

APPRENTISSAGE ET COMMUNICATIONS CONTEXTUELLES
EN SITUATION DE MOBILITE

Sébastien George,

Maître de conférences en informatique

sebastien.george@insa-lyon.fr , + 33 4 72 43 60 27

Aina Lekira,

Doctorante en informatique

aina.lekira@gmail.com

Adresse professionnelle

Université de Lyon, LIESP, INSA-Lyon ★ F-869621 Villeurbanne Cedex, France

Résumé : Notre travail se situe dans le domaine du *context-aware computing*, c'est-à-dire de la prise en compte par un système informatique du contexte de l'utilisateur. Plus particulièrement, notre proposition concerne la contextualisation des discussions dans des situations d'apprentissage en mobilité. L'objectif de cette recherche est de montrer comment il est possible de concevoir des applications mobiles prenant en compte le contexte de l'utilisateur. Nous présentons tout d'abord un modèle générique de communications médiatisées contextuelles qui prend en compte 6 dimensions pour définir le contexte de l'utilisateur (le profil de l'utilisateur, le dispositif qu'il utilise, sa localisation, son environnement, son activité et le temps). Dans un deuxième temps, nous décrirons un prototype nommé MeCoCo (*Mediated Contextual Communications*) fondé sur ce modèle et proposant une interface particulière permettant d'afficher les discussions selon les différentes dimensions du contexte.

Mots clés : communication médiatisée, communication contextuelle, apprentissage en mobilité

Summary: Our work deals with context-aware computing, i.e. how a system can take into account a user's context. More particularly, our proposal concerns the contextualization of discussions in mobile learning situations. The main goal of this research is to show how it is possible to conceive mobile applications that can adapt information to the user's context. We first present a generic model of contextual mediated communications which takes into account 6 dimensions to define the context of the user (user's profile, device, location, environment, activity and time). In addition, we will describe a prototype named MeCoCo (*Mediated Contextual Communications*) based on this model and that proposes a particular interface to display discussions according to the various dimensions of the context.

Keywords: Computer-Mediated Communication, Contextual Communication, Mobile Learning

APPRENTISSAGE ET COMMUNICATIONS CONTEXTUELLES

EN SITUATION DE MOBILITÉ

1 – PROBLEMATIQUE GENERALE

Avec le développement de l'informatique mobile et de l'informatique ubiquitaire, la contextualisation des informations données à l'utilisateur lorsqu'il utilise un dispositif technique est de plus en plus fréquente. Notre travail se situe dans le domaine du *context-aware computing* c'est-à-dire de la prise en compte par un système informatique du contexte de l'utilisateur. Plus particulièrement, notre proposition concerne la contextualisation des discussions lors de situations d'apprentissage en mobilité. Ainsi, en fonction de son activité, de sa localisation ou d'autres paramètres que nous développons par la suite, un utilisateur aura la possibilité de voir les discussions (synchrones ou asynchrones) se rapportant à son contexte à un moment donné. Il pourra aussi entrer en relation avec des personnes se trouvant dans un contexte proche du sien.

Le potentiel des communications contextuelles dans un contexte éducatif a déjà été mis en avant dans des travaux que nous avons menés par ailleurs (George & Labas, 2008). Ainsi, pour améliorer l'apprentissage humain, des forums contextuels particuliers ont été conçus (CONFOR - *contextual forum*) pour lier des discussions médiatisées à des objets pédagogiques d'un cours par exemple.

Dans le travail présenté ici, nous voulons tirer profit de la mobilité pour contextualiser les communications. Ainsi, en fonction de sa localisation et de son profil, un utilisateur aura accès à un certain nombre de discussions et de contacts. Par exemple, un étudiant qui visite un site archéologique peut utiliser son PDA pour voir les commentaires laissés par des scientifiques correspondant à certains endroits de ce site. Ce principe de contextualisation des discussions en situation de mobilité peut aussi être utile pour un apprentissage en situation et juste à temps. Ainsi, dans un contexte de travail, un employé peut se trouver dans une situation nouvelle pour lui (e.g. devant un nouveau matériel à utiliser) et il pourra alors

utiliser son *smartphone* pour entrer en communication synchrone par visio avec une personne plus experte de cette situation (i.e. connaissant bien ce même matériel).

Notre proposition a pour objectif de montrer comment concevoir des applications mobiles qui prennent en compte le contexte de l'utilisateur. Nous présentons dans la partie 2 les travaux en lien avec ce sujet. Dans la partie 3, nous proposons un modèle générique pour les communications médiatisées contextuelles. Ce modèle prend en compte 6 dimensions pour définir le contexte de l'utilisateur (le profil de l'utilisateur, le dispositif qu'il utilise, sa localisation, son environnement, son activité et le temps). Nous décrivons, dans la partie 4, un prototype nommé MeCoCo (*Mediated Contextual Communications*) fondé sur ce modèle et proposant une interface particulière permettant d'afficher les discussions selon les différentes dimensions du contexte. Nous terminons cet article par une conclusion et des perspectives.

