

Cohérence et qualité des informations de contexte en environnement pervasif

Zied Abid, Sophie Chabridon, et Denis Conan
Institut Télécom; Télécom & Management SudParis; CNRS UMR SAMOVAR
9 rue Charles Fourier, 91011 Évry, France

Prenom.Nom@institut-telecom.fr

RESUME

La gestion de contexte en environnement pervasif devient un élément clé pour le développement de nouvelles applications. Cependant, les propriétés des systèmes pervasifs ont un impact sur la cohérence des informations de contexte. La composition d'observations réparties provenant de différentes sources peut introduire des incohérences potentielles. De plus, le contexte est un élément dynamique évoluant fréquemment et affectant en continu les applications sensibles au contexte. Les méthodes traditionnelles de gestion de cohérence ne sont donc pas toujours adaptées pour un système pervasif. Cet article propose de combiner deux approches complémentaires. D'une part, la prise en compte de la qualité des informations de contexte au sein de notre intergiciel de gestion de contexte COSMOS permet de traiter des conflits syntaxiques au sein de contextes de bas niveau. D'autre part, l'utilisation d'ontologies permet de déterminer l'équivalence sémantique de deux contextes pour lever des conflits de plus haut niveau.

MOTS CLÉS

Cohérence, contexte, qualité de contexte, gestion de contexte répartie, ontologie

ABSTRACT

Context management in pervasive environments becomes a key element for the development of new applications. However, the intrinsic properties of pervasive systems impact the consistency of context information. The composition of distributed observations collected from different sources may cause the introduction of potential inconsistencies. Besides, context information is highly dynamic and evolves continuously affecting the execution of context-aware applications. Traditional solutions for consistency management in distributed environments are therefore not well suited to pervasive environments. This paper presents our contribution combining two complementary approaches. We integrate quality of context (QoC) metadata in our COSMOS context management framework to handle syntactic conflicts in low-level contexts and we rely on ontologies for determining whether higher-level con-

texts semantically match.

KEYWORDS

Consistency, context, quality of context, distributed context management, ontology

Categories and Subject Descriptors

C.2.4 [COMPUTER-COMMUNICATION NETWORKS]: Distributed Systems; D.4.7 [OPERATING SYSTEMS]: Organization and Design

General Terms

Design, Algorithms

1. INTRODUCTION ET MOTIVATIONS

Ces dernières années, la gestion de contexte en environnement pervasif suscite l'intérêt d'une communauté de recherche de plus en plus importante. Cet article propose de faire le point sur différents aspects de la cohérence des informations de contexte. Ces informations sont par nature imparfaites puisque d'une part elles résultent d'un modèle représentant une abstraction de la réalité ne pouvant être exhaustive et d'autre part elles dépendent de dispositifs tels que des capteurs ayant des limitations physiques intrinsèques. Les informations de contexte ont donc tendance à être incorrectes car elles ne reflètent pas exactement l'état réel de l'entité observée, incomplètes lorsque certains aspects du contexte sont manquants, ou encore ambiguës si plusieurs valeurs sont collectées et ne correspondent pas entièrement entre elles [?]. Par exemple, deux dispositifs de localisation distincts peuvent fournir des valeurs correspondant à des régions se recouvrant, ayant des niveaux de précision (granularité) différents ou bien encore étant incohérentes si elles présentent des informations contradictoires. Lorsque les informations de contexte collectées sont imparfaites et incertaines, le risque existe de baser une décision sur des informations erronées [?, ?]. De plus, le coût est très élevé pour raisonner sur des informations incertaines vu la complexité des solutions à mettre en œuvre [?, ?]. Ces imperfections doivent être réduites le plus possible par des méthodes de vérification de la cohérence ou de résolution des conflits, cependant elles ne peuvent être complètement levées. Il est donc important de prévoir les mécanismes nécessaires pour limiter l'impact que peuvent avoir ces imperfections, notamment en permettant au système de prendre en compte leur existence même. C'est ainsi que la qualité des informations de contexte devient primordiale pour les applications sensibles au contexte.

La notion de qualité de contexte (QoC) a été définie pour la première fois dans le cadre de la gestion de contexte par [?] en tant que concept à part entière pour décrire la qualité des informations utilisées comme informations de contexte, indépendamment

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CDUR 2009 11 Sept.

