comité de programme pour le *Workshop on Pervasive Information System (PIS 2016)*, intégré à la 4th World Conference on Information Systems and Technologies (WorldCist 2016)
- Membre comité de programme pour le *10th International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2016)*
- Membre comité de programme pour le *6th International Conference on Ambient Computing, Applications, Services and Technologies (AMBIENT 2016)*
- Membre comité de programme pour le *8th International Conferences on Advanced Service Computing (Service Computation 2016)*
- Membre comité de programme pour le *41th IEEE Conference on Local Computer Networks (LCN 2016)*

2015 :

- Membre comité de programme pour le *2nd Workshop on Middleware for Context-Aware Applications in the IoT (M4IOT 2015)*, intégré à ACM/IFIP/USENIX ACM International Middleware Conference 2015 (Middleware 2015)
- Membre comité de programme pour le 16th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing Doctoral Consortium (DCCCGGrid 2015)
- Membre comité de programme pour le *4th International Workshop on Adaptive Case Management and other non-workflow approaches to BPM (AdaptiveCM 2015)*
- Membre comité de programme pour le *6th International Conference on Ambient Systems, Networks and Technologies (ANT2015)*
- Membre comité de programme pour le 1st International Workshop on Big Data and Data Mining Challenges on IoT and Pervasive Systems (BigD2M 2015), intégré à 6th International Conference on Ambient Systems, Networks and Technologies (ANT2015)
- Membre comité de programme pour le *Workshop on Pervasive Information System (PIS 2015)*, intégré à 3rd World Conference on Information Systems and Technologies (WorldCist 2015)
- Membre comité de programme pour le *9th IEEE International Conference on Research Challenges in Information Science (RCIS 2015) - Poster & Demo Track*
- Membre comité de programme pour le *7th International Conferences on Advanced Service Computing (Service Computation 2015)*
- Membre comité de programme pour le *40th Annual IEEE Conference on Local Computer Networks (LCN 2015)*
- Membre comité de programme pour le *Workshop on Healthy and Secure People (HSP 2015)*, intégré à 4th International Conference on Health and Social Care IST (HCist 2015)
- Membre comité de programme pour l'*International Conference on Computing and Network Communications (CoCoNet'15)*

2014 :

- Membre comité de programme pour le *8th IEEE International Conference on Research Challenges in Information Science (RCIS 2014) - Poster & Demo Track*
- Membre comité de programme pour le *3rd International Workshop on Adaptive Case Management and other non-workflow approaches to BPM (AdaptiveCM 2014)*
- Membre comité de programme pour le *1st Workshop on Middleware for Context-Aware Applications in the IoT (M4IOT 2014)*, intégré à ACM/IFIP/USENIX ACM International Middleware Conference 2014 (Middleware 2014)

2013:

- Membre comité de programme pour le *2nd International Workshop on Adaptive Case Management and other non-workflow approaches to BPM (AdaptiveCM 2013)*
- Membre comité de programme pour le *CAISE'13 Forum*, intégré à la 25th International Conference on Advanced Information Systems Engineering (CAISE)

- Membre comité de programme pour le 5th *International Conferences on Advanced Service Computing* (Service Computation 2013)

Synthèse des publications

Les publications ont été organisées en fonction de l'évolution de ma carrière : jusqu'en 2008, date de mon intégration à l'Université Paris 1 Panthéon Sorbonne, entre 2008 et 2014 (inclus), représentant la première partie de ma carrière dans cette université, puis à partir de 2014 jusqu'à présent.

	Nb total de publications	Nb de publi. jusqu'en 2008	Nb de publi. 2008 < x ≤ 2014	Nb de publi. après 2014	Classés A (classement ERA ³⁷)	Classées B (classement ERA)
Chapitres de Livres	9	0	7	2	N/A	N/A
Articles en journal	12	3	3	6	1	2
Articles en conférence international	48	14	20	14	4	8
Articles en conférence national	13	3	5	5	N/A	N/A
Rapports & mémoires	5	4	1	0	N/A	N/A
Total	87	24	36	27	5	10

Enseignements depuis 2008

Matières enseignées

2008 – jusqu'à présent	<p>Bases de données relationnelles :</p> <ul style="list-style-type: none"> • Description : Introduction aux bases de données relationnelles, à leur conception et leur interrogation (algèbre relationnelle et SQL notamment). • Public visé : étudiants en Licence L3 parcours Gestion et Finances. • Volume type : 18h cours magistraux, 33h à 66h de travaux dirigés.
2009 – jusqu'à présent	<p>Systèmes d'Information & Informatique :</p> <ul style="list-style-type: none"> • Description : Introduction aux Systèmes d'Information, modélisation de besoins et de processus en UML, introduction pratique aux projets informatique par la construction d'un site Web.