2 – TRAVAUX CONNEXES

L'idée d'utiliser le contexte de l'utilisateur provient du domaine du CSCW (*Computer Supported Cooperative Work*) et des IHM (Interactions Homme-Machine). Les chercheurs des domaines ont montré l'importance de la prise en compte du contexte pour faciliter l'interaction entre un utilisateur et le système qu'il manipule. Des chercheurs en IHM ont mis en place un cadre de référence pour décrire les multiples contextes d'utilisation dans le domaine du *context-aware computing* (Calvary et al., 2003). Le contexte de l'utilisateur est décomposé en trois facettes : l'utilisateur final d'un système interactif, la plateforme informatique avec laquelle l'utilisateur interagit et l'environnement physique où il travaille. Le contexte est ainsi défini comme un triplet < utilisateur, plateforme, environnement > où :

- L'utilisateur représente celui qui utilise le système interactif. Il peut être décrit par un ensemble de valeurs caractérisant les capacités de perception, de cognition et d'action de l'utilisateur afin de choisir les meilleures modalités pour le rendu et la manipulation du système interactif.
- La plateforme est modélisée en terme de ressources matérielles et logicielles qui déterminent la manière dont l'information est calculée, transmise, rendue et manipulée par l'utilisateur. Ces ressources incluent la taille mémoire, la bande passante et les dispositifs d'interactions. Les ressources motivent le choix d'un ensemble de modalités d'entrées/sorties et pour chaque modalité un ensemble d'informations est disponible. La plateforme n'est pas limitée à un seul ordinateur personnel, elle couvre toutes les ressources d'interactions et toutes les ressources informatiques à un moment donné pour accomplir un ensemble de tâches.
- Un environnement est un ensemble d'objets, de personnes ou d'événements qui sont périphériques à l'activité courante mais qui peuvent avoir un impact sur le comportement du système ou des utilisateurs dans l'immédiat ou dans le futur. L'environnement peut inclure beaucoup de choses. En pratique, la limite est faite par l'analyste du domaine dont le rôle est d'extraire les entités pertinentes au cas étudié. Cela inclut des observations sur la pratique des utilisateurs aussi bien que des considérations pour des contraintes techniques. Par exemple, la luminosité devient un problème lorsqu'elle influe sur la robustesse d'un système informatique de suivi basé sur la vision.

Cette définition du contexte est assez générale et nous allons maintenant décrire des systèmes concrets existants.

ConChat (Ranganathan et al., 2002) est un système de discussions qui fournit des informations contextuelles aux utilisateurs en vue d'améliorer la communication électronique et de réduire les conflits sémantiques potentiels. En utilisant des indices contextuels, les utilisateurs peuvent déduire au cours d'une conversation ce que l'autre personne fait et ce qui se passe dans son environnement immédiat. *ConChat* est utilisé dans un environnement informatique pervasif de travail. Les informations contextuelles partagées sont : la localisation de l'équipe, le nombre de personnes ne faisant pas partie de l'équipe dans la salle, l'identité de ces personnes, la température, la lumière et le bruit dans la salle, l'humeur des utilisateurs (qui peut être « heureux », « triste », ...), le statut de l'utilisateur (qui peut être « au téléphone », « parti déjeuner », ...) et l'activité dans la salle (qui peut être « réunion », « lecture », ...). Par exemple, si un utilisateur A sait qu'un utilisateur B est en train de parler avec quelqu'un d'autre ou est impliqué dans une activité nécessitant toute son attention, A peut s'attendre à ce que B ne lui réponde pas rapidement, voire pas du tout. En revanche, si A apprend que B est en train d'assister à une réunion directement liée à leur conversation, A sait que B sera en mesure de lui répondre assez rapidement.

Kirsch-Pinheiro et al. (2005) ont développé une application qui prend en compte deux aspects du contexte de l'utilisateur : l'aspect physique (localisation, dispositif, application) et l'aspect organisationnel (groupe, rôle, calendrier, activité, objets et processus partagés). Ce dernier fait référence aux connaissances sur les processus collaboratifs dans lesquels l'utilisateur est impliqué.

Wang et al. (2004) ont proposé une ontologie de contexte, nommée *CONON* (*Context Ontology*), fondée sur OWL pour modéliser le contexte dans un environnement pervasif. *CONON* fournit une ontologie de haut niveau avec des concepts généraux de base pour définir un contexte, et fournit aussi des possibilités d'extension en ajoutant de manière hiérarchique une ontologie de domaine spécifique. Pour raisonner sur le contexte, des mécanismes de raisonnement logique sont mis en place pour vérifier la cohérence du contexte mais aussi pour déduire un contexte implicite de haut niveau à partir de contexte explicite de

bas niveau. Par exemple, un téléphone mobile pourrait s'adapter en fonction du contexte de l'utilisateur. Si l'utilisateur est en train de dormir dans sa chambre ou s'il est en train de prendre une douche dans la salle de bain, les appels seront transférés vers la boîte vocale du téléphone. S'il est en train de cuisiner ou s'il regarde la TV dans le salon, le volume de la sonnerie augmentera, au contraire s'il est en train de dîner avec sa famille dans la salle à manger, le téléphone se mettra en mode vibreur.

L'architecture proposée par Basaeed *et al.* (2007) est spécifique pour le m-learning (*mobile learning*). Cette architecture est adaptée pour le contextualisation des ressources (objets d'apprentissage, présentation et navigation) mais elle n'intègre pas la contextualisation des communications entre personnes.

Ces dernières années, plusieurs projets ont été menés pour favoriser les communications médiatisées sensibles au contexte. Parmi ces projets, nous pouvons citer :

- *CybreMinder* (Dey & Abowd, 2000) qui permet à l'utilisateur d'associer l'information contextuelle à des travaux « à faire » (*to-do items*) et les délivre à l'utilisateur selon des conditions prédéfinies.
- *Smart Instant Messenger* (Law *et al.*, 2006) qui est fondé sur les contacts que l'on considère comme des « copains » et qui offre à son utilisateur un groupement dynamique des contacts selon leurs activités ou leurs localisations ainsi que des services liés aux ressources des dits contacts.

