

# EMMA : Modèle Utilisateur pour la Plasticité des Interfaces Homme-Machine en Mobilité

Vincent Ganneau<sup>1,2</sup>

Gaëlle Calvary<sup>2</sup>

Rachel Demumieux<sup>1</sup>

<sup>1</sup> France Télécom Division R&D  
2, avenue Pierre Marzin  
22307, Lannion Cedex, France  
+33 (0)2 96 05 94 65

{vincent.ganneau, rachel.demumieux}  
@orange-ftgroup.com

<sup>2</sup> Laboratoire LIG  
385, rue de la Bibliothèque, BP 53  
38041, Grenoble Cedex, France  
+33 (0)4 76 51 48 54

{vincent.ganneau, gaelle.calvary}  
@imag.fr

## ABSTRACT

Adapting User Interfaces (UI) while in mobility remains challenging as contexts of use identification and changes can obviously not be finely envisioned at design time. However, user's tasks, habits and preferences may be context dependent. As a result, there is a need for managing context-aware adaptation at runtime. This paper describes EMMA (Embedded Manager for Mobile Adaptation), a running system that gathers data in mobility, learns key contexts of use, and provides the end-user with relevant adaptation. The process is based on a bayesian user model embedded on a Smartphone running under Windows Mobile operating system. It is placed under the end-user's control through a dedicated UI, called meta-UI. Besides this end-user's partner role, EMMA can serve as a designer's partner tool for identifying the *key* contexts of use in the wild.

## RESUME

Dès lors que l'utilisateur est mobile, les contextes d'usage ne peuvent pas être précisément identifiés à la conception. Or, les tâches de l'utilisateur, ses habitudes et ses préférences peuvent dépendre du contexte dans lequel il évolue. Les Interfaces Homme-Machine (IHM) doivent, en conséquence, pouvoir s'adapter au contexte d'usage en mobilité. Cet article présente EMMA (Embedded Manager for Mobile Adaptation), un système sensible au contexte développé pour collecter des données d'usage en mobilité, identifier les contextes d'usage clés et offrir une adaptation pertinente sur changement de contexte d'usage. EMMA s'appuie sur un modèle utilisateur bayésien embarqué sur des Smartphones équipés de Windows Mobile. L'adaptation est placée sous le contrôle de l'utilisateur via une IHM additionnelle, appelée méta-IHM. Au-delà de son rôle de partenaire pour l'utilisateur final, EMMA est aussi un outil pour le concepteur : il l'aide à identifier les *bons* contextes d'usages sur le terrain.

## Categories and Subject Descriptors

D.2.2 [Software Engineering]: Design Tools and Techniques –

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, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

*UbiMob'08*, May 28–30, 2008, Saint-Malo, France.

Copyright 2008 ACM 978-1-59593-980-7/08/05...\$5.00.

*User Interfaces*; G.3 [Probability and Statistics]: *Probabilistic algorithms*; H.5.2 [Information Interfaces and Presentation]: *User Interfaces – User-centered design*; I.2.6 [Artificial Intelligence]: *Learning – Parameter learning*

## General Terms

Algorithms, Design, Human Factors.

## Mots clés

Modèle utilisateur, mobilité, Smartphone, plasticité, adaptation, contexte d'usage, réseau bayésien, apprentissage, classification.

## Keywords

User modeling, mobility, Smartphone, plasticity, adaptation, context of use, bayesian network, learning, clustering.

## 1. INTRODUCTION

Le téléphone mobile est aujourd'hui un objet utilisé par des millions de personnes à travers le monde. Il a révolutionné notre façon de communiquer au quotidien. Pourtant, certaines tâches comme l'envoi de message ou l'ajout d'un contact demeurent problématiques pour certains utilisateurs. Ces difficultés d'utilisation s'expliquent en partie par les problèmes de navigation et de saisie dus aux contraintes de ces terminaux (arborescences complexes, taille réduite de l'écran, clavier restreint) [1]. En outre, les Interfaces Homme-Machine (IHM) statiques de nos téléphones mobiles ne répondent pas à des besoins différents entre utilisateurs, à l'exception de quelques éléments personnalisables (écran d'accueil, raccourcis, etc.). Un moyen d'améliorer l'interaction est d'offrir une adaptation dynamique en fonction des usages de l'utilisateur en contexte.