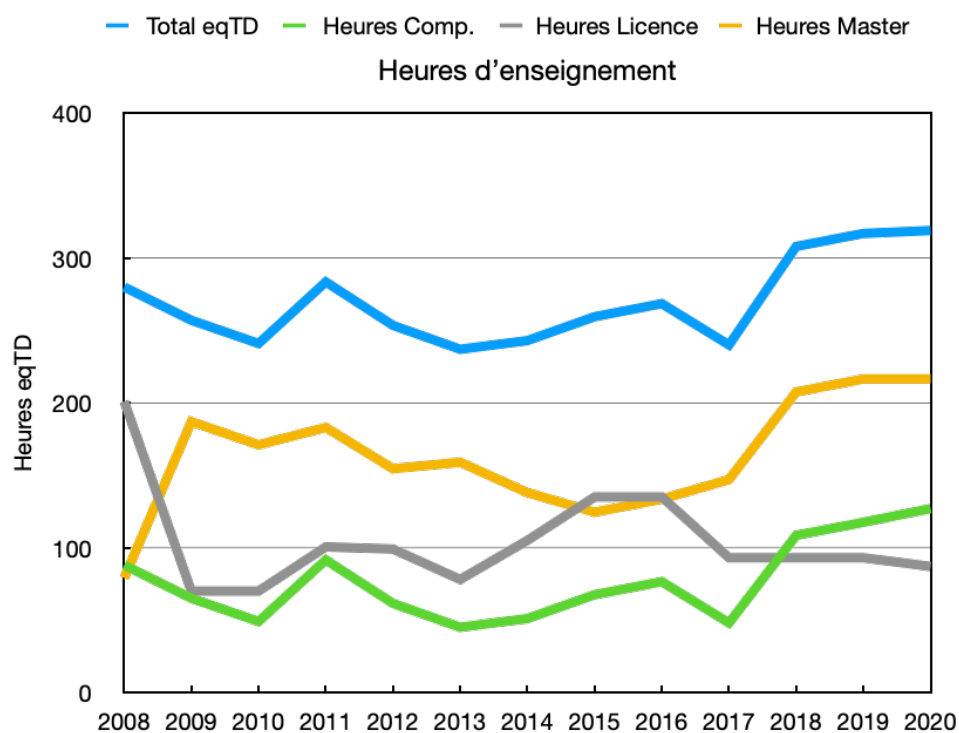
³⁷ Le classement ERA est le classement international le plus connu dans le domaine de l'Informatique (CNU 27). Intentionnellement reconnu par la communauté scientifique, il est élaboré par l'association Australienne CORE (*Computing Research and Education Association*). Il est disponible sur <https://www.core.edu.au/conference-portal>, mais également sur <http://www.conferenceranks.com>, site regroupant d'autres classements en Informatique, comme le Qualis (classement proposé par l'organisme de recherche officiel du gouvernement brésilien CAPES) ou encore MSAR, proposé par Microsoft.

	<ul style="list-style-type: none"> • Public visé : étudiants en Master M1 en Gestion. • Volume type : 36h cours magistraux, 33h à 66h de travaux dirigés.
2016 – jusqu'à présent	<p>Cloud & Pervasive Computing :</p> <ul style="list-style-type: none"> • Description : Introduction au Cloud Computing, étude de l'impact des nouvelles technologies issues de l'Informatique Pervasive sur les Systèmes d'Information. • Public visé : étudiants en Master M2 MIAGE. • Volume type : 12h de cours en français, 12h de cours en anglais.
2019 – jusqu'à présent	<p>Excel appliqué à la gestion</p> <ul style="list-style-type: none"> • Description : application de l'outil Excel pour l'analyse de données en entreprise. • Public visé : étudiants en Master M2 Management Stratégique parcours EMEP, et en M2 Management et Commerce International parcours GEAI. • Volume type : 8h à 16h de cours.
2008 - 2016	<p>Coopération & Ubiquité :</p> <ul style="list-style-type: none"> • Description : étude sur les outils de <i>groupware</i> et autres technologies utilisées pour le support au travail en équipe, ainsi que l'impact de la mobilité sur le travail en équipe • Public visé : étudiants en Master M1 MIAGE. • Volume type : 30h cours magistraux, 30h travaux dirigés.
2009 - 2014	<p>Ingénierie de Systèmes à base de services :</p> <ul style="list-style-type: none"> • Description : étude sur les architectures orientées services et les recherches en <i>Service Oriented Computing</i> • Public visé : étudiants en Master M2 recherche SID. • Volume type : 12h cours.
2011- 2017	<p>Architecture Orientée Objets :</p> <ul style="list-style-type: none"> • Description : étude des principes de qualité sur les architectures OO, programmation OO avancée • Public visé : étudiants en Master L3 MIAGE. • Volume type : 30h cours magistraux, 30h travaux dirigés.
2008 – 2009	<p>Programmation Java :</p> <ul style="list-style-type: none"> • Description : programmation d'interfaces HM en Java • Public visé : étudiants en Master L3 MIAGE. • Volume type : 30h cours magistraux, 30h travaux dirigés.
2008 – 2009	<p>Programmation C :</p> <ul style="list-style-type: none"> • Description : introduction à la programmation en langage C

- Public visé : étudiants en Licence L1 MASS.
- Volume type : 18h cours magistraux, 24h travaux dirigés.

Volume d'enseignement

Année	Total eqTD	Heures Comp.	Heures Licence	Heures Master
2008	280	88	201	79
2009	257	65	70	187
2010	241	49	70	171
2011	283,5	91,5	100,5	183
2012	253,5	61,5	99	154,5
2013	237	45	78	159
2014	243	51	105	138
2015	259,5	67,5	135	124,5
2016	268,5	76,5	135	133,5
2017	240	48	93	147
2018	308	108,5	93	207,5
2019	317	117,5	93	216,5
2020	319	127	87	216,5



Responsabilités administratives

Oct. 2020 - à présent	Référent numérique pour l'École de Management de la Sorbonne
2018 – à présent	Membre de la commission ParcoursSup pour la licence L1 Gestion à l'École de Management de la Sorbonne
2012 - à présent	Responsable équipe pédagogique informatique L3/M1 de l'École de Management de la Sorbonne <ul style="list-style-type: none">• Activités : organisation l'équipe pédagogique, planning de TD, recrutement de vacataires, suivi de dossiers de vacation• Effectif concerné : 27 groupes de TD, ± 15 enseignants/vacataires, ± 600 étudiants organisés en 18 VET (code diplôme)
2018 – 2020	Responsable suivi mémoire au Master M1 Management des Systèmes d'Information parcours SIC <ul style="list-style-type: none">• Activités : formation et suivi des étudiants dans l'élaboration de leur rapport d'activité et mémoire de recherche• Effectif concerné : ± 25 étudiants en apprentissage
2016-2017	Responsable pédagogique de la Licence L3 MIASH parcours MIAGE <ul style="list-style-type: none">• Activités : organisation équipe pédagogique, mise en place enseignement hybride, réalisation de jurys et commissions d'admission, suivi des étudiants• Effectif concerné : ± 60 étudiants, dont 1 groupe en apprentissage
2009 - 2015	Directeur d'études de la licence L3 MIAGE (Informatique des Organisations) <ul style="list-style-type: none">• Activités : soutien au responsable de formation dans le suivi des étudiants (réalisation d'entretiens individuels, participations aux jurys et commission d'admission), participation à l'élaboration du dossier d'évaluation.• Effectif concerné : ± 55 étudiants par an, dont 1 groupe en apprentissage
2009 -2012	Membre du Comité Consultatif Scientifique (CSS) pour les CNU 26-27 de l'Université Paris 1