Nous remarquons que, malgré les efforts fournis, peu de travaux se sont focalisés sur une modélisation complète du contexte dans le cadre de communications médiatisées entre personnes en situation de mobilité. Dans notre travail, l'objectif est de proposer un modèle de contexte générique pour ces communications médiatisées.

3 – PROPOSITION D'UN MODELE DE CONTEXTE GENERIQUE

Dans les systèmes adaptatifs et sensibles au contexte, la notion de contexte est fondamentale. De nombreuses définitions existent. Dans de nombreux travaux, la notion de contexte se focalise uniquement sur la localisation, l'activité ou l'environnement d'une entité (une personne ou un objet). Nous nous intéressons à un point de vue différent puisque nous nous focalisons sur l'utilisateur et donc à un contexte orienté utilisateur, c'est-à-dire défini par rapport ou en fonction d'un utilisateur.

Une des définitions de contexte largement acceptée est celle de (Dey, 2001) qui le définit comme « *any information that can be used to characterize the situation of an entity. An entity is a person, place or object that is considered relevant to the interaction between a user and an application, including the user and the applications themselves* ». Cette définition ne prend pas uniquement en compte l'utilisateur, puisqu'elle se focalise également sur d'autres entités comme la localisation, les dispositifs, etc.

Cette définition est d'autant plus intéressante qu'elle ouvre la voie à la prise en compte de tout ce qui pourrait conduire à expliciter le mieux possible le contexte de l'utilisateur. Nous en tirons une conséquence immédiate : la multi-dimensionnalité du contexte.

Cette analyse nous conduit à la définition du contexte de l'utilisateur à travers 6 dimensions essentielles : le profil de l'utilisateur, l'activité, le dispositif technique, l'environnement de l'utilisateur, la localisation et le temps.

Ainsi, nous décrivons le contexte d'un utilisateur par un sextuplet :

< **Utilisateur, Activité, Dispositif, Environnement, Localisation, Temps** > et un contexte i est défini par la suite comme suit :

$$C_i = \langle U_i, A_i, D_i, E_i, L_i, T_i \rangle$$

Chacune de ces dimensions a son importance dans notre modèle puisqu'elles permettent ensemble de définir le contexte de l'utilisateur de manière complète afin de fournir à ce dernier des discussions/commentaires pertinents. Le modèle est illustré par la

Figure 1 où chaque dimension est représentée par un rectangle (modèle de l'activité, modèle de la localisation, modèle du dispositif, etc.).

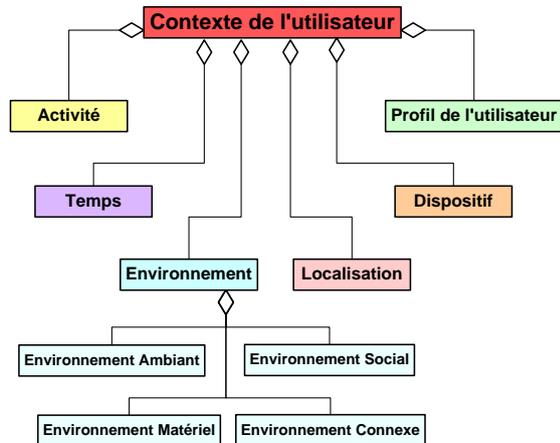


Figure 1 – Le modèle du contexte de l'utilisateur

Nous allons présenter chacune de ces dimensions de manière succincte :

Utilisateur : les informations sur l'utilisateur sont fondamentales dans notre modèle puisque cet utilisateur est au centre du modèle. En effet, la qualité de la définition du profil de l'utilisateur influe grandement sur les informations que l'on va par la suite lui fournir. Ainsi, lors d'une visite d'un site architectural par exemple, nous n'allons pas fournir les mêmes informations ou commentaires à un expert du domaine (un architecte dans notre cas) et à un touriste néophyte dans ce domaine. Au premier, nous pouvons fournir des discussions relatives à l'architecture et à la technicité correspondante tandis qu'au dernier, nous fournirons des informations moins techniques.

Ainsi, dans le modèle proposé, la dimension « profil de l'utilisateur » prend en compte la situation personnelle et/ou professionnelle de l'utilisateur. Cette dimension est organisée et représentée à l'aide d'un modèle de l'utilisateur qui sera spécifique à une application donnée et qui prendra en compte des informations sur la situation de l'utilisateur : son état civil, sa profession, ses centres d'intérêts, ses préférences, son handicap éventuel, sa culture, Cette liste n'est pas exhaustive et peut être complétée selon les besoins d'une application particulière. Nous précisons que le fait de prendre en compte le handicap d'un utilisateur est très important pour notre problématique puisque la

visualisation des informations doit être adaptative. Cette adaptativité peut aussi se faire en fonction de la culture d'une personne, de préférences et de goûts particuliers.

Activité : l'activité d'un utilisateur est importante dans la génération du contexte puisqu'elle permet de savoir ce que l'utilisateur est en train de faire, mais elle est surtout essentielle lors de la communication entre utilisateurs. En effet, un utilisateur ayant comme activité « réunion de travail » serait susceptible de ne pas répondre si un autre utilisateur veut communiquer avec lui de manière synchrone.

