



HAL
open science

Connaissance du Contexte, Confidentialité et Accès Mobiles : une Approche Web Sémantique et Multi-agents

Fabien Gandon, Norman Sadeh

► **To cite this version:**

Fabien Gandon, Norman Sadeh. Connaissance du Contexte, Confidentialité et Accès Mobiles : une Approche Web Sémantique et Multi-agents. Journées Francophones: Mobilité et Ubiquité 2004, Jun 2004, Sophia Antipolis, France. hal-01149946

HAL Id: hal-01149946

<https://inria.hal.science/hal-01149946>

Submitted on 7 May 2015

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.

Connaissance du Contexte, Confidentialité et Accès Mobiles : une Approche Web Sémantique et Multi-agents

Fabien L. Gandon^{1,2}

¹INRIA - ACACIA Lab.

2004 Route des Lucioles, B.P. 93

06902 SOPHIA-ANTIPOLIS

Tel. (33)/(0)4-92-38-77-88

Fabien.Gandon@inria.fr

Norman M. Sadeh²

²Mobile Commerce Lab.

Carnegie Mellon University, School of Computer Science

5000 Forbes Avenue, Pittsburgh, PA 15213-3891, USA

Tel. (1)-412-268-8144

sadeh@cs.cmu.edu

RESUME

Nous présentons une vue globale des travaux effectués dans *myCampus*, un environnement ouvert utilisant la connaissance du contexte pour faciliter l'accès mobile à des services de l'intranet d'un campus tout en préservant la vie privée des utilisateurs.

ABSTRACT

We give an overview of the work achieved in *myCampus*, an open environment relying on context-awareness to assist mobile accesses to services available on the intranet of a campus, while enforcing privacy preferences of the users.

Categories and Subject Descriptors

H.3.5 [Online Information Services]: Web-based services

General Terms

Experimentation, Security, Human Factors, Design

Mots-clefs

Mobilité, Contexte, Confidentialité, Web sémantique, Agents.

Keywords

Mobility, Context, Privacy, Semantic Web, Agents.

1. LE PROJET MYCAMPUS

1.1 Accès mobiles aux services en ligne

Dans le projet *myCampus*, nous concevons une architecture distribuée facilitant l'accès aux services d'un campus au quotidien. L'environnement (figure 1) est ouvert et basé sur les technologies des systèmes multi-agents et du Web sémantique [1] pour réaliser une plate-forme d'accès mobile [16] à des services en ligne [9,15]. Les utilisateurs souscrivent à différents types d'*agents-services* qui répondent à une attente et aident à différentes activités (ex : établir une réunion, filtrer des messages). Pour opérer, chaque agent a besoin d'informations sur son propriétaire et éventuellement sur d'autres utilisateurs. L'accès à une information sur un utilisateur est contrôlé par son agent *e-Wallet* et les règles de confidentialité qu'il contient. Les agents ne sont pas limités aux ressources personnelles des utilisateurs. Ils accèdent également à des services Web publics, à des documents Web, à des ontologies et des

descriptions du Web sémantique, *etc.* Dans *myCampus*, ceci inclut l'accès à une variété de ressources et services tels que des services Web de localisation, de météorologie, *etc.* Dans *myCampus* l'accès au système se fait à partir d'assistants personnels numériques (PDA) reliés au réseau sans-fil de l'université de Carnegie Mellon. Cependant, cette architecture fonctionne aussi pour des scénarios à postes fixes et plus généralement dans des environnements où les utilisateurs se connectent par différents canaux et dispositifs d'accès. Les informations sur le dispositif et le canal peuvent être traitées comme des attributs du contexte et rendues disponibles via le *e-Wallet*. Nous avons suivi une architecture FIPA [7] incluant en particulier des *Agents annuaires* qui permettent à un agent/utilisateur à la recherche d'un service d'obtenir une liste d'agents potentiellement fournisseurs de ce service. Des *Agents d'Interaction* sont responsables des échanges avec l'utilisateur ; ils gèrent les sessions de connexion et les interactions avec d'autres agents. Nous nous concentrerons sur des scénarios impliquant des utilisateurs individuels, mais cette architecture s'étend à des scénarios où les utilisateurs sont des organismes entiers ayant chacun un ou plusieurs *e-Wallet*. Pour plus de détails sur l'architecture et sa mise en oeuvre, se reporter à [9].

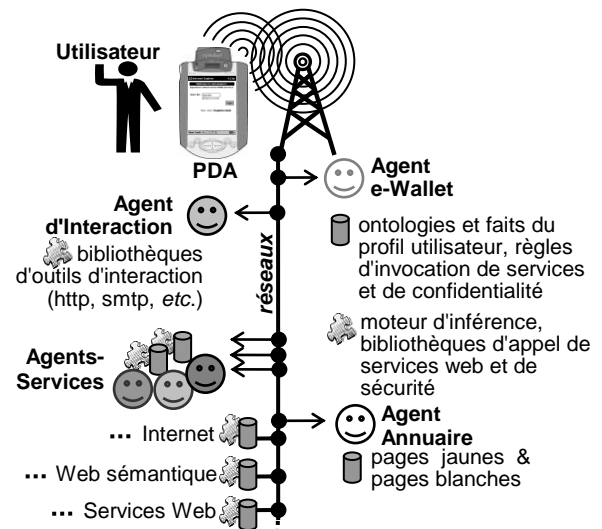


Figure 1. Vue d'ensemble du système *myCampus*.

1.2 Principe du *e-Wallet*

Le *e-Wallet* est un élément central de notre architecture Web sémantique pour la prise en compte du contexte et le respect de la confidentialité. Il représente une interface sémantique unifiée et sécurisée pour l'accès aux ressources personnelles d'un utilisateur

et permet ainsi aux *agents-services* de les mobiliser. Le *e-Wallet* n'est pas une archive statique. Il est capable d'obtenir des connaissances manquantes en faisant appel à des services externes. Il est aussi capable de contrôler les droits d'accès très précisément. Dans le *e-Wallet*, la connaissance de l'utilisateur se divise en 4 catégories.

La *connaissance statique* qui inclut typiquement des connaissances indépendantes du contexte (ex : nom, email, etc.).

La *connaissance dynamique* sensible au contexte de l'utilisateur (ex : je ne veux pas recevoir de messages en conduisant).

Les *règles d'appel de services* qui permettent d'intégrer des sources d'information externes, personnelles (ex : agenda) ou publiques (ex : météorologie). Elles ajoutent au *e-Wallet* les capacités d'un annuaire sémantique de services qui dès lors peuvent être automatiquement identifiés et invoqués pour traiter des requêtes. Les règles d'appel de services relient des types de connaissances contextuelles et des services Web disponibles pour obtenir ces connaissances ; chaque ressource personnelle est transformée en un Service Web Sémantique (ex : une règle indiquant que l'activité courante de l'utilisateur peut être obtenue en appelant le service Web correspondant à son agenda Outlook). Plusieurs règles peuvent fournir la même information, par exemple si le portable de l'utilisateur est connecté au réseau sans-fil c'est une façon de le localiser sinon, son agenda peut suggérer une réponse (ex : une salle de réunion). Enfin, pour répondre à des questions sur l'utilisateur des services publics peuvent également être utiles, ex : la question "l'utilisateur est-il dans un endroit ensoleillé?" requiert typiquement la localisation de l'utilisateur et l'accès à un service public de météorologie.