Annexe II – Liste de Travaux et Publications

Liste de Travaux et Publications

Tableau de synthèse

Les publications ont été organisées selon la période de ma carrière : jusqu'en 2008, date de mon intégration au Centre de Recherche en Informatique de l'Université Paris 1 Panthéon Sorbonne, entre 2008 et 2014 (inclus), représentant la première partie de ma carrière dans cette université, puis à partir de 2014 jusqu'à présent.

	Nb total de publications	Nb de publ. jusqu'en 2008	Nb de publ. 2008 < x ≤ 2014	Nb de publ. après 2014	Classés A (classement ERA ³⁸)	Classées B (classement ERA)
Chapitres de Livres	9	0	7	2	N/A	N/A
Articles en journal	12	3	3	6	1	2
Articles en conférence international	48	14	20	14	4	8
Articles en conférence national	13	3	5	5	N/A	N/A
Rapports & mémoires	5	4	1	0	N/A	N/A
Total	87	24	36	27	5	10

Chapitres de Livres

- | | |
|------|--|
| 2018 | Steffenel, L.A., Kirsch-Pinheiro, M., Vaz Peres, L., Kirsch Pinheiro, D. "Strategies to implement Edge Computing in a P2P Pervasive Grid", In Mehdi Khosrow-Pour et al. (Eds.), Fog Computing: Breakthroughs in Research and Practice , IGI Global, pp 142-157, 2018, doi:10.4018/978-1-5225-5649-7.ch006 . |
| 2016 | Deneckere, R., Hug, C., Jaffal, A. Kirsch Pinheiro, M., Le Grand, B., Mazo, M., Rychkova, I. "Context Management and Intention Mining for Adaptive Systems in Mobile Environments: from Business Process Management to Video Games?". In Digital Interfaces in Situations of Mobility: Cognitive, Artistic, and Game Devices . Common Ground Publishing, 2016. |

³⁸ Le classement ERA est le classement international le plus connu dans le domaine de l'Informatique (CNU 27). Intentionnellement reconnu par la communauté scientifique, il est élaboré par l'association Australienne CORE (*Computing Research and Education Association*). Il est disponible sur <https://www.core.edu.au/conference-portal>, mais également sur <http://www.conferenceranks.com>, site regroupant d'autres classements en Informatique, comme le Qualis (classement proposé par l'organisme de recherche officiel du gouvernement brésilien CAPES) ou encore MSAR, proposé par Microsoft.

- 2014 Rychkova I., Kirsch-Pinheiro M., Le Grand B., "Automated Guidance for Case Management: Science or Fiction?", In : Ficher, L. (Ed.), **Empowering Knowledge Workers: New Ways to Leverage Case Management**, Series BPM and Workflow Handbook Series, Future Strategies Inc., 2014, pp. 67-78. ISBN : 978-0-984976478
- 2013 Vanrompay, Y., Kirsch Pinheiro, M., Ben Mustapha, N., Aufaure, M.-A., "Context-Based Grouping and Recommendation in MANET", In : Kolomvatsos, K., Anagnostopoulos, C., Hadjiefthymiades, S. (Eds.), **Intelligent Technologies and Techniques for Pervasive Computing**, IGI Global, 2013, pp. 157-178. ISBN : 978-1-4666-4040-5. DOI : 10.4018/978-1-4666 4038-2.ch008
- Najar, S., Kirsch Pinheiro, M., Vanrompay, Y., Steffemel, L.A., Souveyet, C., "Intention Prediction Mechanism in an Intentional Pervasive Information System", In : Kolomvatsos, K., Anagnostopoulos, C., Hadjiefthymiades, S. (Eds.), **Intelligent Technologies and Techniques for Pervasive Computing**, IGI Global, 2013, pp. 251-275. ISBN : 978-1-4666-4040-5. DOI: 10.4018/978-1-4666-4038-2.ch014.
- 2012 Vanrompay, Y., Kirsch-Pinheiro, M., Berbers, Y., "Service Selection with Uncertain Context Information", In: Stephan Reiff-Marganiec and Marcel Tilly (Eds.), **Handbook of Research on Service-Oriented Systems and Non-Functional Properties: Future Directions**, IGI Global, pp. 192-215, ISBN 978-1613504321, DOI: 10.4018/978-1-61350-432-1
- Ait-Ali-Slimane, A., Kirsch-Pinheiro, M., Souveyet, C., "Considering Quality of a Service in an Intentional Approach", In: Stephan Reiff-Marganiec and Marcel Tilly (Eds.), **Handbook of Research on Service-Oriented Systems and Non-Functional Properties: Future Directions**, IGI Global, pp. 334-351, ISBN 978-1613504321, DOI: 10.4018/978-1-61350-432-1
- 2010 Kirsch Pinheiro, M., Carrillo-Ramos, A., Villanova-Oliver, M., Gensel, J., Berbers, Y. "Context-Aware Adaptation in Web-based Groupware Systems", In: JingTao Yao (Ed.), **Web-based Support Systems**, Series: Advanced Information and Knowledge Processing, Springer, mars 2010, ISBN: 978-1-84882-627-4, pp. 3-31. DOI: 10.1007/978-1-84882-628-1_1
- 2009 Carrillo-Ramos, A., Kirsch Pinheiro, M., Villanova-Oliver, M., Gensel, J., Berbers, Y., "Collaborating agents for adaptation to mobile users", In: Max Chevalier, Chantal Soule-Dupuy, Christine Julien (Eds.), **Collaborative and Social Information Retrieval and Access: Techniques for Improved User Modeling**, IGI Global, 2009, ISBN 978-1605663067.
- Preuveneers, D., Victor, K., Vanrompay, Y., Rigole, P., Kirsch Pinheiro, M., Berbers, Y., "Context-Aware Adaptation in an Ecology of Applications", In: Dragan Stojanovic (Ed.), **Context-Aware Mobile and Ubiquitous Computing for Enhanced Usability: Adaptive Technologies and Applications**, IGI Global, 2009, ISBN 978-1605662909.