Copyright 2009 ACM X-XXXXX-XX-X/XX/XX ...\$5.00.

du processus ou du composant matériel ayant fourni ces informations. Cette définition a été complétée par [?] en ajoutant la notion de valeur pour une application spécifique. La QoC apparaît donc comme un ensemble de méta-données caractérisant une information de contexte. Elle permet de gérer des priorités entre toutes les informations de contexte collectées. Ceci est exploité par [?] qui présente un exemple d'application pour la gestion de situations d'urgence. Des sauveteurs sont présents sur le terrain et utilisent des appareils sans fil pour communiquer, collecter des informations et prendre des photos. Par ailleurs, les responsables disposent d'images satellite prises sur le lieu du sinistre. En tenant compte de la qualité des informations de contexte pouvant être déduites des photos, un responsable pourra privilégier les photos envoyées par un sauveteur ayant un niveau de confiance élevé indiquant qu'il est proche de la zone du sinistre et dont l'appareil de prise de photo possède une grande précision, par rapport à des images satellites.

Nous présentons dans la section suivante notre plate-forme de gestion de contexte COSMOS, comment nous prenons en compte la qualité de contexte au sein de cette plate-forme et comment cela permet de gérer une partie des conflits possibles au sein de contextes de bas niveau. Nous considérons ces conflits comme étant de type syntaxique car ils concernent des contextes que l'on peut identifier comme équivalents par une simple comparaison de leur structure. Lorsqu'une comparaison sémantique devient nécessaire, nous faisons appel à des modèles à base d'ontologies. Nous décrivons ainsi dans la section 3 comment la plate-forme COSMOS peut être complétée par des algorithmes prenant en compte la sémantique pour lever des conflits de plus haut niveau. La section 4 positionne notre proposition par rapport à des travaux connexes. Nous concluons dans la section 5 en donnant des perspectives à nos travaux.

2. PRISE EN COMPTE DE LA QUALITÉ DE CONTEXTE

Dans cette section, nous décrivons comment nous prenons en compte la qualité de contexte au sein de notre plate-forme COSMOS. Nous introduisons tout d'abord dans la section 2.1 le canevas logiciel COSMOS, puis nous détaillons comment la QoC est intégrée à COSMOS dans la section 2.2 incluant la définition de contrats d'observation et de QoC et la sélection des sources de contexte.

2.1 Présentation de COSMOS

Nous présentons dans cette partie le canevas logiciel COSMOS (Context entitieS coMpositiOn and Sharing) [?, ?] que nous utilisons pour construire des gestionnaires de contexte. Les trois principes sur lesquels repose COSMOS sont la séparation entre les activités de collecte et de synthèse des données de contexte, l'organisation des politiques de gestion de contexte en assemblages de composants logiciels dans une hiérarchie avec partage et l'utilisation systématique de patrons de conception. Par rapport aux approches à base de moteurs d'inférences utilisées pour répondre au problème de la composition de contexte, l'approche proposée est performante : elle contrôle et minimise l'utilisation des ressources pour son propre fonctionnement, ce qui est essentiel pour certaines applications et terminaux mobiles.

Dans COSMOS, le contexte est constitué de différentes catégories d'entités observables (ressources logicielles telles que les ressources système et les préférences des utilisateurs, et ressources matérielles telles que les capteurs), des rôles entre ces entités dans le contexte et des relations entre ces entités. La gestion de toutes les informations qui décrivent le contexte d'exécution des applications est centralisée dans un gestionnaire de contexte. Sur un termi-

nal, son rôle est de construire une vue unifiée de l'environnement d'exécution, et de proposer aux applications des mécanismes de description et d'identification de situations d'adaptation. Dans COSMOS, les politiques sont décomposées en unités à grain fin appelées nœuds de contexte qui sont des composants logiciels. Les politiques de gestion de contexte ont été introduites dans COSMOS afin d'identifier les situations d'adaptation pour lesquelles une réaction de l'application est attendue. L'utilisation de Fractal¹ et Fractal ADL pour la réalisation de COSMOS autorise une conception avec des composants à granularité fine.