En informatique ambiante, l'adaptation au contexte a été largement traitée [8] [18]. La diversité et la variabilité du contexte d'usage (utilisateur, plate-forme, environnement) ont motivé l'introduction de la propriété de *plasticité* des IHM, définie comme la capacité des IHM à s'adapter à leur contexte d'usage dans le respect de leur utilisabilité [20]. Ici, nous nous intéressons au cas particulier de l'utilisateur en mobilité. La plate-forme de référence est le téléphone mobile. Nos travaux sont motivés par les constats suivants :

#1. Les tâches de l'utilisateur peuvent varier en fonction du contexte d'usage. Elles dépendent de l'utilisateur (par

exemple, la messagerie instantanée est plutôt utilisée par les adolescents), et de l'environnement (par exemple, les jeux permettent de passer le temps dans les transports publics, le calendrier est un outil au travail) ;

- #2. Les changements de contexte introduisent des tâches répétitives (par exemple, basculer son téléphone en profil silencieux au commencement d'une réunion). Ces tâches répétitives sont d'autant plus contraignantes qu'elles impliquent les problèmes d'utilisabilité évoqués auparavant ;
- #3. Identifier les *bons* contextes d'usage dès la conception est utopique. Par essence, la modélisation du contexte reste imparfaite et incomplète. Limiter les contextes d'usage à des contextes approximatifs définis à la conception peut s'avérer problématique.

Ces observations appellent deux requis :

- #1. Identifier les contextes d'usage clés à l'exécution ;
- #2. Proposer des adaptations sur la base des actions utilisateur.

Des précédents travaux sur les modèles utilisateur [10] [12], la programmation par l'exemple [6] [9] [17] et la gestion de rendez-vous [14], sont d'un grand intérêt pour traiter ces requis. Sur téléphone mobile, des travaux plus récents ont appliqué la programmation par l'exemple à l'apprentissage des changements de profil en mobilité [3] ainsi qu'à la création de raccourcis permettant d'achever des tâches répétitives [4]. D'autres travaux ont expérimenté l'adaptativité au contexte pour contrôler les paramètres du téléphone selon l'activité de l'utilisateur (occupé, interruptible) [19] ou encore adapter l'IHM. Pour exemple, le projet Adamos propose un menu adaptatif en fonction de trois contextes identifiés à la conception (maison, travail, loisir) [16]. Le menu se divise en deux parties : une partie statique donnant accès aux six fonctions principales du téléphone mobile ; une partie adaptative mettant en avant trois fonctionnalités supposées pertinentes dans le contexte courant.

Dans ce travail, nous explorons l'adaptation au contexte et la programmation par l'exemple au travers d'un modèle utilisateur. Nous proposons EMMA (Embedded Manager for Mobile Adaptation), un système implémenté pour répondre aux deux requis formulés. EMMA est un outil partenaire permettant à l'utilisateur final de conformer son téléphone mobile à son contexte d'usage. EMMA peut également servir d'outil pour le concepteur désirant observer les usages réels sur le terrain et par là même identifier les contextes d'usage clés. L'article se structure en deux parties. Il décrit EMMA des points de vue de l'utilisateur final (section 2) puis du système (section 3).

## 2. EMMA EN ACTION

Qui n'a jamais oublié de couper le son de son téléphone mobile en réunion? Qui n'a jamais manqué un coup de fil parce que son téléphone était resté en mode silencieux? Probablement personne car changer le profil du téléphone est une tâche articuloire qui, en outre, dépend du contexte dans lequel on se trouve. Ces tâches de paramétrage sont non seulement répétitives mais également longues et pénibles en regard des problèmes de navigation rencontrés sur téléphone mobile [20]. En faisant l'hypothèse que l'on peut apprendre à partir de données contextuelles et d'usage recueillies en mobilité, EMMA étudie comment déléguer au

système l'adaptation au contexte sous l'éventuel contrôle de l'utilisateur final.

EMMA est un logiciel pour Smartphones équipés de Windows Mobile. EMMA offre une adaptation basée sur les usages réels recueillis à l'exécution. Les adaptations implémentées sont listées ci-après :

- Réorganisation du menu Démarrer ;
- Commutation du profil du téléphone ;
- Changement des paramètres de personnalisation.