Articles en journal ou revue

- 2019 Kirsch-Pinheiro, M. & Souveyet, C. "Is Group-Awareness Context-Awareness?: Converging Context-Awareness and Group Awareness Support", **International Journal of e-Collaboration (IJeC)**, 15(3) , 2019, DOI: 10.4018/IJeC.2019070101.
Classement ERA : B
- 2018 Kirsch-Pinheiro, M. & Souveyet, C. "Supporting context on software applications: a survey on context engineering" (« Le support applicatif à la notion de contexte : revue de la littérature en ingénierie de contexte »), **Modélisation et utilisation du contexte**, 2(1), 2018
- Steffenel, L.A., Kirsch-Pinheiro, M., Vaz Peres, L., Kirsch Pinheiro, D. "Strategies to implement Edge Computing in a P2P Pervasive Grid", **International Journal of Information Technologies and Systems Approach (IJITSA)**, 11(1), IGI Global, pp 1-15, 2018, doi:10.4018/IJITSA/2018010101.
- 2016 Cassales, G. W., Charao, A., Kirsch-Pinheiro, M., Souveyet, C., Steffenel, L.A. "Improving the Performance of Apache Hadoop on Pervasive Environments through Context-Aware Scheduling ", **Journal of Ambient Intelligence and Humanized Computing**, 7(3), Springer, pp. 333-345, 2016. doi:10.1007/s12652-016-0361-8.
- 2015 Najar, S., Kirsch-Pinheiro, M., Souveyet, C. "Service discovery and prediction on Pervasive Information System", **Journal of Ambient Intelligence and Humanized Computing**, vol. 6, issue 4, June 2015, Springer Berlin Heidelberg, pp. 407-423. ISSN: 1868-5137. DOI: <http://dx.doi.org/10.1007/s12652-015-0288-5>
- Engel, T., Charão, A., Kirsch-Pinheiro, M., Steffenel, L.-A., "Performance improvement of data mining in Weka through multi-core and GPU acceleration: opportunities and pitfalls", **Journal of Ambient Intelligence and Humanized Computing**, vol. 6, issue 4, June 2015, Springer Berlin Heidelberg, pp. 377-390. ISSN: 1868-5137. DOI: <http://dx.doi.org/10.1007/s12652-015-0292-9>
- 2014 Steffenel, L. A., Flauzac, O., Charao, A. S., Barcelos, P. P., Stein, B., Cassales, G., Nesmachnow, S., Rey, J., Cogorno, M., Kirsch-Pinheiro, M., Souveyet, C. "MapReduce Challenges on Pervasive Grids", **Journal of Computer Science**, vol. 10, issue 11, 2014, Science Publications, pp. 2192-2207. DOI : 10.3844/jcssp.2014.2192.2207. URL: <http://thescpub.com/abstract/10.3844/jcssp.2014.2194.2210>
- 2012 Najar S., Kirsch-Pinheiro, M., Souveyet, C. "Bringing Context to Intentional Services: Enriched Semantic Service Description for Service Discovery", **International journal On Advances In Intelligent Systems**, vol. 5, issue 1 & 2, June 2012, pp. 159-174, IARIA Journals / ThinkMind, ISSN: 1942-2679
- 2010 Rolland, C., Kirsch-Pinheiro, M., Souveyet, C. "An Intentional Approach to Service Engineering", **IEEE Transactions on Services Computing**, vol.3, no.4, pp.292-305, Oct.-Dec. 2010. DOI: <http://dx.doi.org/10.1109/TSC.2010.26>
Classement ERA : A
- 2003 Kirsch-Pinheiro, M.; Lima, J.V.; Borges, M.R.S. "A Framework for Awareness Support in Groupware Systems". **Computer in Industry**, vol. 52 n° 1, Elsevier, 2003, pp. 47-57.
Classement ERA : B

- 2001 Kirsch-Pinheiro, M.; Telecken, T.; Zeve, C.D.C; Lima, J.V.L.; Edelweiss, N. "A Cooperative Environment for E-Learning Authoring". **Documents Numériques : Espaces d'Information et de Coopération**, vol. 5, n° 3-4, Hermes Science, France, 2002, pp. 89-114.
- Lima, J.V. de; Kirsch-Pinheiro, M.; Telecken, T.; Edelweiss, N.; Zeve, C.D.C; Borges, T. B.; Maciel, C. B. "O uso de open source e standards para o ensino a distância dentro do projeto CEMT". **Cadernos de Informática**, vol. 2, n° 1, Instituto de Informática – URGs, Porto Alegre, Brésil, pp. 103-105, 2001.