Le nœud de contexte est un concept de base de COSMOS. Il correspond à une information de contexte modélisée par un composant et possède des propriétés qui définissent son comportement :

- passif ou actif : un nœud passif est utilisé par des activités extérieures au nœud qui l'interrogent pour obtenir une information. Un nœud actif est équipé d'une activité pour exécuter une tâche donnée ;
- observation ou notification : les communications peuvent s'opérer du bas vers le haut ou du haut vers le bas de l'arborescence de nœuds de contexte. Les notifications correspondent aux messages envoyés par les nœuds à leurs parents (du bas vers le haut), tandis que les observations sont déclenchées par un nœud parent. Un nœud de l'arborescence peut être à la fois observateur et notificateur actifs ;
- passant ou bloquant : un nœud passant propage les observations et les notifications. Dans le cas bloquant, le nœud observé fournit l'information de contexte qu'il détient sans observer les nœuds enfants, et le nœud notifié modifie son état interne sans notifier les nœuds parents ;
- nommage : les noms des nœuds de contexte sont uniques pour permettre les parcours dans le graphe et les configurations.

Un nœud récupère des informations de contexte de nœuds enfants de la hiérarchie et infère une information de plus haut niveau d'abstraction. Le traitement correspondant est effectué dans l'opérateur de contexte. Le cycle de vie des nœuds de contexte enfants est contrôlé par les nœuds de contexte parents. Pour la gestion des activités, les nœuds de contexte actifs enregistrent leurs activités auprès d'un gestionnaire d'activités. Ainsi, le gestionnaire d'activités, lui-même paramétrable, peut créer une activité par traitement (observation ou notification) ou bien une activité par nœud de contexte ou encore une activité pour tout ou partie de la hiérarchie. Pour la consommation mémoire, un gestionnaire de messages gère des réserves de messages et autorise des duplications par référence ou par valeur.

2.2 Gestion de la QoC avec COSMOS

Même s'il n'existe pas de consensus sur les critères de qualité à utiliser, un certain nombre revient dans plusieurs travaux du domaine ([?, ?, ?, ?]).

- exactitude (ou estimation): décrit dans quelle mesure la valeur de l'information de contexte représente la réalité. L'exemple typique est celui de la localisation. Son exactitude dépend du mécanisme utilisé. Avec un récepteur GPS, une estimation d'environ 4 mètres est attendue, alors qu'un réseau cellulaire atteint des estimations d'au mieux 500 mètres en zone urbaine;

¹<http://fractal.objectweb.org/>

- probabilité de correction: liée à la présence d'erreurs non intentionnelles. La source de l'information indique au travers de la probabilité de correction combien de fois l'information de contexte qu'elle fournit risque d'être erronée en raison de problèmes internes;
- fidélité: concerne le point de vue du fournisseur de contexte pour qualifier les différentes sources d'information qu'il utilise. Par exemple, le fournisseur de contexte A envoie une information de contexte à B. A indique que la probabilité de correction de cette information est de 100%. Cependant, B recevant souvent des informations incorrectes de la part de A, il transmettra les informations à ses clients avec une mise en garde sur le fait que la source de l'information n'est pas entièrement digne de confiance;
- résolution, ou granularité (appelée précision par [?]). Un fournisseur de contexte annonce que la température d'une pièce est de 17 °C. Ceci est vrai en moyenne, mais pas nécessairement à proximité du radiateur présent dans la pièce. La valeur pourra ainsi être complétée par des bornes min et max indiquant le niveau de détail de la mesure: 17 ± 1 °C.
- fraîcheur (ou âge des informations [?]): indique si les informations sont suffisamment récentes pour représenter la situation présente.
- portée géographique: indique la zone géographique dans laquelle l'information peut être utilisée. La température fournie par une station météo extérieure n'est pas la plus adaptée pour contrôler le chauffage central à l'intérieur d'un bâtiment;

Deux approches peuvent être envisagées pour intégrer la qualité de contexte dans un canevas logiciel [?]. Une approche homogène permet d'associer à chaque information de contexte une mesure numérique de sa qualité. Ceci facilite les comparaisons et la composition des mesures de qualité, mais n'est pas toujours possible car la qualité ne peut pas toujours être déterminée de manière systématique. Une approche hétérogène part du principe qu'il ne peut y avoir une mesure unique et une représentation standard du degré de certitude d'une information de contexte. Chaque type d'information possède ainsi son propre ensemble de mesures de qualité. C'est l'approche que nous suivons dans COSMOS au sein duquel nous mettons en place une gestion de QoC souple et flexible, fonction de la nature des informations de contexte et des capacités des sources de contexte. Ceci permet également une meilleure optimisation des performances en modulant le coût de calcul des critères de qualité en fonction de ce qui est effectivement nécessaire.