Cette dimension est organisée et représentée à l'aide d'un modèle d'activité qui sera spécifique à une application donnée. Dans un contexte de formation, le modèle sera un modèle d'activité pédagogique. Par exemple, dans le modèle d'activité d'apprentissage de Bouzhegoub *et al.* (2007), les informations sont représentées sous forme d'ontologies où sont organisées les activités pédagogiques des apprenants (apprendre, réviser, lire, écrire, résoudre, étudier, discuter, collaborer, etc.).

Dispositif : le dispositif utilisé par l'utilisateur influe sur son contexte puisque l'on ne fournira pas les mêmes informations ou les mêmes moyens de communication à un utilisateur ayant un petit écran (e.g. PDA ou *smartphone*) et à un utilisateur travaillant sur un ordinateur portable. En effet, l'interaction entre les utilisateurs va dépendre de la capacité du réseau qui les relie et du dispositif de chaque utilisateur. Par exemple, si deux utilisateurs ayant un contexte proche veulent communiquer de manière synchrone via une conversation téléphonique cela ne pourrait se faire que si leurs dispositifs respectifs permettent de téléphoner.

Cette dimension est organisée et représentée à l'aide d'un modèle du dispositif qui prendra en compte les informations suivantes:

- le type (exemple : PDA, *Smartphone*, PC fixe etc.),
- le type de connexion possible (exemple : Wifi, connexion filaire, Bluetooth, etc.),

- la vitesse de connexion (exemple : 100,0 Mbits/s, etc.),
- autonomie de la batterie,
- l'espace mémoire,
- la taille de l'écran,
- type d'entrées/sorties (exemple : clavier, écran tactile, micro, écouteur, etc.),
- ...

Temps : la dimension Temps donne une temporalité au contexte. En effet, l'information contextuelle est très dynamique, l'utilisateur changeant souvent de contexte, et celle-ci n'est « valide » qu'à un moment ou dans un intervalle de temps donné. Cette information temporelle est aussi importante pour l'utilisateur à qui l'on fournit des discussions/commentaires, puisqu'elle lui permet de situer la « validité » ou la « fraîcheur » des discussions fournies.

Localisation : la dimension Localisation est primordiale puisqu'elle indique l'endroit où se situe l'utilisateur et le situe spatialement par rapport aux autres utilisateurs. Le modèle de la localisation peut fournir les informations suivantes :

- localisation logique de l'utilisateur (par exemple une page URL)
- localisation physique de l'utilisateur (par exemple, les coordonnées GPS)

Environnement : l'Environnement qui « entoure » l'utilisateur fait partie de son contexte puisqu'au moment où il accomplit sa tâche ou fait son activité, il peut agir sur et interagir avec cet environnement qui l'entoure. Une définition résume bien cette dimension : « *An environment covers the set of objects, persons and events that are peripheral to the current task(s) but that may have an impact on the system and/or the user's behaviour, either now or in the future* » (Calvary et al., 2001).

L'environnement comprend quatre parties distinctes :

- ***l'environnement social*** répertorie les personnes proches « physiquement » de l'utilisateur et avec qui il pourrait éventuellement interagir directement.
- ***l'environnement matériel*** répertorie tous les appareils électroniques ou informatiques proches physiquement de l'utilisateur.
- ***l'environnement connexe*** donne un aperçu des lieux proches de la localisation courante de l'utilisateur (adresse physique ou logique).
- ***l'environnement ambiant*** donne des informations sur la température ambiante, la qualité de l'air, le bruit, la luminosité, etc. Cette partie est importante pour contextualiser les communications. Par exemple, si le niveau sonore ambiant est trop élevé la fonctionnalité téléphone serait déconseillée à l'utilisateur.

Les dimensions du contexte sont reliées entre elles grâce à des relations sémantiques. Elles montrent les éventuelles relations entre les dimensions du contexte. Par exemple, un dispositif est localisé à un endroit donné, une activité est lié à un environnement, etc.

Le modèle proposé sert de base générique pour les communications contextuelles en situation de mobilité. Le modèle a pour objectif d'être exhaustif sur les types d'information prise en compte. Cependant, pour des raisons de coûts de certains capteurs par exemple, un sous ensemble peut être utilisé pour certaines applications.

4 – LE PROTOTYPE MECOCO

Dans cette section, nous présentons le système MeCoCo (*Mediated Contextual Communications*). Nous détaillons dans une première partie l'architecture globale de MeCoCo et dans une deuxième partie un prototype fondé sur ce modèle.

4.1 - Architecture du système MeCoCo

L'architecture générale du système est illustrée sur la Figure 2. Nous distinguons deux parties dans le gestionnaire de contexte (*Context Manager*): une partie du côté client (*Client Context Manager*) et une autre du côté serveur (*Server Context Manager*).

Le *Client Context Manager* se compose de trois parties :

(1) **Interface manager** : qui gère l'interface et la visualisation des informations venant du serveur.

(2) **Information collector** : qui collecte les informations qui vont composer le contexte via des capteurs externes (capteurs de niveau sonore ambiant, de température ambiante ou puces RFID pour connaître l'existence de matériels proches, un GPS pour connaître la localisation,...) ou internes (capteurs pour connaître la vitesse de connexion du dispositif, ...) mais aussi via l'Interface manager.