Articles en conférences et workshops internationaux avec comité de lecture

- 2020 Ben Rabah, N.; Kirsch Pinheiro, M.; Le Grand, B.; Jaffal, A. & Souveyet, C., "Machine Learning for a Context Mining Facility", **16th Workshop on Context and Activity Modeling and Recognition, (COMOREA 2020), IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)**, 2020, pp.678-684
- 2018 Kirsch Pinheiro, M., Souveyet, C., "Is Group-Awareness Context-Awareness?", In: Rodrigues, A.; Fonseca, B. & Pregoça, N. (Eds.), **24th International Conference on Collaboration and Technology (CRIWG 2018)**, Lecture Notes on Computer Science, vol. 11001, Springer, 2018, 198-206
- Steffenel, L.A., Kirsch-Pinheiro, M., "Improving Data Locality in P2P-based Fog Computing Platforms", **9th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2018)**, Leuven, Belgium, November 5-8, 2018.
- 2017 Kirsch-Pinheiro, M., Souveyet, C. "Quality on Context Engineering Modeling and Using Context", **10th International and Interdisciplinary Conference, CONTEXT 2017**, Paris, France, June 20-23, Lecture Notes on Computer Science, vol. 10257, 2017, pp. 432-439
Classement ERA : C
- Beserra, D., Kirsch-Pinheiro, M., Steffenel, L.A., Moreno, E.D., "Comparing the Performance of OS-level Virtualization Tools in SoC-based Systems: The Case of I/O-bound Applications", **22nd IEEE Symposium on Computers and Communications (ISCC 2017)**, Heraklion, Greece, July 3-6, 2017. doi: 10.1109/ISCC.2017.8024598
Classement ERA : B
- Charao, A., Hoffmann, G., Steffenel, L.A, Kirsch-Pinheiro, M., Stein, B., "Performance Evaluation of Cloud-based RDBMS through a Cloud Scripting Language", **19th International Conference on Enterprise Information Systems (ICEIS)**, Porto, Portugal, April 26-29, 2017. doi: 10.5220/0006350803320337
Classement ERA : C
- Beserra, D., Kirsch-Pinheiro, M., Steffenel, L.A., Souveyet, C., Moreno, E.D., "Performance Evaluation of OS-level Virtualization Solutions for HPC Purposes on SoC-based Systems", **IEEE International Conference on Advanced Information Networking**

- and Applications (AINA)**, Taipei, Taiwan, March 27-29, 2017. DOI: 10.1109/AINA.2017.73
Classement ERA : B
- 2016 Kirsch-Pinheiro, M., Mazo, R., Souveyet, C., Sprovieri, D., "Requirements Analysis for Context-oriented Systems", **7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016)**, Procedia Computer Science, vol. 83, 2016, Elsevier, pp. 253–261. DOI : <http://dx.doi.org/10.1016/j.procs.2016.04.123>
- Steffenel, L.A., Kirsch-Pinheiro, M., Kirsch Pinheiro, D., Perez, L.V. "Using a Pervasive Computing Environment to Identify Secondary Effects of the Antarctic Ozone Hole", **BigD2M Workshop, ANT/SEIT 2016**, Madrid, Spain. Procedia Computer Science, vol. 83, 2016, Elsevier, pp. 1007-1012, 2016.
- 2015 Steffenel, L.A., Kirsch-Pinheiro, M., "When the Cloud goes Pervasive: approaches for IoT PaaS on a ubiquitous world", **EAI International Conference on Cloud, Networking for IoT systems (CN4IoT 2015)**, Rome, Italy, October 126-27, 2015.
- Cassales, G.W., Charao, A., Kirsch-Pinheiro, M., Souveyet, C., Steffenel, L.A., "Context-Aware Scheduling for Apache Hadoop over Pervasive Environments", **6th International Conference on Ambient Systems, Networks and Technologies (ANT 2015)**, Procedia Computer Science, vol. 52, Jun 2015, Elsevier, pp. 202–209. DOI: <http://dx.doi.org/10.1016/j.procs.2015.05.058>.
- Steffenel, L.A., Kirsch-Pinheiro, M., "Leveraging Data Intensive Applications on a Pervasive Computing Platform: The Case of MapReduce", **BigD2M Workshop, ANT/SEIT 2015**, London, England. Procedia Computer Science, vol. 52, pp. 1034-1039, 2015.
- Jaffal, A., Le Grand, B., Kirsch-Pinheiro, M., "Refinement Strategies for Correlating Context and User Behavior in Pervasive Information Systems", **BigD2M Workshop, ANT/SEIT 2015**, London, England. Procedia Computer Science, vol. 52, pp. 1040-1046, 2015.
- Steffenel, L.A., Kirsch-Pinheiro, M., " CloudFIT, a PaaS Platform for IoT Applications over Pervasive Networks ", **ESOC Workshops 2015**: 20-32, Taormina, Italy. pp. 20-32, 2015.
- 2014 Engel, T.A., Charão, A.S., Kirsch-Pinheiro, M., Steffenel, L.A., "Performance Improvement of Data Mining in Weka through GPU Acceleration", **5th International Conference on Ambient Systems, Networks and Technologies (ANT-2014)**, Procedia Computer Science, vol. 32, 2014, Elsevier, pp. 93–100. DOI: <http://dx.doi.org/10.1016/j.procs.2014.05.402>
- Najar, S., Kirsch Pinheiro, M., Souveyet, C., "A new approach for service discovery and prediction on Pervasive Information System", **The 5th International Conference on Ambient Systems, Networks and Technologies (ANT-2014)**, 2-5 June 2014, Hasselt, Belgium, Procedia Computer Science, vol. 32, 2014, Elsevier, pp. 421–428. DOI: <http://dx.doi.org/10.1016/j.procs.2014.05.443>
- Najar, S., Kirsch Pinheiro, M., Le Grand, B., Souveyet, C. "A user-centric vision of service-oriented Pervasive Information Systems", **8th International Conference on Research Challenges in Information Science**, May 28-30 2014, Marrakesh, Morocco, pp. 359-370
Classement ERA : B

Najar, S., Kirsch Pinheiro, M., Souveyet, C. "A context-aware intentional service prediction mechanism in PIS", In: David De Roure, Bhavani Thuraisingham and Jia Zhang (Eds.), **IEEE 21st International Conference on Web Services (ICWS 2014)**, 27 June - 2 July 2014, Anchorage, Alaska, USA, IEEE CS, pp. 662-669. DOI : [10.1109/ICWS.2014.97](https://doi.org/10.1109/ICWS.2014.97)
Classement ERA : A