Afin d'illustrer les principes d'EMMA, étudions un scénario. Soit Bob, un ingénieur informaticien qui partage sa semaine entre son bureau et sa maison. Au travail, Bob utilise son téléphone de façon professionnelle. Il l'utilise principalement pour appeler des contacts et comme un gestionnaire de rappel d'événements, qu'il synchronise avec son ordinateur via une connexion sans-fil. Pendant le travail, Bob n'apprécie pas d'être dérangé car il est souvent en réunion. Aussi, lorsqu'il arrive au travail, EMMA bascule automatiquement le profil de son téléphone en mode silencieux. EMMA réordonne également le menu Démarrer du téléphone de façon à mieux calquer les besoins de Bob dans ce contexte. Enfin, ses préférences en matière de personnalisation (thème graphique, image de fond) sont appliquées (Figure 1).



**Figure 1. Ecran d'accueil du téléphone et réorganisation du menu Démarrer dans le contexte « Travail ». Le profil du téléphone bascule en mode silencieux.**

Après le travail, Bob rentre chez lui. EMMA trace les déplacements de Bob via GSM (Global System for Mobile communications) et détecte un changement dans sa localisation géographique. Le changement de contexte est notifié à Bob qui peut alors accepter ou refuser la proposition d'EMMA d'entrer dans le contexte « Maison » (partie gauche de la Figure 2). Bob accepte le changement ce qui amène EMMA à lui proposer d'adapter son téléphone au contexte « Maison » : la réorganisation du menu Démarrer est placée sous son contrôle (partie droite de la Figure 2 et partie gauche de la Figure 3) tandis que les changements de profil et de personnalisation sont effectués automatiquement (partie droite de la Figure 3). Lors de la réorganisation du menu Démarrer, EMMA peut avoir à créer des raccourcis afin de réduire la navigation, comme proposé dans [4]. Combiner la création de raccourcis à la réorganisation du menu Démarrer peut économiser un grand nombre d'appuis de touche.



Figure 2. Les changements de contexte et les adaptations sont placés sous le contrôle de l'utilisateur final.



Figure 3. Dans le contexte « Maison », le menu Démarrer est réorganisé et le profil du téléphone bascule en mode normal.

Comme illustré dans la Figure 2, les changements de contexte et les adaptations peuvent être placés sous le contrôle de l'utilisateur via des boîtes de dialogue. Cette stratégie n'est pas systématique : elle s'applique lorsque l'utilisateur demande explicitement à confirmer ou lorsqu'EMMA n'est pas pleinement certain de ses propositions. La Figure 4 (partie gauche) montre l'IHM permettant à l'utilisateur de choisir entre les modes automatique et négocié : lorsque le mode automatique est actif, seul un retour d'information est donné à l'utilisateur (partie droite).

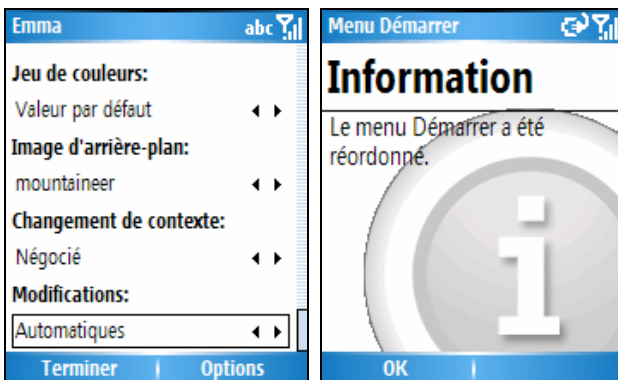


Figure 4. IHM pour contrôler la stratégie d'adaptation.

En plus d'un retour graphique, EMMA supporte aussi un retour d'information sonore. Ce deuxième retour est basé sur une synthèse vocale. Il est également placé sous le contrôle de l'utilisateur (Figure 5). Le retour vocal peut s'avérer pertinent car les retours d'information graphiques disparaissent après quelques secondes en fonction de l'activité courante de l'utilisateur (occupé, interruptible) et peuvent en conséquence être manqués.