Les *préférences de confidentialité* expliquent quelles informations l'utilisateur est disposé à révéler, à qui et sous quelles conditions. Ces préférences se divisent elles-mêmes en deux catégories : (1) *Les règles de contrôle d'accès* explicitent qui a le droit de voir quelle information et sous quelles conditions, ex : ma localisation est accessible uniquement à mes collègues et pendant les jours de travail entre 8H et 20H. (2) *Les règles de révision* nuancent les accès : les préférences de confidentialité d'un utilisateur sont rarement booléennes et impliquent différents niveaux d'exactitude ou d'inexactitude. La *révision par abstraction* permet de contrôler le niveau de détails d'une réponse en fonction du contexte (ex : indiquer que vous êtes en ville sans donner l'adresse exacte). La *révision par falsification* est lorsque l'utilisateur ne veut pas que l'on sache qu'il cache une information et préfère fournir une réponse fautive (ex : ne pas vouloir indiquer son véritable courriel par crainte de recevoir des publicités non sollicitées).

Tous ces types de connaissances sont représentés en OWL [21] par une modélisation orientée ontologies. Ceci requiert des ontologies : des attributs contextuels, des ressources personnelles, des domaines des services (ex : types de cuisines et les préférences culinaires). La réalisation du *e-Wallet* repose sur JESS [8], un moteur de règles et son langage CLIPS. Comme le montre la figure 2, la base de connaissances du *e-Wallet* est initialisée avec : (a) un modèle des triplets RDF [23], (b) un modèle gérant l'appel au services et la confidentialité (c) un méta-modèle de OWL. Dans un deuxième temps, les connaissances additionnelles sont chargées en traduisant des descriptions OWL en des faits et règles CLIPS. Ces descriptions incluent des ontologies et leurs instances qui sont transformées en triplets, des règles en chaînage-avant utilisées pour compléter la base (ex : définitions,

préférences dynamiques) ainsi que des règles d'appel de services et des règles de confidentialité en chaînage-arrière. Les langages ROWL, WOWL, SOWL et QOWL sont des extensions de OWL pour décrire respectivement, des règles de déduction, des règles d'appel de services, des règles de confidentialité et des requêtes. Les traductions sont en XSLT [24].

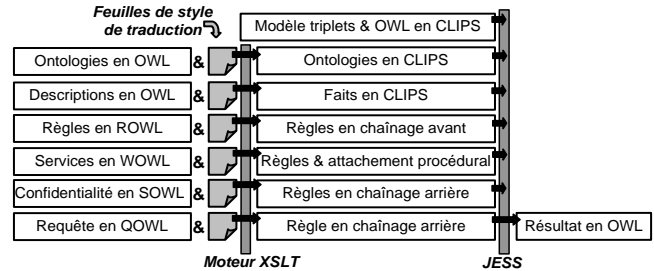


Figure 2. Fonctionnement du *e-Wallet*.

Les détails techniques de la réalisation du *e-Wallet* ainsi que des scénarios et des tests sont donnés dans [9]. La figure 3 montre deux interfaces du *e-Wallet* : l'écran de gauche est une vue des connaissances statiques sur l'utilisateur et l'écran de droite permet d'éditer une règle révisant l'accès à sa localisation, en ajoutant/enlevant des contraintes sur le contexte d'accès.

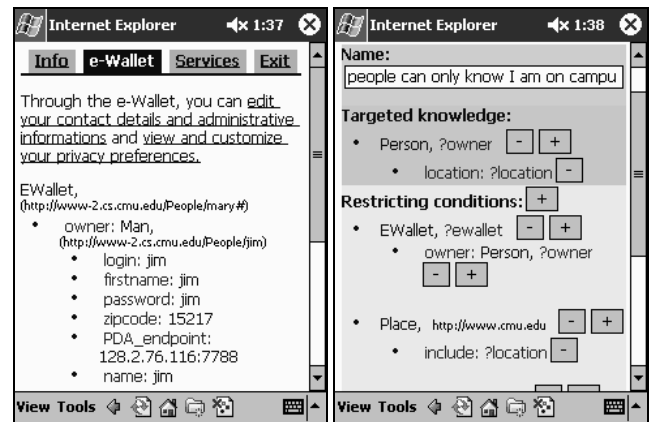


Figure 3. Deux écrans de l'interface du *e-Wallet*.

1.3 Travaux similaires

De plus en plus, les concepteurs de logiciels cherchent à améliorer la facilité et le confort d'utilisation en capturant le contexte dans lequel leurs utilisateurs opèrent afin d'y adapter le comportement des applications ; cette caractéristique s'appelle la conscience du contexte [18]. Il existe une grande variété d'informations sur le contexte et une grande variété de sources susceptibles de nous les fournir. De plus, ces informations et ces sources tendent à changer d'un utilisateur à l'autre et au cours du temps.

Les contributions au développement d'applications conscientes du contexte sont nombreuses : Active Badge développé chez Olivetti réoriente les appels téléphoniques en fonction de votre localisation [20] ; ParcTab développé à Xerox Palo Alto permet grâce à des assistants personnels numériques (PDA) de localiser les ressources à proximité de l'utilisateur (ex : imprimante), de laisser des notes électroniques virtuelles dans une pièce, etc. [17]. Des efforts plus récents incluent Oxygen [5], Aura [10] et plusieurs projets à Berkeley (ex. [11]) pour n'en nommer que quelques uns. D'autres architectures ont été proposées pour

réaliser des guides touristiques électroniques dans un musée ou dans une ville [3] Nous nous intéressons à des scénarios sur un campus ; le scénario du guide est intéressant pour un nouvel arrivant, mais n'est que l'un des nombreux scénarios possibles dans la vie quotidienne d'un campus. Il est donc important que notre architecture soit ouverte non seulement pour permettre l'arrivée et le départ des utilisateurs, mais aussi l'ajout et le retrait de services pour répondre à la demande et à ses évolutions. Nous utilisons des technologies standard (navigateur web, langages du W3C, etc.) afin de permettre aux utilisateurs de se connecter avec le terminal de leur choix. Nos scénarios concernent tout le campus, aussi bien dans les bâtiments, qu'entre les bâtiments ou dans les résidences étudiantes voisines. Carnegie Mellon est couvert par plus de 650 bornes Wifi 802.11b. La cohérence du comportement est assurée d'un scénario à l'autre grâce au *e-Wallet* qui, par sa conception, est persistant et indépendant des scénarios choisis. Il n'est pas attaché à un terminal particulier et peut résider n'importe où ex : sur un serveur du département de l'utilisateur.