2.2.1 Contrats d'observation et de QoC

Par analogie avec ce qui existe pour la qualité de service (QoS), [?] préconise la négociation de contrats de QoC. Nous prévoyons d'intégrer de tels contrats avec les contrats d'observation de COSMOS. Un contrat d'observation permet à l'application d'exprimer ses besoins concernant les informations de contexte qui sont pertinentes pour elle ainsi que les situations d'adaptation auxquelles elle est sensible et qu'elle souhaite détecter. L'observation collectée peut ainsi être qualifiée pour améliorer cette observation ainsi que les prises de décision qui en découlent. Les informations telles que la localisation, la précision, la fiabilité, la fraîcheur peuvent être exprimées dans les contrats d'observation et être collectées et traitées par la chaîne d'observation. COSMOS permet de considérer le gestionnaire de contexte comme un service d'observation

pour les composants logiciels qui annoncent par contrat leurs exigences d'observation.

Un utilisateur de contexte peut ensuite vérifier que les informations de contexte fournies par les sources sont d'une qualité au moins égale au minimum défini dans le contrat. Il faut également prévoir la re-négociation des contrats permettant les modifications nécessaires en raison de la forte dynamique d'un environnement pervasif.

Ainsi, la demande exprimée sur un contrat d'observation d'une bande passante peut être enrichie par des caractéristiques liées à la fraîcheur de l'information (différence entre les dates de livraison et de collecte effective) ou par la précision (en % par exemple). La contrainte sur la qualité des informations d'observation peut ensuite être utilisée lors de la recherche d'une source d'observation acceptable.

2.2.2 Sélection des sources de contexte

Lorsque plusieurs sources de contexte fournissent des informations sur une même entité observée, il est nécessaire de prévoir la manière dont doit se faire la sélection de sources de contexte à privilégier. Ceci repose en partie sur les contrats d'observation qui permettent de choisir les sources de contexte en fonction du niveau de QoC qu'elles peuvent garantir. Si plusieurs sources sont capables de satisfaire le contrat, il faut ensuite choisir des règles simples pour éviter une diminution des performances par des calculs d'inférence inutiles. En complément, nous évoquons dans la section 3 comment des ontologies peuvent aider à découvrir et sélectionner les sources de contexte.

D'autre part, dans [?], les auteurs proposent de procéder à une rétropropagation des erreurs à la manière des réseaux neuronaux depuis les utilisateurs de contexte jusqu'aux sources dans les cas où un contexte s'avère non pertinent ou inutile, où un contexte est très peu utilisé en demandant à ce qu'il soit fourni moins souvent, ou bien encore lorsqu'il y a duplication dans le cas où la même information de contexte est déjà fournie par un autre chemin.

3. PRISE EN COMPTE DE LA SÉMANTIQUE DES INFORMATIONS DE CONTEXTE

Différencier les informations de contexte selon leur qualité peut ne pas suffire à lever tous les conflits pouvant se produire. Nous évoquons dans cette section l'apport des ontologies pour comparer des contextes de haut niveau d'abstraction et pouvoir établir des équivalences sémantiques (*semantic matching*).

Pour prendre en compte la sémantique des informations de contexte, l'utilisation d'ontologies apparaît très populaire dans les travaux de recherche récents. Une ontologie est une description sémantique, structurée et formelle des concepts d'un domaine et leurs inter-relations [?], formée par un ensemble de classes, de relations entre ces classes (héritage ou sémantique), de propriétés décrivant les classes et d'axiomes.

[?] propose une méthode de vérification d'équivalence sémantique permettant de dépasser la seule prise en compte de l'intersection entre deux concepts qui apparaît comme insuffisante pour détecter un conflit. Une définition formelle de la substituableté entre deux concepts est élaborée. [?] exploite l'interface de vérification de cohérence de Jena² qui est un outil du web sémantique permettant de manipuler facilement une ontologie. Un conflit entre deux concepts d'une ontologie est détecté lorsqu'une assertion sur l'ontologie ne peut pas être validée. Une fonction détec-

²<http://jena.sourceforge.net>

minant la fréquence relative d'un contexte est également utilisée pour associer une valeur (un poids) à chaque contexte : plus un contexte a été utilisé fréquemment récemment, plus sa valeur est grande. Lorsqu'un conflit entre deux contextes est détecté, celui avec la plus faible fréquence relative est ignoré.