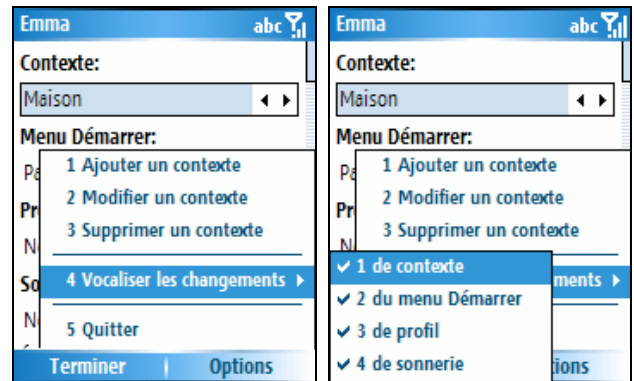


Figure 5. IHM pour activer ou désactiver le retour vocal.

EMMA est développé comme un plug-in pour l'écran d'accueil du téléphone, ce qui permet à l'utilisateur de visualiser d'un simple coup d'œil le contexte dans lequel il se trouve et de changer manuellement de contexte. Il peut aussi accéder à l'IHM d'EMMA depuis le plug-in ou le menu Démarrer du téléphone. Cette IHM permet à l'utilisateur d'ajouter (partie gauche de la Figure 6), modifier et supprimer des contextes clés. Elle rend aussi possible la personnalisation et le déclenchement de l'adaptation dans un contexte donné (partie droite de la Figure 6).

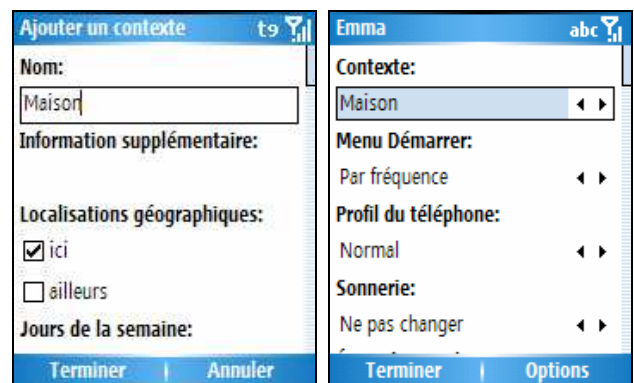


Figure 6. Ajout de contexte et paramétrage de l'adaptation.

La section suivante décrit EMMA d'un point de vue système.

### 3. ARCHITECTURE LOGICIELLE

Nous articulons la description technique d'EMMA autour de son architecture logicielle. Une décomposition fonctionnelle est donnée en sous-section 3.1. La collecte des données puis leur exploitation sont ensuite examinées dans les sous-sections suivantes.

### 3.1 Décomposition Fonctionnelle

La décomposition fonctionnelle d'EMMA (Figure 7) est une révision de la décomposition proposée par [2] pour l'ingénierie d'IHM plastiques. Cinq fonctions sont identifiées :

- L'*IHM* dénote l'IHM du téléphone avec lequel l'utilisateur interagit. Les interactions de l'utilisateur sont collectées dans l'*historique de l'interaction* (voir 3.2) ;
- L'*Observateur du contexte d'usage* est en charge de la reconnaissance des variations pouvant survenir dans le contexte d'usage. Ces changements sont collectés dans l'*historique du contexte* (voir 3.2). Tout changement est communiqué au *Moteur d'adaptation* ;
- Le *Moteur d'adaptation* est la fonction centrale de la décomposition. Il gère l'intégralité du processus d'adaptation en mettant en relation les différentes fonctions. Lorsqu'un changement survient dans le contexte, le *Moteur d'adaptation* sollicite le *Modèle utilisateur* pour déterminer quelles adaptations conviendraient au nouveau contexte. Si l'adaptation doit être négociée, le *Moteur d'adaptation* sollicite la *Méta-IHM* pour gérer la négociation avec l'utilisateur final. C'est le *Moteur d'adaptation* qui à terme met en œuvre l'adaptation de l'*IHM* lorsque celle-ci est nécessaire ;
- Le *Modèle utilisateur* maintient la connaissance du système à propos du contexte d'usage ainsi que les interactions, les habitudes et les préférences de l'utilisateur final. Le *Modèle utilisateur* embarque des mécanismes pour l'apprentissage et l'inférence à partir d'observations. Il aide à l'identification du contexte d'usage ainsi qu'au calcul des adaptations à effectuer en cas de changement de contexte (voir 3.3) ;
- La *Méta-IHM* est en charge de rendre le processus d'adaptation observable et contrôlable par l'utilisateur final [5]. En pratique, les méta-IHM et IHM peuvent être tissés. C'est le cas du plug-in (partie gauche de la Figure 1) : le contexte d'usage courant est affiché au sein de l'IHM du téléphone. Ce n'est pas le cas dans la Figure 2 : la méta-IHM a sa fenêtre propre. Les actions de l'utilisateur entreprises via la méta-IHM reflètent ses *préférences de plasticité*. Ces préférences peuvent avoir un impact sur la négociation des adaptations futures. La méta-IHM a été illustrée dans la section précédente. Elle n'est pas décrite dans cette partie technique.