Traditionnellement, les applications et les services utilisant le contexte sont liés de manière prédéfinie à des sources d'informations contextuelles et cela souvent directement au cœur de leur code. Rapidement les praticiens du domaine ont réalisé qu'il fallait séparer les mécanismes d'acquisition d'informations contextuelles des services susceptibles de les utiliser. La bibliothèque d'outils proposée par Georgia Tech représente l'un de ces efforts [6] permettant d'intégrer de façon modulaire des sources d'informations contextuelles tout en isolant des applications. Nous avons choisi de transformer ces sources en services Web. Ainsi, le *e-Wallet* peut utiliser des descriptions sémantiques de ces ressources et permettre la découverte et l'accès automatique. Cette architecture permet une intégration en temps réel des ressources personnelles et publiques pour une utilisation immédiate par des agents logiciels sensibles au contexte.

Une architecture ouverte pour l'accès à des informations et des services personnels pose immédiatement le problème de la sécurité et de la confidentialité. Les utilisateurs doivent pouvoir contrôler qui a accès à leurs ressources personnelles et dans quel contexte. La notion de *e-Wallet* comme présentée dans le .NET se limite à stocker un faible nombre d'informations sans offrir une réelle flexibilité pour en définir les règles d'accès : les utilisateurs peuvent uniquement indiquer s'ils sont disposés ou non à partager une partie de leur profil avec tous les services proposés sans pouvoir distinguer entre les différents services. Notre *e-Wallet* permet aux utilisateurs de spécifier l'accès à n'importe laquelle de leurs ressources personnelles pour n'importe quel service.

Le *e-Wallet* s'inscrit aussi dans la continuité des efforts récents pour développer des langages plus riches afin de capturer les préférences des utilisateurs en matière de confidentialité sur le Web, tels que P3P et APPEL [22]. Nous intégrons les ontologies à ce domaine et les utilisons pour décrire des règles de confidentialité qui peuvent ainsi mobiliser une grande variété d'attributs. Cela permet de plus aux utilisateurs d'indiquer des règles de révision contrôlant le niveau d'exactitude (ou d'inexactitude) avec lequel leurs informations sont révélées. Vous pouvez accepter de dire à vos collègues dans quelle pièce vous êtes pendant les heures de bureau, ou vous contenter de confirmer que vous êtes dans l'enceinte de votre lieu de travail, ou même simplement donner le nom de la ville où vous êtes. Cela inclut également des scénarios où vous pourriez vouloir feindre d'être dans un endroit, alors qu'en réalité vous êtes ailleurs. En effet,

dans certains cas, refuser de répondre à une question éveille une suspicion toute aussi dangereuse que la réponse à cette question. Un employé de banque qui aurait accès à la salle des coffres mais refuserait de donner sa localisation lorsqu'il est dans cette salle éveillerait rapidement les soupçons par ses refus ponctuels. Si au lieu de refuser il répond qu'il est dans un bureau adjacent, il devient beaucoup plus difficile de savoir qu'il a accès aux coffres.

Enfin, la communauté s'intéressant à la sécurité a développé de puissants langages pour décrire les droits d'accès tels que SAML [13], XACML [14] et EPAL [19] mais ces langages ne tirent pas profit des efforts du Web sémantique comme OWL sur lequel repose notre travail. Nous suivons aussi l'évolution des Services Web Sémantiques [4] et les ontologies sur lesquelles ils reposent pour la découverte et l'accès automatisés à des services.

2. EXEMPLES D'AGENTS-SERVICES

A partir de cette section nous présentons des *agents-services* développés et testés dans *myCampus*. Les deux premiers agents présentés sont issus de la première version testée début 2003 sur le campus de Carnegie Mellon pour valider le concept. Les agents suivants se divisent en deux initiatives : (1) InfoBridge : un projet avec quatre étudiants en IHM se focalisant sur l'identification et l'évaluation d'un service utile et utilisable (2) la deuxième version de *myCampus* : se focalisant sur des services et des scénarios démontrant les potentialités du *e-Wallet*. Comme le montre la figure 4, une session commence par un écran d'accueil et en cliquant sur les onglets, l'utilisateur peut consulter son *e-Wallet* ou la liste des services qui lui sont accessibles.

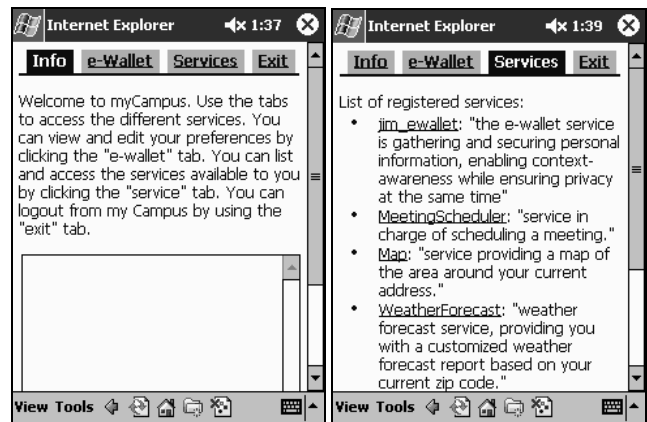


Figure 4. Ecran d'accueil et liste des services disponibles.

2.1 Recommander des restaurants

Testé dans la première version de *myCampus*, le "Concierge" est un agent suggérant aux utilisateurs où prendre leurs repas, en tenant compte de leurs préférences, de l'endroit où ils se trouvent et du temps qu'il fait. Pour cela, l'agent s'adresse au *e-Wallet* de l'utilisateur qui l'a invoqué, afin d'obtenir ses préférences (goûts culinaires, budget, distance, etc.) et sa localisation. Les préférences font partie de la connaissance gardée au cœur du *e-Wallet* et éditables à travers des interfaces dédiées comme montré sur l'écran de gauche de la figure 3. La position de l'utilisateur sur le campus est obtenue à travers le *e-Wallet* qui invoque le service de localisation par le réseau sans-fil. Le Concierge se connecte aussi à un service Web public donnant la situation météorologique. Il utilise ensuite UDDI [12] pour obtenir une liste de services Web correspondant aux restaurants des environs ;

nous avons créé cette liste en utilisant la base de test d'IBM. Le concierge en retire pour chaque restaurant l'URL d'une description formelle (cuisine, localisation, prix, etc.) qu'il utilise pour suggérer une liste ordonnée de restaurants (figure 5 - à droite).