Malgré leur popularité croissante, les ontologies restent limitées pour prendre en compte les incertitudes liées aux informations de contexte [?]. Nous considérons qu'il n'est donc pas envisageable de reposer exclusivement sur des moteurs d'inférence ontologiques pour la gestion de contexte, également pour des raisons de performances. Nos travaux en cours concernent l'intégration de COSMOS avec des moteurs ontologiques sur deux niveaux, en amont et en aval de la gestion de contexte. En amont de la gestion de contexte, une comparaison sémantique peut être effectuée lors de la découverte des sources de contexte pour la mise en place d'une politique de contexte. À partir des contrats d'observation demandés par l'application, il s'agit d'identifier quelles sources sont capables de répondre aux besoins en respectant le contrat en termes d'observations attendues et de QoC. Lorsque plusieurs sources sont candidates, des comparaisons sémantiques permettent de décider quelle source choisir. D'autre part, des ontologies peuvent intervenir en aval de la gestion de contexte au moment où des contextes de haut niveau sont inférés par COSMOS. L'utilisation de la QoC dans COSMOS permet de gérer une partie des incohérences syntaxiques survenant dans les contextes de bas niveau et dans le processus d'inférence de contexte. Les conflits résiduels présents dans les contextes de haut niveau peuvent ensuite être traités par des ontologies avec des comparaisons sémantiques.

4. ÉTUDE DE CAS

Nous nous basons sur une étude de situations d'adaptation [?] faisant une synthèse de plusieurs scénarios de gestion de contexte (Active Campus [?], SOCAM [?], Gaia [?], CARISMA [?], EgoSpaces [?] and Runes [?]).

Le scénario retenu simule les activités quotidiennes de personnes vivant dans une ville. Une personne se lève le matin vers 7h, puis quitte son domicile et part travailler, et ensuite soit rentre chez elle pour y passer la soirée, soit va dans un lieu de distraction, et éventuellement dans un autre jusqu'au moment de rentrer dormir chez elle. Le cycle reprend le jour suivant. Trois types de lieux sont définis : domicile, bureau et lieu de distraction. Les profils des personnes (sexe, âge, préférence...) sont paramétrables. Une personne part travailler en moyenne à 8h, avec une amplitude de variation de 2 heures. La durée d'une journée de travail est normalement de 8 heures; l'heure du coucher est prévue 9 heures avant le lever, mais varie suivant les activités sociales en soirée. Huit activités différentes sont donc possibles : endormi, en train de travailler, marchant dans la rue, allant au travail, à la maison, allant à une soirée, en train de s'amuser, rentrant à la maison. D'autres paramètres concernent les goûts culinaires des personnes, le type de travail, les préférences en matière d'activités sociales.

Un ensemble de 16 situations d'adaptation est envisagé dans [?] à partir des travaux de la littérature. Nous nous intéressons dans cet article à quatre situations représentatives des aspects de cohérence.

1. Le service d'aide-mémoire permet de recevoir des messages de rappel pour ne pas oublier un événement ou un rendez-vous. Autoriser le service d'aide-mémoire de Marie à envoyer une alerte sonore lorsque Marie est à l'extérieur d'un bâtiment.
2. Autoriser le service d'aide-mémoire de Marie à envoyer une alerte par vibreur lorsque Marie est au bureau.

3. Autoriser le service d'aide-mémoire de Marie à envoyer une alerte silencieuse par affichage d'un message clignotant lorsque Marie est en train d'utiliser intensivement son assistant personnel.
4. Marie ayant tendance à travailler de manière addictionnelle une fois rentrée à la maison, elle a souscrit à un service d'alerte pour lui rappeler de changer d'activité selon certaines conditions : par exemple lorsque le signal WIFI devient trop faible ou bien lorsque le nombre d'heures de travail dépasse un seuil prédéfini.