La collecte des données en mobilité est un point clé. Il fait l'objet de la sous-section suivante.

### 3.2 Collecte de Données en Mobilité

La collecte de données en mobilité a déjà fait l'objet d'études. Pour exemple, [7] expérimente la collecte de données sur téléphone mobile à l'aide d'un outil embarqué recueillant des données d'usage objectives sur le terrain. L'outil observe les fonctionnalités utilisées, leur temps d'utilisation et leur fréquence, ainsi que la navigation de l'utilisateur. Comme expliqué précédemment, EMMA doit collecter trois types de données pour alimenter le modèle utilisateur : des données contextuelles, des données d'interaction et des données de plasticité. Nous examinons chacune d'elles ci-dessous.

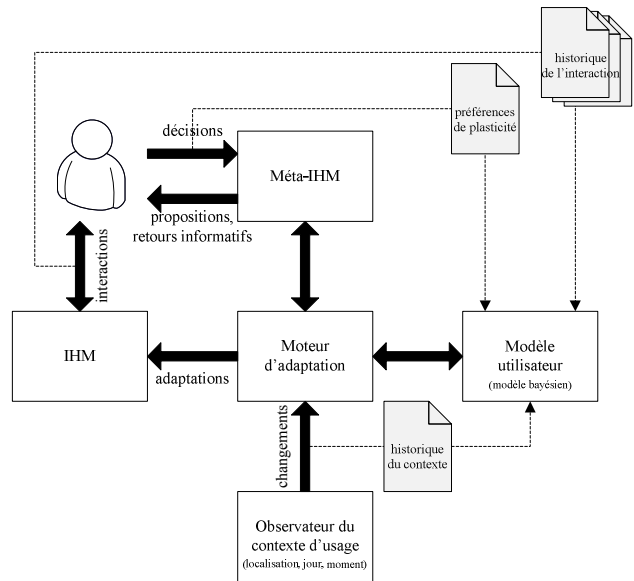


Figure 7. Décomposition fonctionnelle d'EMMA.

#### 3.2.1 Collecte des données contextuelles

EMMA collecte deux types de changements pouvant survenir dans le contexte d'usage : des changements dans le temps (jour de la semaine, moment de la journée) et dans l'espace (localisation géographique). Par expérience et sur la base d'études précédentes sur les usages en mobilité, nous supposons que ces changements agissent sur les habitudes et préférences de l'utilisateur, comme illustré dans le scénario de la section 2. Ces données contribuent à l'identification du contexte par le modèle utilisateur.

La sonde permettant de capturer les changements survenant dans le contexte est développée en C#. Les changements dans le temps sont notifiés via des objets `SystemState` auxquels nous avons ajouté nos propres procédures de rappel d'évènement :

```
protected SystemState Day;
protected SystemState Time;
[ ... ]
Day = new SystemState(SystemProperty.Date);
Time = new SystemState(SystemProperty.Time);
Day.Changed +=
new ChangeEventHandler(Day_Changed);
Time.Changed +=
new ChangeEventHandler(Time_Changed);
[ ... ]
void Day_Changed
(object sender, ChangeEventArgs args) { [ ... ] }
void Time_Changed
(object sender, ChangeEventArgs args) { [ ... ] }
```