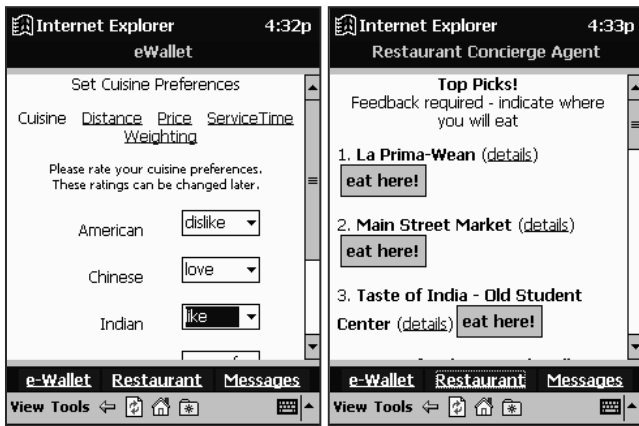


Figure 5. Deux écrans de l'interface du Concierge.

2.2 Filtrer et router des messages

Le "Messenger", aussi testé dans la première version de *myCampus*, a été particulièrement populaire. Son rôle est de filtrer les messages envoyés à l'utilisateur en tenant compte de ses centres d'intérêt et de sa disponibilité (ex : "quand je suis occupé, retarde les messages jusqu'à ce que mon activité soit terminée"). Là encore, les centres d'intérêt et les préférences sont extraits de *e-Wallet* et modifiables à travers des interfaces dédiées (figure 4 - à gauche). L'activité courante de l'utilisateur est nécessaire à la prise de décision et est obtenue à travers le *e-Wallet* qui se connecte à un service Web correspondant à l'agenda Outlook sur le PDA de son propriétaire. Le Messenger utilise sa connaissance des contextes et des préférences des utilisateurs pour trier, retarder et router les messages afin que seuls les messages jugés pertinents arrivent à l'utilisateur et ce au moment le plus propice.

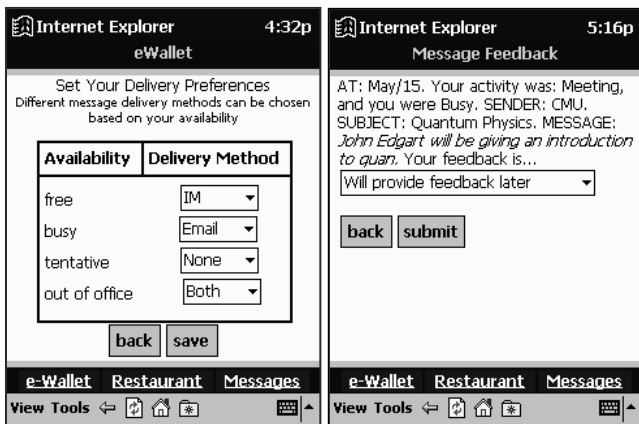


Figure 6. Deux écrans de l'interface du Messenger.

2.3 Gérer des posters virtuels

À la suite de la première version de *myCampus* et de son évaluation (c.f. section 4). Nous avons commencé un projet avec des étudiants en IHM pour identifier et évaluer un service qui, avec notre plate-forme déployée sur le campus de Carnegie Mellon, serait à la fois utile et utilisable. Les étudiants se sont focalisés sur un cas particulier du Messenger : l'annonce

d'événements / manifestations / etc. Traditionnellement le système le plus efficace et le plus ancien pour ce genre d'annonces est le poster. Comme l'illustre la figure 7, les zones de posters sont un moyen simple de réaliser une communication de masse ou de personne à personne, le graphisme et l'accrochage aux tableaux permet un balayage et un filtrage rapide des annonces, et le choix du lieu où le poster est accroché est important (bâtiment des arts, résidences, etc.) c.-à-d. les posters sont des messages situés. En contrepartie, les posters sont dans des lieux de passage et les passants intéressés mais pressés, risquent de les oublier rapidement. De plus ils sont contraignants dans leur production (mise en page, graphisme, impression), dans leur diffusion (afficher dans les lieux où passent des gens susceptibles d'être intéressés) et dans leur maintenance.



Figure 7. Une zone de posters à Carnegie Mellon.

Nous avons donc développé et testé un environnement de posters virtuels. Nous avons en particulier étudié le lien entre le profil d'un utilisateur (ses centres d'intérêts) et son parcours typique maximal dans une journée de travail sur le campus. La figure 7 illustre cette idée que les utilisateurs sont, dans une certaine mesure, décrits par les lieux qu'ils fréquentent : le parcours en trait plein est celui d'un étudiant du département *design* alors que le parcours en pointillés est celui d'un étudiant du département *computer science*. L'idée a donc été d'introduire le parcours de l'utilisateur dans les mécanismes de filtrage des messages.

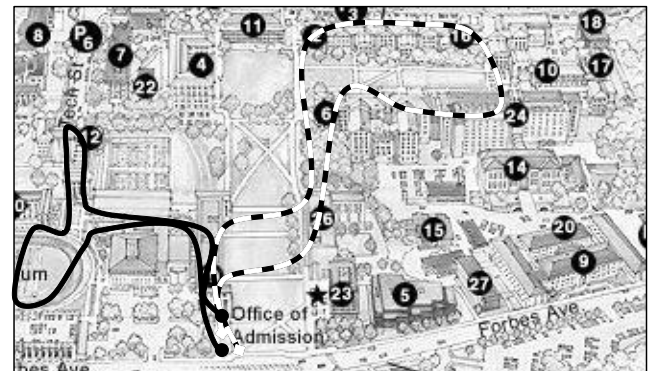


Figure 8. Parcours quotidiens typiques d'un étudiant en design (à gauche) et d'un étudiant en informatique (à droite).

Le résultat fut le prototype InfoBridge [2] dont deux écrans sont montrés en figure 9. Comme le montre la partie gauche l'utilisateur peut chercher des posters, consulter les posters qu'il a collectés avec son PDA lors de ses déplacements sur le campus, composer un poster (détails dans la partie droite) et éditer ses préférences. Si un utilisateur affirme avoir un intérêt (resp. un désintérêt) pour un sujet, tous les posters portant sur ce sujet lui seront envoyés (resp. cachés) quels que soient ses déplacements

sur le campus. Si l'utilisateur n'a pas donné de préférences sur un sujet, son parcours détermine s'il collecte les posters sur ce sujet.

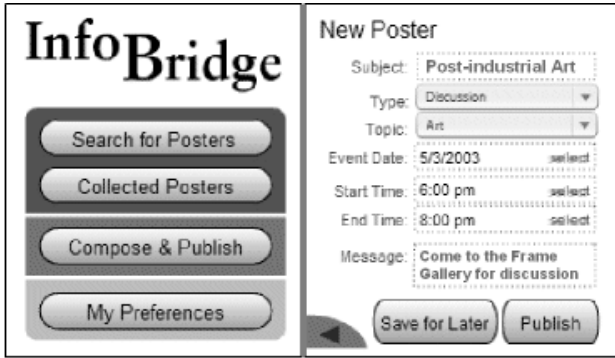


Figure 9. Deux écrans de l'interface InfoBridge.