Jaffal, A., Kirsch-Pinheiro, M., Le Grand, B., "Unified and Conceptual Context Analysis in Ubiquitous Environments", In: Jaime Lloret Mauri, Christoph Steup, Sönke Knoch (Eds.), **8th International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2014)**, August 24 - 28, 2014 – Rome, Italy, ISBN: 978-1-61208-353-7, IARIA, pp. 48-55.
Classement ERA : C

Cassales, G.W., Charão, A.S., Kirsch-Pinheiro, M., Souveyet, C., Steffemel, L.A., "Bringing Context to Apache Hadoop", In: Jaime Lloret Mauri, Christoph Steup, Sönke Knoch (Eds.), **8th International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2014)**, August 24 - 28, 2014 – Rome, Italy, ISBN: 978-1-61208-353-7, IARIA, pp. 252-258.
Classement ERA : C

2013

Garcia, K., Kirsch-Pinheiro, M., Mendoza, S., Decouchant, D., "An Ontological Model for Resource Sharing in Pervasive Environments", **2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)**, volume 1, Nov. 17-20, 2013 Atlanta, GA USA, pp. 179-184.
DOI : <http://dx.doi.org/10.1109/WI-IAT.2013.27>
Classement ERA : C

Garcia, K., Kirsch-Pinheiro, M., Mendoza, S., Decouchant, D., "Ontology-Based Resource Discovery in Pervasive Collaborative Environments", **19th International Conference on Collaboration and Technology, CRIWG 2013**, Lecture Notes in Computer Science, vol. 8224, 30 October - 1 November 2013, Wellington, New Zealand, pp. 233-240. DOI : http://dx.doi.org/10.1007/978-3-642-41347-6_17

Steffemel, L.A., Flauzac, O., Charao, A.S., Barcelos, P.P., Stein, B., Nesmachnow, S. Kirsch Pinheiro, M., Diaz, D., "PER-MARE: Adaptive Deployment of MapReduce over Pervasive Grids", **8th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC'13)**, Compiègne, France, Oct. 28-30, 2013, pp. 17-24.

Rychkova I., Kirsch Pinheiro M., Le Grand B., "Context-Aware Agile Business Process Engine: Foundations and Architecture", In: Nurcan, S., Proper, H., Soffer, P., Krogstie, J., Schmidt, R., Halpin, T. & Bider, I. (Eds.), **Enterprise, Business-Process and Information Systems Modeling**, Proceedings of the 14th Working Conference on Business Process Modeling, Development, and Support (BPMDS 2013), Lecture Notes in Business Information Processing, vol. 146, Valence : Espagne, 2013, pp. 32-47.
Classement ERA : C

Kirsch Pinheiro M., Rychkova I., "Dynamic Context Modeling for Agile Case Management", In: Y.T. Demey and H. Panetto (Eds.), **2nd International Workshop on Adaptive Case Management and other non-workflow approaches to BPM (AdaptiveCM 2013)**, OnTheMove Federated Workshop (OTM 2013 Workshops), LNCS 8186, Graz, Austria, 9-13 September 2013, Springer, pp. 144–154, 2013.

- 2012 Najar, S. Kirsch-Pinheiro, M., Souveyet, C., Steffemel, L. A. "Service Discovery Mechanisms for an Intentional Pervasive Information System", **Proceedings of 19th IEEE International Conference on Web Services (ICWS 2012)**, Honolulu, Hawaii, 24-29 June 2012, pp. 520-527, DOI: 10.1109/ICWS.2012.84
Classement ERA : A
- 2011 Najar, S., Kirsch Pinheiro, M., Souveyet, C. "Bringing context to intentional services". **3rd Int. Conference on Advanced Service Computing, Service Computation'11**, Rome, Italy, pp. 118-123, 2011. URL: http://www.thinkmind.org/index.php?view=article&articleid=service_computation_2011_5_30_10136 (Best Paper Award)
- Najar, S., Kirsch Pinheiro, M., Souveyet, C. "The Influence of Context on Intentional Service". **5th Int. IEEE Workshop on Requirements Engineerings for Services (REFS'11)**, IEEE Conference on Computers, Software, and Applications (COMPSAC'11), Munich, Germany, pp. 470-475, 2011.
- Najar, S., Kirsch Pinheiro, M., Souveyet, C. "Towards semantic modeling of intentional pervasive information systems". **6th International Workshop on Enhanced Web Service Technologies (WEWST'11)**, European Conference on Web Services (ECOWS'11), Lugano, Switzerland, pp. 30-34, 2011.
- 2009 Steffemel, L. A., Kirsch-Pinheiro, M. "Strong Consistency for Shared Objects in Pervasive Grid". **Proceedings of the 5th IEEE International Conference on Wireless and Mobile Computing, Networking and Communication (WiMob'2009)**, Marrakesh, Morocco, 12-14 October 2009. pp. 73-78.
Classement ERA : C
- Ait Ali Slimane, A., Kirsch Pinheiro, M., Souveyet, C. "Goal Reasoning for Quality Elicitation in the ISOA approach", **3rd International Conference on Research Challenges in Information Science (RCIS)**, April 22-24, 2009, Fes, Morocco.
Classement ERA : B
- Devlic, A., Reichle, R., Wagner, M., Kirsch-Pinheiro, M., Vanrompay, Y., Berbers, Y., Valla, M. "Context Inference of Users' Social Relationships and Distributed Policy Management". **6th IEEE Workshop on Context Modeling and Reasoning (CoMoRea), 7th IEEE International Conference on Pervasive Computing and Communication (PerCom'09)**, Galveston, Texas, 13 March 2009, pp. 1-8
- Vanrompay, Y., Kirsch-Pinheiro, M., Berbers, Y., "Context-Aware Service Selection with Uncertain Context Information", **Context-Aware Adaptation Mechanism for Pervasive and Ubiquitous Services 2009 (CAMPUS 2009)**, Electronic Communications of the EASST, Volume 19.
- Najar, S., Saidani, O., Kirsch-Pinheiro, M., Souveyet, C., and Nurcan, S. "Semantic representation of context models: a framework for analyzing and understanding". In: J. M. Gomez-Perez, P. Haase, M. Tilly, and P. Warren (Eds), **Proceedings of the 1st Workshop on Context, information and ontologies (CIAO 09), European Semantic Web Conference (ESWC'2009)**, (Heraklion, Greece, June 01 - 01, 2009). ACM, New York, NY, 1-10. DOI=<http://doi.acm.org/10.1145/1552262.1552268>