Les situations 1 à 3 sont dérivées de Carisma [?] où les conflits sont gérés par un système d'enchères secrètes. Dans ce système, un conflit apparaît lorsque deux politiques différentes peuvent être envisagées dans un même contexte pour la fourniture d'un service. Une hypothèse préalable impose qu'un service ne peut être mis en œuvre que selon une seule politique. Deux types de conflits sont distingués. Un conflit intra-profil survient au sein du profil de contexte d'une même application. Un conflit inter-profil a lieu entre deux applications s'exécutant sur deux terminaux différents. Si Marie est au bureau et se sert intensivement de son assistant personnel, les conditions des situations 2 et 3 sont remplies et peuvent donc être appliquées, traduisant un conflit entre ces 2 politiques. Dans Carisma, une fonction d'utilité indiquant la priorité (le prix) qu'une application accorde aux ressources critiques, batterie par exemple, et à certains critères de qualité de service, tels que le respect de la vie privée, doit être fournie par chaque application. Cette fonction d'utilité s'apparente à un contrat d'observation tel que nous le proposons dans COSMOS et permet à l'application de spécifier quels sont ses besoins en informations de contexte. Pour le choix de la décision d'adaptation, COSMOS exploite la QoC associée aux informations de contexte lorsqu'elle est présente. La fraîcheur permettra ici de déterminer si l'information de contexte est suffisamment récente et de donner une priorité à l'une ou l'autre des situations. Si Marie travaille intensivement sur son PDA depuis peu de temps, une alerte visuelle peut être préconisée.

Un exemple de politique de contexte définie avec COSMOS pour la situation 4 est présenté sur la figure 1. Les nœuds du graphe sont repérés par leur nom, chacun indiquant intuitivement le type d'opérateur du nœud de contexte. Les arcs du graphe représentent les relations d'inclusion, y compris les partages de nœuds enfants par plusieurs nœuds parents. À côté d'un nœud, figurent ses propriétés : actif/passif, bloquant/passant, etc. Dans notre exemple, la plupart des nœuds actifs observent ; seuls les nœuds détectant les changements d'états (« détecteur de changement des préférences utilisateur » et « détecteur de connectivité ») et de décision notifient ces changements vers l'application. Les nœuds sans flèche sont passants pour l'observation et la notification. Dans le cas du nœud gestionnaire du réseau WiFi (en bas à droite de la figure), le nœud est bloquant pour l'observation car il gère lui-même une tâche d'observation. Le nœud gestionnaire du réseau WiFi permet de regrouper toutes les informations (qualité du lien, débit et « le débit est-il variable ? ») pour éviter que les trois nœuds parents ne multiplient les accès aux informations système et ne provoquent un grand nombre d'appels système.

Pour la situation 4, l'utilisation de la probabilité de correction, issue des informations de QoC, permet de déterminer si l'assertion indiquant quelle est l'activité de Marie sur ordinateur est fiable. Si Marie utilise en fait deux activités en même temps sur son ordinateur, une application de traitement de texte et une activité de jeux, ceci peut être détecté comme une situation de conflit. De telles situations peuvent alors être modélisées par une ontologie, avec des règles associées indiquant les contraintes sémantiques à respecter