Un scénario type résume ce service : "Delphine sait qu'un groupe d'étudiants organise un débat sur les logiciels libres à la cafétéria à 16H. Elle est actuellement à la cafétéria et utilise son PDA, pour y déposer un poster virtuel contenant des informations sur l'événement. Thomas, qui est un fervent défenseur de Linux, reçoit immédiatement le poster même s'il ne se trouve pas à la cafétéria. Plus tard, Renaud passe par la cafétéria et n'a pas stipulé d'intérêt ou de désintérêt pour le sujet. Puisqu'il est près de la cafétéria, son PDA sonne/vibre ; il peut consulter immédiatement le poster ou l'ignorer pour l'instant et le regarder plus tard avec les autres posters qu'il aura collectés dans la journée au cours de ses déplacements sur le campus. Quand il regarde le poster il peut décider d'ajouter l'événement à son agenda s'il est intéressé, l'ignorer s'il n'est pas intéressé ou même stipuler qu'il ne veut plus recevoir de posters sur ce sujet. Une fois la date du débat passée, le poster est invalidé, et automatiquement supprimé du système."

2.4 Cinéma et météorologie

Ces deux services ainsi que les services suivants ont été développés dans la version 2 de *myCampus* pour tester et démontrer les potentialités du *e-Wallet*. Comme le montre la figure 10, ces deux services proposent respectivement une liste des films projetés dans les environs, et les prévisions météorologiques. Aucune interface n'est nécessaire pour invoquer ces services : ils s'adressent directement aux *e-Wallets* pour obtenir les informations dont ils ont besoin (localisation). Ils illustrent l'aspect interface unifiée que le *e-Wallet* propose pour l'accès aux connaissances personnelles ou contextuelles.

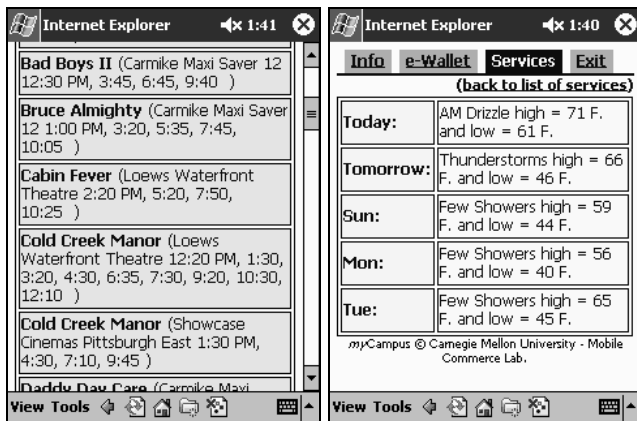


Figure 10. Les services de Cinéma et Météorologie.

2.5 Cartographie

L'agent de cartographie n'a pas non plus d'interface et renvoie instantanément une carte des environs à l'utilisateur. Cet agent nous est particulièrement utile pour démontrer l'importance et le fonctionnement des règles de révision du *e-Wallet*. Considérez la figure 11 : la partie de gauche montre le résultat obtenu par Marie qui n'a pas de règle de révision et donne directement le code postal du lieu où elle se trouve ; la partie de droite montre, pour le même service, le résultat obtenu par Jim qui ne souhaite pas donner plus de précision que la ville la plus proche.

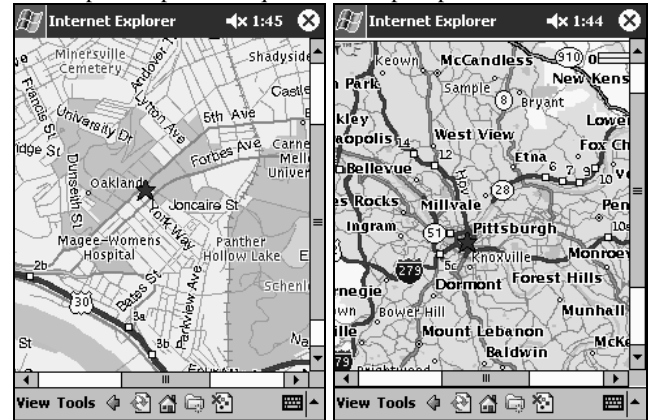


Figure 11. Cartographie et niveaux de confidentialité

2.6 Organiser une réunion

L'agent "Assistant de réunion" permet à un utilisateur de trouver un créneau libre dans son agenda et celui d'une autre personne afin de proposer une réunion. La figure 12 montre l'interface nécessaire pour donner les contraintes portant sur la réunion ; le reste des informations sur les utilisateurs (ex : disponibilité) est obtenu à travers leurs *e-Wallets* qui contactent les agendas respectifs des utilisateurs. Ce service donne un exemple de scénario impliquant plusieurs *e-Wallets*. A titre indicatif un extrait du diagramme d'interaction est donné en figure 12.

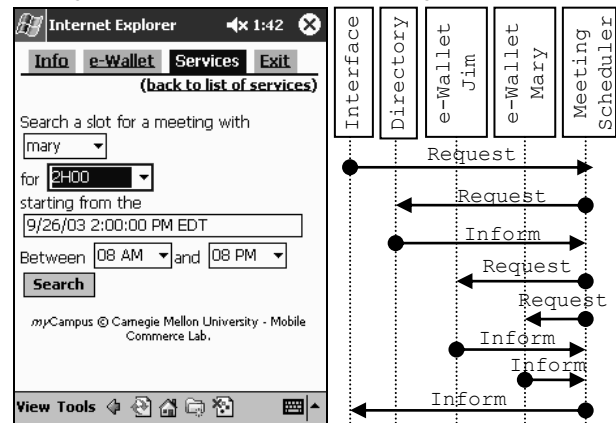


Figure 12. Organiser une réunion avec 2 e-Wallets

2.7 Projeter une présentation

Le dernier service est notre première expérience en matière d'informatique ambiante. Comme le montre la figure 13, ce service permet à un utilisateur de choisir une présentation PowerPoint parmi celles annotées dans son *e-Wallet* et de la projeter sur le vidéo-projecteur le plus proche en prenant le contrôle à travers une interface/télécommande sur son PDA.