2008

Kirsch-Pinheiro, M. Vanrompay, Y., Berbers, Y. "Context-aware service selection using graph matching", **2nd Non Functional Properties and Service Level Agreements in Service Oriented Computing Workshop (NFPSLA-SOC'08)**, at ECOWS 2008, Dublin, Ireland, November 12, 2008

Steffenel, L.A.; Kirsch-Pinheiro, M.; Bebers, Y. "Total Order Broadcast on Pervasive Systems", **23rd Annual ACM Symposium on Applied Computing (SAC 2008)**, Fortaleza, Brazil, March 2008.

Classement ERA : B

Kirsch-Pinheiro, M., Vanrompay, Y. Victor, K., Berbers, Y., Valla, M.; Frà, C., Mamelli, A., Barone, P., Hu, X., Devlic, A., Panagiotou, G. "Context Grouping Mechanism for Context Distribution in Ubiquitous Environments", In: Robert Meersman, Zahir Tari et al.(eds.), **10th International Symposium on Distributed Objects, Middleware, and Applications (DOA'08), OTM 2008 Conferences**, Nov 10 - 12, 2008, Monterrey, Mexico.

Kirsch Pinheiro, M., Villanova-Oliver, M., Gensel, J., Berbers, Y., Martin, H. "Personalizing Web-Based Information Systems through Context-Aware User Profiles", **2nd International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2008)**, September 29 - October 4, Valencia, Spain.

Classement ERA : C

Hu, X., Ding, Y., Paspallis, N., Bratskas, P., Papadopoulos, G.A., Vanrompay, Y., Kirsch Pinheiro, M., Berbers, Y., "A Hybrid Peer-to-Peer Solution for Context Distribution in Mobile and Ubiquitous Environments", **17th International Conference on Information Systems Development (ISD2008)**, Paphos, Cyprus, August 25-27, 2008, Spring, pp. 501-510.

Classement ERA : A

2006

Kirsch-Pinheiro, M.; Villanova-Oliver, M.; Gensel, J.; Martin, H. "A Personalized and Context-Aware Adaptation Process for Web-Based Groupware Systems", **4th Int. Workshop on Ubiquitous Mobile Information and Collaboration Systems (UMICS'06), CAISE'06 Workshop**, Luxembourg, June 5-6, pp. 884-898.

2005

Kirsch-Pinheiro, M., Villanova-Oliver, M., Gensel, J., Martin, H. "Context-Aware Filtering for Collaborative Web Systems: Adapting the Awareness Information to the User's Context", **20th ACM Symposium on Applied Computing (SAC 2005) - Web Technologies and Applications (WTA)**, Santa Fe, New Mexico, Mars 2005, ACM Press, 2005, pp. 1668-1673.

Classement ERA : B

Kirsch-Pinheiro, M., Villanova-Oliver, M., Gensel, J., Martin, H. "BW-M: A framework for awareness support in web-based groupware systems". **9th International Conference On Computer Supported Cooperative Work In Design (CSCWD)**, 2005, Coventry, UK, pp. 240-246.

Classement ERA: B

- 2004 Kirsch-Pinheiro, M., Villanova-Oliver, M., Gensel, J., Martin, H., "A Context-Based Awareness Mechanism for Mobile Cooperative Users", In: R. Meersman, Z. Tari, A. Corsaro (Eds.), LNCS 3292 - **On the Move to Meaningful Internet Systems 2004: OTM 2004 Workshops: OTM Confederated International Workshops and Posters, GADA, JTRES, MIOS, WORM, WOSE, PhDS, and INTEROP 2004**, octobre 2004. Springer-Verlag, 2004, pp. 9-10.
- Kirsch-Pinheiro, M., Gensel, J., Martin, H., "Representing Context for an Adaptive Awareness Mechanism", In: G.-J. de Vreede; L.A. Guerrero, G.M.Raventos (Eds.), **LNCS 3198 - X International Workshop on Groupware (CRIWG 2004)**, San Carlos, Costa Rica, septembre 2004, Springer-Verlag, 2004, pp. 339-348.
- Kirsch-Pinheiro, M., Gensel, J., Martin, H., "Awareness on Mobile Groupware Systems", In: A. Karmouch, L. Korba, E.R.M. Madeira (Eds.), **LNCS 3284 - 1st International Workshop on Mobility Aware Technologies and Applications (MATA 2004)**, Florianópolis, Brésil, octobre 2004. Springer-Verlag, 2004, pp. 78-87.
- 2003 Kirsch-Pinheiro, M., Villanova-Oliver, M., Gensel, J., Lima, J.V., Martin, H. "Providing a Progressive Access to Awareness Information". In: R. Meersman et al. (Eds.), **LNCS 2888 - CoopIS/DOA/ODBASE 2003**, Springer-Verlag, 2003, pp. 336-353.
Classement ERA : A
- 2002 Kirsch-Pinheiro, M., Lima, J.V., Borges, M.R.S., "A Framework for Awareness Support in Groupware Systems", **7th International Conference on Computer Supported Cooperative Work in Design**, Rio de Janeiro, Brazil, 2002. pp. 13-18.
Classement ERA : B
- 2001 Kirsch-Pinheiro, M., Lima, J.V., Borges, M.R.S. "Awareness em Sistemas de Groupware" (Awareness on Groupware Systems). In: **Proceedings of IDEAS'01**, Centre de Información Tecnológica (CIT), San Diego, Costa Rica, 2001. pp 323-335