(3) **Context generator** : qui reçoit les informations de l'*Information collector*, trie et regroupe ces informations pour générer un fichier XML qu'il envoie par la suite au

serveur.

Le *server Context Manager* se compose de deux parties :

(1) **Contextualizer** : qui est la partie la plus importante de notre système puisque c'est elle qui gère le moteur permettant d'avoir les discussions contextuelles.

En effet, le fichier XML « standardisé » envoyé par le client est reçu par ce module dans la partie serveur. Tout d'abord, le *Contextualizer* stocke dans la base de données des contextes (*Contexts DB*), le nouveau contexte C_x de l'utilisateur X qui vient de lui arriver. Ensuite, il prend ce nouveau contexte et via l'*Inquirer*, fait des requêtes dans la base de données des contextes, pour avoir tous les contextes « proches » de celui de l'utilisateur X.

Pour ce faire, la notion de proximité est primordiale puisqu'elle guide le moteur dans sa recherche. Ainsi, nous définissons une proximité entre deux contextes C_x (avec $\langle U_x, A_x, D_x, E_x, L_x, T_x \rangle$) et C_i (avec $\langle U_i, A_i, D_i, E_i, L_i, T_i \rangle$) par la formule suivante :

$$\text{Proximité_Globale}(C_x, C_i) = \sum_{j=1}^6 (\sigma_j * \text{Proximité}(D_x[j], D_i[j]))$$

$$D[j] \in \{U, A, D, E, L, T\}$$

σ_j = coefficient de pondération d'une dimension D

Le calcul de la proximité entre deux dimensions du contexte est complètement dépendant du domaine d'application. Par

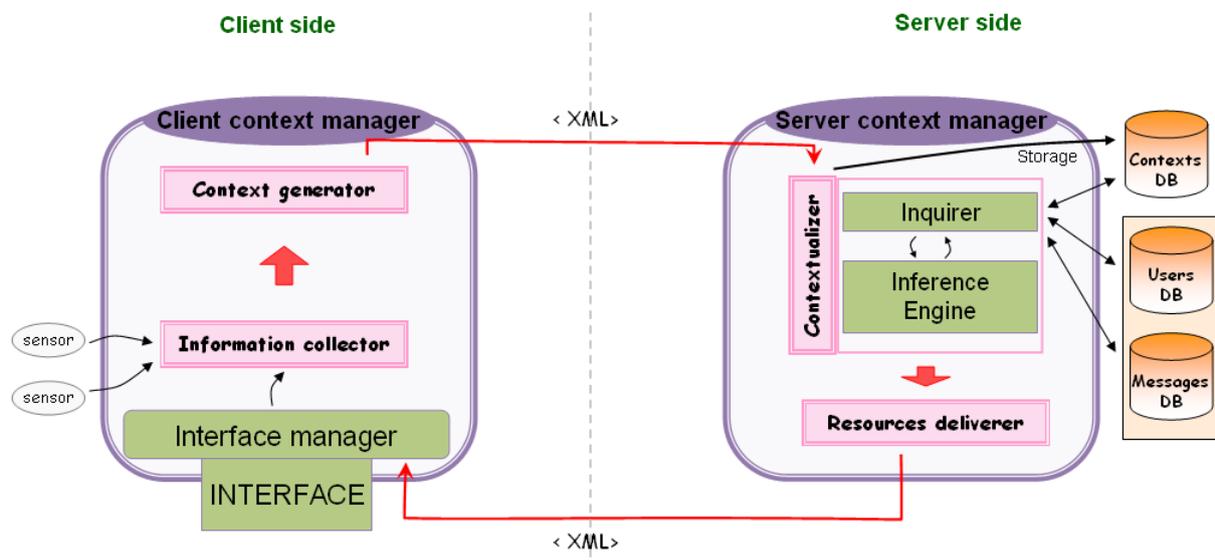


Figure 2 – L'architecture MeCoCo

exemple, la proximité entre deux dimensions T_{x_1} et T_{x_2} peut être considérée comme grande s'il y a une heure d'écart pour une certaine application et comme faible pour une autre application. Pour simplifier, nous avons choisi de ramener la proximité à une valeur décimale comprise entre 0 et 1.

Le coefficient de pondération des dimensions (σ_j) est important car il permet de mettre l'accent sur certaines dimensions par rapport aux autres. Encore une fois, ces coefficients seront établis en fonction de l'application mise en œuvre et des besoins des utilisateurs.

Enfin, après avoir répertorié tous les contextes « proches » de C_x , le *Contextualizer* recherche les personnes correspondantes ainsi que les messages éventuels écrits par ces personnes.

(2) Ressources deliverer : qui s'occupe de renvoyer sous forme XML les données du côté client à l'Interface manager qui va s'occuper de l'affichage des informations.

Le système MeCoCo a été mis en œuvre en se fondant sur l'architecture proposée.

4.2 – Le prototype MeCoCo

Le principal objectif du prototype est de montrer la faisabilité de l'approche proposée et de l'architecture. De plus, nous souhaitons proposer des interfaces homme-machine dédiées aux communications contextuelles en situation de mobilité.

Lors de l'élaboration de MeCoCo, nous nous sommes attelés à la création des différents modules composant le *Client Context Manager* et le *Server Context Manager*. Certaines fonctionnalités du système sont simulées dans le prototype à savoir l'*Information Collector*. Ceci évite de mettre en œuvre tous les capteurs mais n'enlève rien au fonctionnement global du système. Tous les autres modules ont été développés avec des langages tels que PHP, Javascript et Flash en les combinant avec des fichiers HTML et XML, ainsi qu'avec des bases de données relationnelles (SQL). Les technologies AJAX ont été utilisées pour favoriser des interfaces dynamiques.