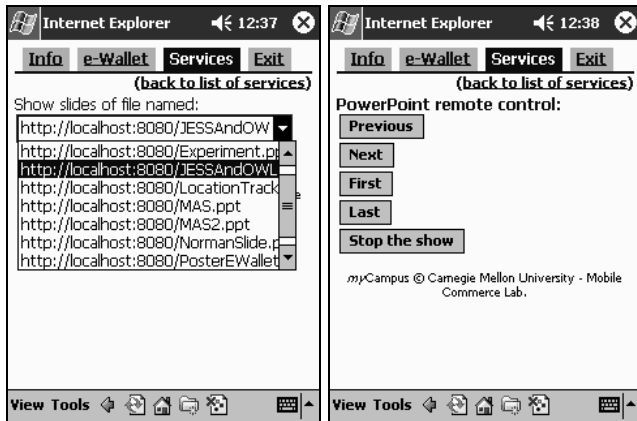


Figure 13. Projeter une présentation

3. TESTS ET EVALUATIONS

3.1 Tests d'utilisation de la première version

La première version du système a été validée sur le campus de Carnegie Mellon où l'environnement était accessible à travers des PDA connectés et localisés à travers le réseau sans-fil de l'université. En utilisant ce premier prototype, nous avons entrepris une première expérience pour évaluer la faisabilité, l'utilisabilité et l'utilité. L'expérience de 3 jours a impliqué 11 utilisateurs choisis pour couvrir un large spectre de profils. Ils étaient libres de se connecter depuis n'importe où sur le campus et d'utiliser le Concierge (recommandation de restaurants) et le Messenger (filtrage et routage des messages). Ils devaient maintenir leur agenda à jour de façon à assurer une bonne connaissance du contexte tout au long de l'expérience. L'expérience a totalisé 44 envois de messages filtrés par le Messenger et chacune de ses décisions était évaluée par l'utilisateur concerné donnant lieu à une base de 484 retours des utilisateurs. Les utilisateurs ont également demandé 28 recommandations de restaurant au Concierge. Les actions, les résultats, le contexte et les incidents ont été enregistrés dans une base d'historique pour chacun des agents. En outre, chaque utilisateur a dû remplir un questionnaire et passer un entretien final en tête-à-tête avec un membre du projet pour faire un compte-rendu de son expérience personnelle et nous permettre de comprendre certains des résultats observés dans la base d'historique. Nous ne nous étendons pas sur les problèmes dus au matériel. Cependant la majeure partie des incidents étaient dus soit aux aléas de la connexion WiFi, soit aux PDA. Notons aussi que les utilisateurs ont trouvé les PDA trop larges pour être transportés comme un téléphone cellulaire et trop petits pour remplacer les ordinateurs portables dont sont équipés presque tous les acteurs du campus. De plus l'autonomie d'un PDA connecté au WiFi est très mauvaise (parfois moins d'une heure). Nous envisageons de migrer le système vers des téléphones capables d'utiliser le WiFi et/ou le Bluetooth.

Pour le Concierge, les résultats ont prouvé que 12.5% des recommandations étaient acceptables pour l'utilisateur uniquement parce que la connaissance du contexte avait permis de faire le bon choix ; en d'autres termes le restaurant choisi par l'utilisateur n'aurait pas été recommandé si le système avait seulement utilisé le profil statique des utilisateurs ignorant leur localisation et le temps. Ceci représente déjà une amélioration de 14.29%, mais nous attendions plus. Le compte rendu des entretiens a mis à jour plusieurs problèmes, en particulier :

- La plupart des utilisateurs mécontents des recommandations avaient été sur le campus assez longtemps pour connaître tous les restaurants et pour avoir développé des habitudes ; ils n'auraient pas consulté un concierge en temps normal.
- La logique de l'agent était trop naïve, ex : nous basons la recommandation sur le lieu où l'utilisateur se trouve actuellement, il est tout aussi intéressant de recommander des restaurants proches de l'endroit où il doit se rendre dans l'heure suivante ; cette connaissance est disponible dans l'agenda.
- Des connaissances et des fonctionnalités étaient absentes : régimes spéciaux, plats du jour, longueur de la file d'attente, fonctionnalité pour organiser un repas collectif, etc.

Pour le Messenger, les 44 messages envoyés à fréquence croissante sur 3 jours ont donné lieu à 484 évaluations : pour chacun des messages l'utilisateur devait donner la décision que le message aurait dû prendre idéalement (ignorer le message, l'envoyer immédiatement, l'envoyer durant le prochain créneau libre dans l'agenda, l'envoyer en fin de journée après toutes les activités). La connaissance du contexte a montré une amélioration systématique des résultats par rapport à des profils utilisateurs statiques. Dans 70% des cas la meilleure décision de routage n'aurait pas pu être prise sans connaissance du contexte.

3.2 Analyse des traces d'utilisation

Pour illustrer la richesse et la complexité des résultats enregistrés, nous allons commenter quelques extraits de la base d'historique du Messenger (figure 14). Dans l'ensemble, un filtrage basé sur un profil statique a été jugé trop lâche ; il y a un besoin de filtrage plus intelligent et nous verrons qu'il y a des occasions pour où la connaissance du contexte apporte des améliorations. La figure 14a montre l'intérêt porté à cinq des messages en fonction du sexe des utilisateurs, le sexe n'ayant pas été pris en compte dans l'algorithme de filtrage. Le message (1) était une information au sujet d'une attaque par bombe et le message (3) donnait les prévisions météorologiques ; tous les deux sont clairement indépendants du sexe. Le message (5) est une annonce pour une conférence au sujet de la place des femmes dans la recherche ; ici le sexe est sensiblement distinctif et c'est compréhensible. Cependant les messages (2) et (4), annonçant respectivement la réunion d'un cercle de compositeurs et un prix bon marché pour des billets vers la Jamaïque, montrent également des différences significatives entre les deux sexes sans qu'il n'y ait aucune raison évidente. Si l'on veut envisager un apprentissage symbolique des préférences cela pose un problème et demande une connaissance plus fine des utilisateurs et des critères de filtrage plus précis.

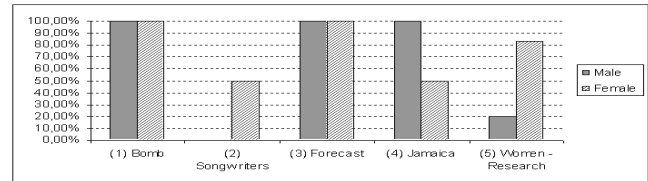
La figure 14b montre l'intérêt porté à cinq des messages en fonction du centre d'intérêt principal des utilisateurs. Comme prévu, l'intérêt pour des informations générales, telles que le message (2) au sujet d'une attaque par bombe, est clairement indépendant du centre d'intérêt. De même le message (1) annonçant une conférence technologique et le message (3) annonçant une conférence en biologie, voient leur intérêt varier en fonction du centre d'intérêt principal de l'utilisateur. Par contre, le message (4) est pour le moins étonnant car il montre un intérêt relativement bas des sciences humaines pour l'annonce d'une pièce de théâtre alors que l'on attendrait le contraire. La vraie raison est que la plupart des personnes intéressées par des sciences humaines avaient déjà reçu cette annonce par d'autres voies d'information ; ceci montre encore un problème qu'un algorithme d'apprentissage risquerait de mal interpréter et aussi les risques d'un raisonnement en vase clos.