Articles en conférences et ateliers nationaux avec comité de lecture

- 2019 Kirsch-Pinheiro, M., Souveyet, C. « Le Rôle des Ressources dans l'Évolution des Systèmes d'Information », **Congrès INFORSID 2019 : Informatique des Organisations et Systèmes d'Information et de Décision 2019**: 85-97
- Souveyet, C., Villari, M., Steffanel, L.A., Kirsch Pinheiro, M. « Une approche basée sur les MicroÉléments pour l'Évolution des Systèmes d'Information ». **Atelier "Evolution des SI : vers des SI Pervasifs ?"**, **INFORSID 2019**, Paris, Jun 2019, Paris. Disponible sur : <https://evolution-si.sciencesconf.org/274949>
- Steffanel, L.A., Kirsch-Pinheiro, M. « Accès aux Données dans le Fog Computing : le cas des dispositifs de proximité ». **Conférence d'informatique en Parallélisme, Architecture et Système (Compas 2019)**, Jun 2019, Anglet, France

- 2016 Jaffal, A., Le Grand, B., Kirsch-Pinheiro, M., « Extraction de connaissances dans les Systèmes d'Information Pervasifs par l'Analyse Formelle de Concepts », **Extraction et Gestion des Connaissances (EGC 2016)**, Revue des Nouvelles Technologies de l'Information, RNTI-E-30, 2016, pp 291-296.
- Steffenel, L.A., Kirsch-Pinheiro, M., « Stratégies Multi-Échelle pour les Environnements Pervasifs et l'Internet des Objets », **11èmes Journées Francophones Mobilité et Ubiquité (UBIMOB 2016)**. July 5, 2016. Lorient, France. Disponible sur https://ubimob2016.telecom-sudparis.eu/files/2016/07/Ubimob_2016_paper_6.pdf
- 2013 Kirsch Pinheiro, M., Le Grand, B., Souveyet, C., Najar, S., « Espace de Services : Vers une formalisation des Systèmes d'Information Pervasifs », **XXXIème Congrès INFORSID 2013 : Informatique des Organisations et Systèmes d'Information et de Décision**, 29-31 Mai 2013, Paris : France, pp. 215-223.
- Rychkova, I., Kirsch-Pinheiro, M., Le Grand, B. « CAPE: Context-Aware Agile Business Process Engine », **Workshop GT EASY-DIM 2013 - Vers l'Ingénierie d'Entreprise de demain : les enjeux d'une maquette numérique de l'entreprise**, abstract paper, Mercredi 22 mai 2013, Paris, pp. 8-11
- 2012 Najar, S. Kirsch-Pinheiro, M., Steffenel, L. A., Souveyet, C., « Analyse des mécanismes de découverte de services avec prise en charge du contexte et de l'intention ». In : Philippe Roose & Nadine Rouillon-Couture (dir.), **8èmes Journées Francophones Mobilité et Ubiquité (UbiMob 2012)**, June 4-6, 2012, Anglet, France. Cépaduès Editions, pp. 210-221. ISBN: 978.2.36493.018.6
- Najar, S. Kirsch-Pinheiro, M., Souveyet, C., « Mécanisme de prédiction dans un système d'information pervasif et intentionnel », In : Philippe Roose & Nadine Rouillon-Couture (dir.), **8èmes Journées Francophones Mobilité et Ubiquité (UbiMob 2012)**, June 4-6, 2012, Anglet, France. Cépaduès Editions, pp. 146-157. ISBN: 978.2.36493.018.6
- 2009 Najar, S., Kirsch Pinheiro, M., Le Grand, B., Souveyet, C., « Systèmes d'Information Pervasifs et Espaces de Services : Définition d'un cadre conceptuel », **UbiMob 2013 : 9èmes journées francophones Mobilité et Ubiquité**, 5-6 juin 2013 Nancy (France), sciencesconf.org/ubimob2013:19119
- 2008 Gensel, J., Villanova-Oliver, M., Kirsch-Pinheiro, M., « Modèles de contexte pour l'adaptation à l'utilisateur dans des Systèmes d'Information Web collaboratifs », **8èmes Journées Francophones d'Extraction et Gestion des Connaissances (EGC'08), Atelier sur la Modélisation Utilisateur et Personnalisation d'Interfaces Web**, Article invité, Sophia-Antipolis, France, janvier 2008.
- 2005 Kirsch-Pinheiro, M. « Adaptation contextuelle de l'information de conscience de groupe dans les SIW collaboratifs ». **Actes du XXIIème Congrès Informatique des organisations et systèmes d'information et de décision – INFORSID 2005**, Grenoble, France, 2005. pp. 285-300.
- Kirsch-Pinheiro, M.; Villanova-Oliver, M.; Gensel, J.; Martin, H. « Une formalisation du cotexte dans les environnements collaboratifs nomades ». **Actes des Deuxièmes Journées Francophones : Mobilité et Ubiquité 2005 (UbiMob'05)**, Grenoble, France, 2005. pp. 1-8.

Rapports & Mémoires

- | | |
|------|---|
| 2013 | Kirsch Pinheiro, M., Requirements for Context-aware MapReduce on Pervasive Grids, PER-MARE Deliverable D3.1, 09/07/2013 |
| 2006 | Kirsch Pinheiro, M. Adaptation contextuelle et personnalisée de l'information de conscience de groupe au sein des Systèmes d'Information Coopératifs. Thèse en informatique, Université Joseph-Fourier, Grenoble, 2006. |
| 2001 | Kirsch Pinheiro, M. Mecanismo de Suporte a Percepcao em Ambientes Cooperativos. Dissertation PPGC/UFRGS, Porto Alegre, Brasil, 2001. |
| 1999 | Kirsch Pinheiro, M.; Lima, J.V. Edição Cooperativa de Hiperdocumentos na WWW. Rapport technique , PPGC/UFRGS, Porto Alegre, Brasil, 1999. |
| 1997 | Kirsch Pinheiro, M.; Augustin, I. Compilação Simulada na Web. Rapport technique, Curso de Informática da UFSM, Santa Maria, Brasil, 1997. |