L'interface de MeCoco offre à l'utilisateur différentes vues sur les informations. Deux

informations contextuelles sont proposées à l'utilisateur :

- Les contacts contextuels (i.e. les utilisateurs) qui ont été ou qui sont dans un contexte proche de lui.
- Les messages contextuels postés par ces contacts.

Les contacts présentés à l'utilisateur courant ont au moins une dimension commune avec lui. Ces contacts sont représentés par un avatar avec des accessoires bien déterminés, chacun correspondant à une dimension. La Figure 3 montre un avatar et la Figure 4 un avatar ayant posté un message.

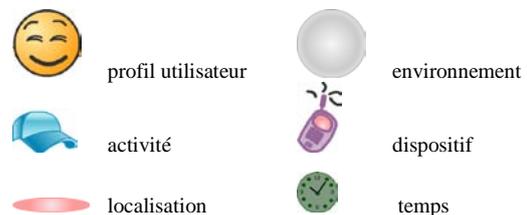


Figure 3 – Avatar



Figure 4 – Avatar ayant posté un message

Légende des 6 dimensions:



Si une dimension d'un contact n'est pas similaire ou proche de celle de l'utilisateur, l'icône correspondant n'est pas colorée sur l'avatar (Exemple : 😊 s'il y a proximité de profil et 😐 sinon).

Dans le prototype, nous avons volontairement simplifié le calcul des proximités entre dimensions du contexte. Ainsi, la distance entre deux éléments d'une dimension est soit 1 si ces deux éléments sont similaires et 0 sinon. Le prototype a été choisi en prenant comme scénario un apprentissage culturel lors de la visite d'un musée. Ainsi, dans ce scénario, la proximité est égale à 1 dans les cas suivants :

- Localisation : la même pièce dans le musée (« Salle Picasso », « Salle Monet », ...).

- **Activité** : la même activité (visiter le musée, guider les visiteurs, ...).
- **Profil** : le même profil général (étudiant, touriste, guide, ...).
- **Temps** : connecté en même temps (connecté ou non).
- **Dispositif** : le même type de dispositif (PDA, smartphone, ordinateur portable, ...).
- **Environnement** : le même bruit ambiant (silencieux, bruyant, ...).

De plus, nous avons choisi de prendre le même coefficient de pondération (=1) pour les 6 dimensions. Ainsi, la proximité globale peut prendre une valeur entre 0 et 6. Nous sommes conscients que ces choix ne sont pas forcément les plus réalistes mais ils conviennent pour tester ce prototype.

Nous avons mis en place 3 vues IHM dans notre prototype :

(1) Une vue globale qui place les informations contextuelles sur un radar selon une échelle de un à six. Un contact dans un contexte C_i est placé sur le radar selon les valeurs de proximité de ses dimensions. Si un contexte C_i obtient une proximité globale de 6, c'est qu'il a pour chaque dimension une proximité de 1 et par conséquent le contact correspondant a le même profil, le même dispositif, le même environnement, la même activité, la même localisation et qu'il est connecté en même temps que l'utilisateur courant (celui à qui on donne les informations contextuelles).

La Figure 5 nous montre l'interface dans sa vue globale où l'utilisateur courant Matthieu Lemarchand voit son contexte apparaître dans la bulle d'information située en dessous de son image à l'extrémité gauche du radar. Cela donne une certaine transparence au système qui n'est pas une « boîte noire » pour l'utilisateur puisqu'ainsi il lui est possible de comprendre le fonctionnement du système et donc de l'adopter par la suite. Cet utilisateur courant pourra comparer son contexte à celui d'un contact en passant la souris sur l'avatar correspondant à celui-ci. Il pourra aussi voir les informations concernant ce contact à savoir son nom, son prénom, etc., en cliquant sur l'avatar. Pour lire un commentaire laissé par un

contact, il suffit à l'utilisateur de cliquer sur l'icône du message.

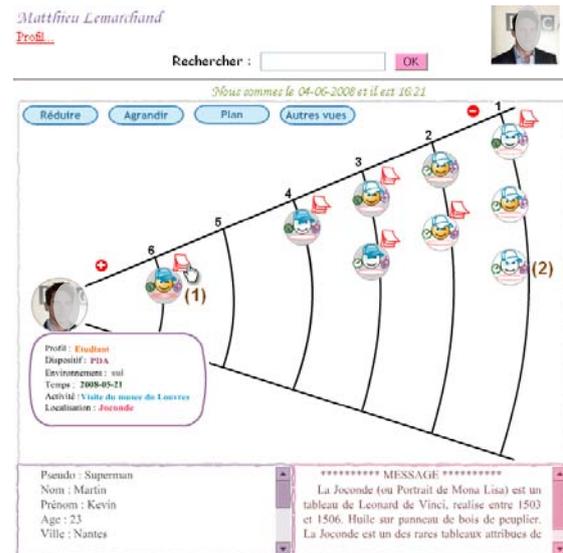


Figure 5 – Interface globale sous forme de radar

Dans la Figure 5, l'utilisateur (1) représente un utilisateur ayant 6 dimensions identiques à celles de l'utilisateur courant et ayant posté un message. L'utilisateur (2) représente un utilisateur ayant une seule dimension (Dispositif) identique à celle de l'utilisateur courant et n'ayant pas écrit de message.