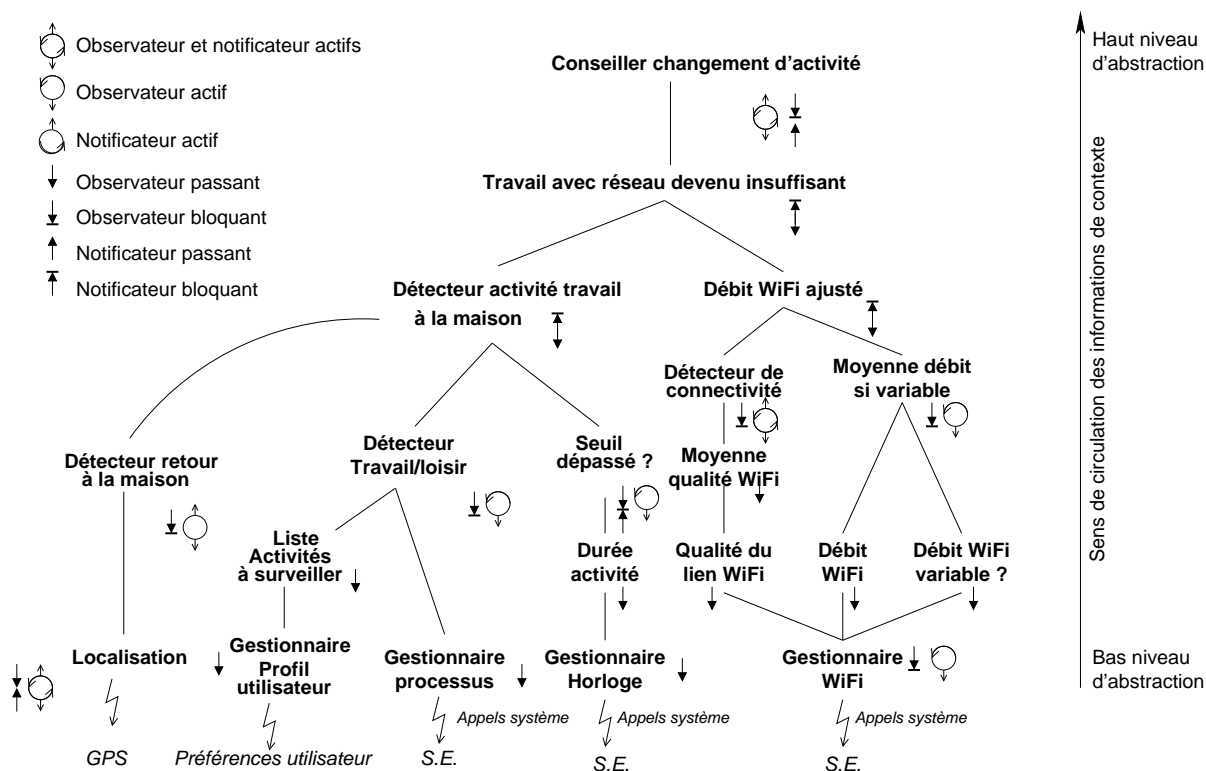


Figure 1: Composition de nœuds de contexte pour la situation 4

exprimées en logique des prédicats de la forme (*sujet, prédicat, objet*).

5. TRAVAUX CONNEXES

Des états de l'art récents sur les systèmes sensibles au contexte ne traitent que succinctement des aspects qualité : [?] mentionne la nécessité pour un serveur de contexte de préciser les paramètres de qualité de service réseau, ce qui diffère de la QoC, et [?] indique que parmi tous les intergiciels passés en revue, seul MiddleWhere [?] traite explicitement de la qualité de contexte, mais limitée aux informations de localisation. Des travaux de recherche prenant en compte la qualité de contexte dans le cadre de l'intelligence ambiante commencent à apparaître mais n'en sont qu'aux prémices ([?, ?, ?, ?, ?, ?, ?]) et la proposition de solutions intergicielles flexibles et performantes reste indispensable.

Le système CARISMA [?] est l'un des premiers travaux de recherche à prévoir des mécanismes spécifiques pour la résolution des incohérences de contexte. Son protocole d'enchères secrètes est original et permet de déterminer dynamiquement quelle politique adopter. Il repose sur des fonctions d'utilité définies par les applications utilisatrices de contexte qui doivent ainsi déterminer de manière très précise des compromis entre coût des ressources et bénéfices attendus. Il serait intéressant d'ajouter un mécanisme permettant d'amender et modifier si nécessaire ces fonctions d'utilité, les besoins des applications pouvant varier au cours du temps.

Les travaux sur CML (Context Modelling Language) [?] permettent la gestion des incertitudes présentes dans les informations de contexte en prévoyant la définition de plusieurs alternatives conflictuelles. Ceci permet de prendre en compte un grand nombre

d'observations et de retarder la prise de décision le plus possible. L'environnement changeant dynamiquement en permanence, ceci permet aussi de pouvoir vérifier ultérieurement si le conflit a toujours lieu.

6. CONCLUSION ET PERSPECTIVES

Dans cet article, nous décrivons notre position vis-à-vis de la gestion de la cohérence des informations de contexte en environnement pervasif. De par les spécificités de ces informations qui sont par nature imparfaites, il est indispensable de fournir des mécanismes permettant de maîtriser le niveau de qualité du contexte. Notre contribution est représentée par le canevas logiciel à base de composants COSMOS prenant en compte la notion de qualité de contexte et permettant de gérer une partie des incohérences et conflits syntaxiques pouvant survenir au cours du processus d'inférence de contextes de haut niveau. Pour traiter les conflits d'ordre sémantique, nous proposons d'associer COSMOS avec un moteur ontologique, sur deux niveaux : en amont de la gestion de contexte, pour la découverte des sources de contexte à partir des contrats d'observation demandés par l'application, et en aval, pour lever certains cas de conflit par des comparaisons sémantiques. Nous présentons également une étude de cas illustrant l'utilisation de COSMOS. Nous travaillons actuellement à la mise en œuvre de COSMOS avec les éléments proposés et prévoyons d'évaluer notre approche sur plusieurs scénarios applicatifs.

7. RÉFÉRENCES