Enfin, la figure 14c montre les préférences de routage des utilisateurs pour 8 des messages. Pour chacun des messages, les utilisateurs pouvaient indiquer : que le message était d'aucun intérêt pour eux ("No interest") ; que le message était tellement intéressant qu'ils auraient voulu le recevoir immédiatement ("Instantly") c.-à-d. ils étaient disposés à être dérangés dans leur activité pour un tel message ; que le message était intéressant mais qu'il aurait pu attendre qu'ils soient disponibles ("Available") c.-à-d. ils ne souhaitent pas être dérangés pour un tel message s'ils étaient occupés ; et enfin que le message était d'intérêt général et devrait être envoyé plus tard dans la journée, après le travail ou même être vu uniquement à la demande ("Anytime"). La barre "Undecided" montre le pourcentage des utilisateurs qui n'ont pas voulu fournir de retour pour un message. Le message (1) au sujet d'une attaque par bombe, montre typiquement les deux types de comportements pour des informations : certains des utilisateurs veulent savoir immédiatement tandis que d'autres sont du style à attendre le journal de 20H. Le message (6) au sujet du programme des Cinémas peut clairement être envoyé n'importe quand ou consulté sur demande. Nous avons déjà commenté le message (3) annonçant une réunion de compositeurs et il pourrait sembler que, de la même façon, le message (8) au sujet d'informations sur les impôts réclame un filtrage basé sur les centres d'intérêt ; cependant la raison du faible intérêt était qu'aucun des utilisateurs n'avait travaillé l'année précédente et donc qu'aucun d'eux n'était concerné cette année là par ce type de messages. Un tel filtrage est beaucoup plus complexe que le filtrage par centres d'intérêt et exige une connaissance approfondie de l'utilisateur et du domaine. Le message (5) apparaît comme un type de messages qui devrait être envoyé immédiatement, mais en fait cela est dû à un effet secondaire : ce message donne les prévisions météorologiques, mais parce qu'il a été envoyé tôt le matin, le fait qu'il soit montré immédiatement a été bien perçu, alors qu'en fait ce type de message réclame la capacité de lier aux messages/sujets des intervalles de temps de sorte que le système se rende compte de la pertinence d'un message donné à un moment donné. Les messages (4) et (7) étaient au sujet d'événements ayant lieu le jour même (une présentation sur les carrières dans la consultance et un concert en soirée) et un grand nombre de personnes ont souhaité que ces messages leur soient envoyés dès qu'elles sont disponibles dans la journée. Par conséquent les messages (4), (5) et (7) montrent bien l'intérêt d'un filtrage et d'un routage conscients du contexte : en vérifiant l'activité et la disponibilité des utilisateurs dans leurs agendas et en les combinant avec des contraintes sur l'heure de livraison des messages, un système peut trouver les meilleures fenêtres de temps pour envoyer des messages à un utilisateur de façon moins intrusive et plus efficace.

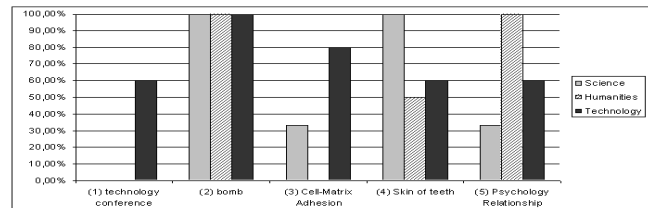
3.3 Evaluation du service InfoBridge

Les résultats de cette première expérience ont motivé une deuxième expérience (InfoBridge) en parallèle avec le développement de la deuxième génération de l'architecture. Une équipe d'IHM a commencé par étudier les habitudes et les pratiques [2] pour annoncer et être informés des événements prévus, ainsi que la façon dont nous organisons notre vie. Ils ont appliqué la méthode du "cheminement cognitif" sur les différentes versions des interfaces du Concierge et d'InfoBridge afin d'identifier un maximum des problèmes d'interactions. Des exemples de problèmes identifiés et résolus [2] sont: la nécessité sur un PDA, qui peut être utilisé par exemple en marchant, de donner des indices visuels évidents du fonctionnement et de

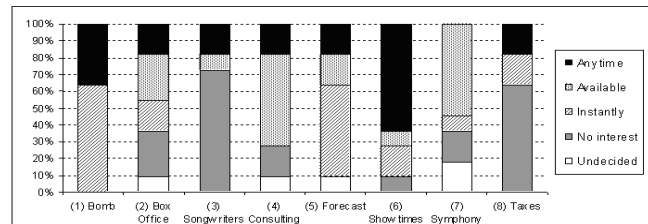
l'activité du système (une fois cliqués les boutons reste enfoncés jusqu'à ce que le système traite l'évènement, de façon à montrer visuellement et rapidement que le clic a été enregistré et est en cours de traitement) ; l'utilisation de groupement par couleurs pour les fonctionnalités du système afin de bien montrer l'étape courante ; la mise en place de contrôles graphiques simplifiant la saisie notamment difficile sur un PDA ; l'introduction d'une carte du campus pour déposer des posters à distance ; etc.



(a) extrait : nombre d'utilisateurs intéressés / sexe



(b) extrait : nombre d'utilisateurs intéressés / centre d'intérêt



(c) extrait des retours sur le filtrage et le routage

Figure 14. Extraits des statistiques sur l'historique

La dernière version d'InfoBridge a été testée sur le terrain avec deux personnes du projet suivant personnellement chacun des 4 bêta-testeurs dans leurs déplacements et leur utilisation du système. Après une introduction au système il était demandé aux testeurs de suivre le(s) chemin(s) qu'ils empruntent dans une journée très chargée. Une séance de tests durait environ une heure, comportait la soumission d'au moins 3 posters et était suivie d'un entretien pour recueillir les retours d'utilisation. Les utilisateurs étaient définitivement séduits par le système mais ont soulevé un certain nombre de points [2]: la logique et les critères de sélection d'un poster demandent à être expliqués pour justifier la liste des posters collectés ; le lieu de collage virtuel des posters doit être clairement dissocié du lieu où l'évènement aura lieu ; des écrans de confirmations des actions ont été rajoutés pour conforter des utilisateurs dans leurs actions ; la dimension graphique des posters papiers n'est pas compatible avec les limitations d'un PDA et le couplage avec d'autres systèmes semble nécessaire, par exemple un URL vers une page Web ; les accidents tels qu'un double clic involontaire sont fréquents sur un système mobile et l'interface doit les prendre en compte ; etc.

La deuxième version du système myCampus a été terminée et démontrée fin 2003 sur des scénarios illustrant l'intérêt du e-Wallet. Nous projetons maintenant une deuxième phase d'évaluation de myCampus sur le terrain.