(2) Une vue par rapport au temps qui place les informations contextuelles par rapport à deux axes dont l'ordonnée correspond au temps et l'abscisse à la proximité globale sur les cinq autres dimensions du contexte. Cette fois-ci on prend seulement cinq dimensions pour calculer la proximité globale, puisque l'on veut voir le contexte par rapport à la dimension temps, on n'intègre donc plus celle-ci dans ce calcul. Contrairement à la vue globale où il n'était pas possible pour un contact d'avoir une proximité globale de 0 sur 6, il est envisageable qu'un contact ait une proximité de 0 dans la vue temps cela signifie qu'il n'y a que la dimension temps qui soit proche entre l'utilisateur courant et ce contact.

Comme le montre la Figure 6, sur l'axe des ordonnées où l'on a le temps (maintenant, aujourd'hui, hier,...), nous avons mis en place un modèle du temps très simple puisque l'on peut avoir une relation d'ordre dans cette dimension.

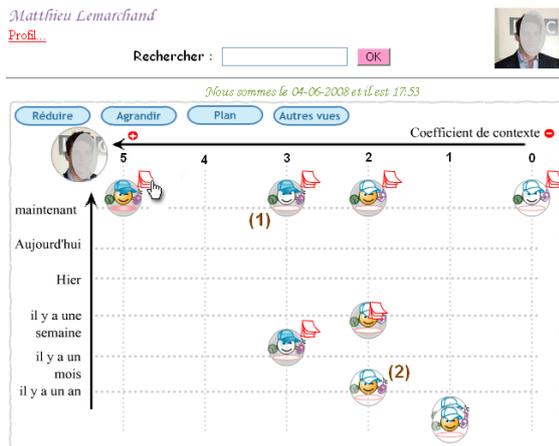


Figure 6 – Interface de visualisation par rapport au temps

Dans la Figure 6, l'utilisateur (1) représente un utilisateur qui est connecté en même temps que l'utilisateur courant, qui a 3 dimensions (Activité, Dispositif et Environnement) identiques aux siennes et qui a posté un message. L'utilisateur (2) représente un utilisateur ayant deux dimensions (Profil et Dispositif) identiques à celles de l'utilisateur courant mais qui s'est connecté il y a plus d'un an.

(3) Une vue par rapport à la localisation qui place les informations contextuelles par rapport à deux axes dont l'ordonnée correspond à la localisation et l'abscisse à la proximité. De même que dans la vue Temps, le calcul de la proximité globale se fait sur 5 dimensions, on n'intègre pas la localisation, puisque l'objectif est de voir les autres dimensions par rapport à celle-ci.

Pour le prototype, nous avons créé un modèle de localisation fondée sur les peintres et les principaux mouvements (Renaissance, Impressionnisme, cubisme, ...). Ce modèle n'est pas détaillé ici. C'est une taxonomie avec différents niveaux : peinture, peintre, mouvement. En fait, cette classification correspond à l'organisation des salles de notre musée utilisé pour le prototype. Par conséquent, avec la vue par rapport à la localisation, un utilisateur peut facilement voir la proximité (physique ou virtuelle) des contacts en fonction de la peinture qu'il regarde.

Par exemple, sur la Figure 7, l'utilisateur courant se situe devant La Joconde. Il peut voir les contacts qui sont aussi devant ce tableau ou

qui ont laissé un message sur ce tableau. De même, il peut avoir les mêmes informations pour l'ensemble des tableaux de Léonard de Vinci ou pour la partie du musée concernant la Renaissance. Les autres dimensions sont toujours représentées sur chaque contact.

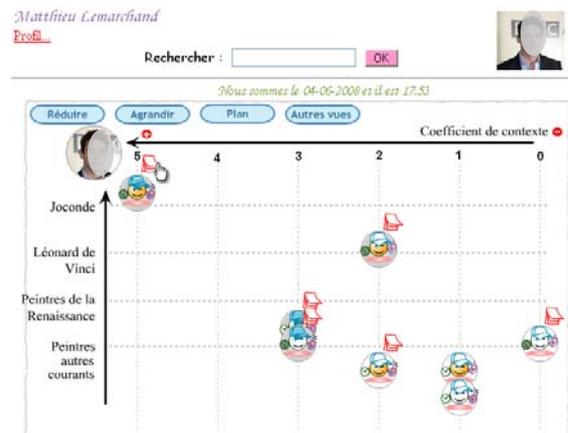


Figure 7 – Interface de visualisation par rapport à la localisation

Le prototype montre qu'il est possible de définir la proximité d'une dimension avec un grain assez fin (exemple donné pour le temps et la localisation). Les différentes vues proposées devront être choisies en fonction de l'application cible et des objectifs spécifiques.

5 – CONCLUSION ET PERSPECTIVES

L'objectif de cette recherche est de proposer un modèle générique de contexte centré utilisateur pour favoriser les communications médiatisées entre humains en situation de mobilité. Ces communications contextuelles sont particulièrement utiles dans une situation d'apprentissage pour trouver des personnes et messages pertinents. Nous avons pu donner une définition du contexte qui prend en compte six dimensions essentielles : le profil de l'utilisateur, la temporalité, le dispositif qu'il utilise, sa localisation, son environnement et l'activité qu'il est en train de faire à un moment donné. Le domaine de généralité du modèle est celui des CMC (Communications Médiatisées Contextuelles).