4. DISCUSSION

myCampus est un environnement ouvert basé sur les technologies des systèmes multi-agents et du Web sémantique, une plate-forme d'accès mobile à des services en ligne utilisant des informations contextuelles et personnelles tout en respectant la confidentialité des utilisateurs. Nous avons mis ici l'accent sur les scénarios et les tests d'utilisations du système dans son ensemble, et nous vous invitons à regarder [9, 15] pour plus de détails sur la réalisation. L'architecture est ouverte en reposant sur l'intégration dynamique des ressources contextuelles représentées par des services Web et des services représentés par des agents. Les techniques de modélisations utilisées sont elles aussi ouvertes: en reposant sur les ontologies et les langages du Web sémantique nous assurons l'extensibilité des modèles. Le *e-Wallet* fournit une interface sémantique unifiée et sécurisée pour l'accès aux ressources personnelles d'un utilisateur et permet ainsi aux services de les mobiliser. Il est persistant, indépendamment des scénarios choisis et exploite la flexibilité des modèles orientés ontologie.

Les résultats de nos tests réclament plus d'intelligence et de connaissances sur l'utilisateur et son contexte. L'analyse des interfaces a montré l'importance de réduire la charge cognitive des utilisateurs lors d'accès mobiles. L'ajout d'indices visuels de l'état du système fut une réponse immédiate, mais à plus long terme l'augmentation de la conscience du contexte permettra de réduire considérablement les besoins d'interactions et de les rendre les moins intrusives possible. Toutes les interfaces n'ont pas été étudiées en détail car l'évaluation et la re-conception sont extrêmement coûteuses. Seul le service des posters a fait l'objet d'une étude approfondie et complète pour nous donner une idée des contraintes existantes. Les autres interfaces sont plus ou moins dédiées: certaines permettent une manipulation générique des structures conceptuelles sous-jacentes (ex : interface du *e-Wallet*), elles sont très flexibles et utiles aux experts du système mais souvent trop complexes ; d'autres interfaces sont dédiées à un scénario (ex : les posters), elles sont beaucoup plus faciles d'accès et parlantes pour les utilisateurs, mais ne permettent que des actions anticipées et limitées au scénario et à son domaine. Ainsi, un défi demeure, héritage de toutes les approches utilisant des représentations riches : réconcilier l'expressivité des langages de représentation avec les exigences ergonomiques des utilisateurs finaux. Nous expérimentons actuellement différentes approches pour l'édition et l'apprentissage des préférences utilisateurs afin d'améliorer les interfaces actuelles.

Remerciements : les membres de myCampus (Joseph Kwon, Enoch Chan, Linh Van, Kazuaki Takizawa), les étudiants d'IHM (Tiffany Chang, Asa Sherril, Gil Tolle and Brandon Weber), DAML, Air Force Research Lab., DARPA, IBM, HP, Symbol, Boeing, Amazon, IST Program.

5. REFERENCES

- [1] Berners-Lee, Hendler, Lassila : The Semantic Web, *Scientific American*, May (2001)
- [2] Chang, Sherrill, Tolle, Weber : InfoBridge: Peer-to-peer location-aware virtual posters, *project report* <http://www-sop.inria.fr/acacia/personnel/Fabien.Gandon/research/infobridge2003/>
- [3] Cheverst, Davies, Mitchell, Friday, Efstratiou : GUIDE project de Lancaster Developing a Context-aware Electronic Tourist Guide: Some Issues and Experiences, *CHI 2000*, Netherlands, (2000) 17-24
- [4] DAML-S Coalition : DAML-S: Web Service Description for the Semantic Web, *ISWC'02, LNCS 2342*, (2002) 348-363
- [5] Dertouzos, M. : The Future of Computing, *Scientific American*, August (1999)
- [6] Dey, Salber, Futakawa, Abowd : An Architecture to Support Context Aware Computing, Tech. Report *GIT-GVU-99-23.*, Georgia Tech., Nov. (2000)
- [7] FIPA, *Specifications* (2002) <http://www.fipa.org/>
- [8] Friedman-Hill : Jess in Action: Java Rule-based Systems, Manning Publications Com-pany, June, ISBN 1930110898, (2003) voir aussi <http://herzberg.ca.sandia.gov/jess/>
- [9] Gandon, Sadeh : Semantic Web Technologies to Reconcile Privacy and Context Awareness, *Web Semantics Journal*, 1(3) (2004) voir aussi CMU-CS-03-211 Tech. Report, School of Computer Science, Carnegie Mellon Uni. <http://reports-archive.adm.cs.cmu.edu/anon/2003/abstracts/03-211.html>
- [10] Garlan, Siewiorek, Smailagic, Steenkiste : Project Aura: Towards Distraction-Free Pervasive Computing, *IEEE Pervasive Computing, Integrated Pervasive Computing Environments*, 1(2), (2002) 22-31
- [11] Hong, Llanday : A Context/Communication Information Agent, in Personal and Ubiquitous Com-puting Special Issue Situated Interaction and Context-Aware Computing, 5(1), (2001) 78-81
- [12] OASIS: Universal Description, Discovery and Integration standard, <http://www.uddi.org/>
- [13] OASIS: Security Assertion Markup Language (SAML), Technology Reports, April 14 (2003) <http://xml.coverpages.org/saml.html>
- [14] OASIS: Extensible Access Control Markup Language (XACML), Technology Reports, March 28 (2003) <http://xml.coverpages.org/xacml.html>
- [15] Sadeh, Chan, Van, Kwon, Takizawa : Creating an Open Agent Environment for Context-aware M-Commerce, *Agentcities: Challenges in Open Agent Environments*, LNAI, Springer Verlag, (2003) 152-158
- [16] Sadeh : m-Commerce: Technologies, Services and Business Models, Wiley (2002)
- [17] Schilit : A System Architecture for Context-Aware Mobile Computing, *Ph.D. Thesis*, Columbia Univ., 1995
- [18] Schilit, Adams, Want : Context-Aware Computing Applications, *IEEE Workshop on Mobile Computing Systems and Applications*, (1994) 85-90.
- [19] Schunter, Powers : The Enterprise Privacy Authorization Language (EPAL 1.1), IBM Research Lab., <http://www.zurich.ibm.com/security/enterprise-privacy/epal/>
- [20] Want, Hopper, Falcao, Gibbons : The Active Badge Location System, *ACM Transactions on Information Systems* 10(1) (1992) 91-102.
- [21] W3C: OWL Web Ontology Language Reference, Working Draft 31/03/2003, <http://www.w3.org/TR/owl-ref/>
- [22] W3C: The Platform for Privacy Preferences 1.0 (P3P1.0) Specification, Recommendation 16 April 2002, <http://www.w3.org/TR/P3P/>
- [23] W3C: RDF (1999) Resource Description Framework, <http://www.w3.org/TR/REC-rdf-syntax/>
- [24] W3C: XSLT (1999) XSL Transformations Version 1.0, Recommendation, <http://www.w3.org/TR/xslt>