La conception d'un prototype nous a permis de valider notre modèle tout en ayant un outil de base pour permettre une visualisation originale du contexte qui aide l'utilisateur dans ses choix de discussions ou de ses contacts. Le prototype MeCoCo offre une interface facile à manipuler et intuitive pour l'utilisateur. En effet, grâce

aux différentes vues proposées, le système laisse une grande liberté à l'utilisateur par rapport aux discussions qu'il veut voir mais aussi quant à la dimension qu'il veut mettre en avant. La vue globale de MeCoCo donne une visualisation immédiate de la proximité du contexte des contacts par rapport à celui de l'utilisateur courant. En effet, d'un seul coup d'œil, l'utilisateur pourra voir, pour un contact donné les dimensions similaires aux siennes. Le problème de la concurrence des dimensions est résolu par ce mode de visualisation car le système n'a pas à arbitrer cette concurrence. En effet, lorsqu'un contexte C_i a une proximité de $4/6$ et qu'un autre contexte C_j a la même proximité, la question de savoir si on affiche C_j ou C_i ne se pose pas puisque l'on affichera les deux contextes. Cet affichage se fera selon la proximité de chacune des dimensions de ces contextes par rapport au contexte de l'utilisateur courant. Ce sera alors à l'utilisateur de choisir quelles informations sont importantes dans son cas et de se retrouver ainsi dirigé en fonction de ses priorités. L'avantage principal pour l'utilisateur est qu'il va pouvoir préférer entrer en contact avec un utilisateur qui n'a pas forcément la plus grande proximité de contexte mais dont une dimension lui semble plus importante pour ses besoins du moment.

Les vues Temps ou Localisation de MeCoCo viennent en complément de la vue globale puisqu'elles permettent de favoriser une dimension par rapport aux cinq autres. En effet, ces vues permettent de mettre en évidence une dimension et de voir les discussions et les contacts, respectivement, selon l'éloignement temporel ou selon leur localisation (plus ou moins proche physiquement ou logiquement de l'utilisateur). À terme, nous comptons proposer une vue spécifique pour isoler chaque dimension par rapport aux autres (six vues spécifiques au total).

Enfin, ce type de travail ne peut se faire sans se poser des questions quant à la confidentialité des données contextuelles qui contiennent des informations personnelles. Cela soulève des questions d'ordre d'éthique tout d'abord puisque la divulgation de telles données peut nuire au principe de protection de la vie privée des individus. Par ailleurs, cela pose aussi la question du consentement de l'utilisateur quant à la divulgation de ces informations

personnelles. Ainsi dans un souci de confidentialité de données et de protection de la vie privée des utilisateurs, nous envisageons la mise en place de métadonnées pour notre modèle, qui traiteraient pour chaque dimension et plus particulièrement sur la dimension 'Profil Utilisateur', du niveau de confidentialité que l'utilisateur veut mettre. L'idée est de jouer la transparence complète et de laisser le choix à l'utilisateur de voir et de montrer ce qu'il veut à tout moment.

BIBLIOGRAPHIE

- Basaeed, E., Berri, J., Zemerly, J., Benlamri, R. (2007). Web-based Context-Aware m-Learning Architecture. *International Journal of Interactive Mobile Technologies (iJIM)*, Vol. 1, n°1, pp. 1-6.
- Bouzeghoub, A., Do, K., Lecocq, C. (2007). A Situation-Based Delivery of Learning Resources in Pervasive Learning. In *Creating New Learning Experiences on a Global Scale*, pp. 450-456.
- Calvary, G., Coutaz, J., Thevenin, D. (2001). A Unifying Reference Framework for the Development of Plastic User Interfaces. In *Engineering for Human-Computer Interaction*, Lecture Notes in Computer Science, pp. 173-192.
- Calvary, G., Coutaz, J., Thevenin, D., Limbourg, Q., Bouillon, L., Vanderdonckt, J. (2003). A Unifying Reference Framework for multi-target user interfaces. *Interacting with Computers*, Vol. 15, n°3, pp. 289-308.
- Dey, A. K. (2001). Understanding and Using Context. *Personal Ubiquitous Comput.*, Vol. 5, n°1, pp. 4-7.
- Dey, A. K., Abowd, G. D. (2000). CybreMinder: A Context-Aware System for Supporting Reminders, pp. 172-186.
- George, S., Labas, H. (2008). E-learning standards as a basis for contextual forums design. *Computers in Human Behavior*, Vol. 24, n°2, pp. 138-152.
- Kirsch-Pinheiro, M., Villanova-Oliver, M., Gensel, J., Martin, H. (2005). Context-aware filtering for collaborative web systems: adapting the awareness information to the

user's context. In *Proceedings of the 2005 ACM symposium on Applied computing*, Santa Fe, New Mexico: ACM, pp. 1668-1673.

Law, C., Zhang, X., Chan, S., Wang, C. (2006). Smart Instant Messenger in Pervasive Computing Environments. In *Advances in Grid and Pervasive Computing*, pp. 32-41.

Ranganathan, A., Campbell, R., Ravi, A., Mahajan, A. (2002). ConChat: a context-aware chat program. *Pervasive Computing, IEEE*, Vol. 1, n°3, pp. 51-57.

Wang, X., Zhang, D., Gu, T., Pung, H. (2004). Ontology based context modeling and reasoning using OWL. In *Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, pp. 18-22.