La capture de la localisation géographique a fait l'objet d'autres travaux [13]. Via GSM, nous utilisons l'antenne relais la plus proche pour détecter les changements de localisation de l'utilisateur. A tout moment, le téléphone mobile est connecté à une antenne relais sauf lorsque l'utilisateur se trouve dans une zone où il ne capte pas. Chaque antenne relais est identifiée par son

*Cell Identifier* (CID) et son *Location Area Code* (LAC). Nous utilisons la *Radio Interface Layer* (RIL) accessible sur les plateformes Windows Mobile pour détecter les changements d'antenne. Chaque changement est contextualisé avec les informations temporelles dans lesquelles il survient. Nous découpons les moments de la journée en cinq intervalles de temps de la manière suivante : nuit (0h-6h), petit matin (6h-8h), matinée (8h-12h), après-midi (12h-18h), et soir (18h-24h). Le Tableau 1 montre les changements d'antenne collectés pendant un jeudi.

**Tableau 1. Changements d'antenne collectés sur une journée**

Localisation	Jour	Moment
CID25550LAC4354	Jeudi	matinée
CID22057LAC4354	Jeudi	matinée
CID40506LAC4354	Jeudi	matinée
CID40511LAC4354	Jeudi	matinée
CID58063LAC4354	Jeudi	matinée
CID58063LAC4354	Jeudi	matinée
CID40511LAC4354	Jeudi	après-midi
CID40506LAC4354	Jeudi	après-midi
CID40511LAC4354	Jeudi	après-midi
CID58063LAC4354	Jeudi	après-midi
CID40506LAC4354	Jeudi	après-midi
CID40511LAC4354	Jeudi	après-midi
CID40506LAC4354	Jeudi	soir
CID29075LAC4354	Jeudi	soir
CID64457LAC4354	Jeudi	soir
CID22057LAC4354	Jeudi	soir

Les données du Tableau 1 sont collectées par l'*Observateur du contexte d'usage*. Elles sont transmises au *Moteur d'adaptation* et stockées dans l'*historique du contexte*. Lorsque le contexte courant correspond à un contexte identifié comme clé par le *Modèle utilisateur*, le *Moteur d'adaptation* enrichit le Tableau 1 avec le contexte clé correspondant, comme le montre le Tableau 2. Le point d'interrogation marque le fait que le contexte d'usage courant ne correspond à aucun contexte clé pour le moment. Ce type de symbole est souvent utilisé pour marquer l'incomplétude des données dans les algorithmes d'apprentissage.

**Tableau 2. Changements de contexte clé en mobilité**

Localisation	Jour	Moment	Contexte
CID40506LAC4354	Mardi	matinée	Travail
CID22057LAC4354	Mardi	soir	Maison
CID40511LAC4354	Mercredi	matinée	Travail
CID58063LAC4354	Mercredi	matinée	?
CID40511LAC4354	Mercredi	après-midi	Travail
CID25550LAC4354	Mercredi	après-midi	Maison
CID40511LAC4354	Jeudi	matinée	Travail

### 3.2.2 Collecte des données d'interaction

EMMA capture deux types de tâches sur le téléphone mobile : des tâches applicatives (messagerie, appels, jeux, etc.) et des tâches de personnalisation (profil du téléphone, look and feel, etc.).

Aujourd'hui, les téléphones offrent un nombre croissant d'applications, allant de la vidéo caméra à la télévision. Pour exemple, le SPV E650 équipé de Windows Mobile 6.0 offre 34 applications de base contre 27 pour le SPV C100 équipé de Windows Mobile 5.0, sans compter que ces Smartphones permettent en plus à l'utilisateur d'installer des applications supplémentaires à prendre en compte dans notre capture des interactions utilisateur. Chaque application graphique est généralement accessible depuis un raccourci situé dans le menu Démarrer du téléphone. EMMA collecte l'historique de l'interaction via un événement système notifiant un changement d'application active :

```
protected SystemState ActiveApplication;
[... ]
ActiveApplication = new
SystemState(SystemProperty.ActiveApplication);
ActiveApplication.Changed += new
ChangeEventHandler(ActiveApplication_Changed);
[... ]
void ActiveApplication_Changed
(object sender, EventArgs args) { [... ] }
```

Ainsi, l'historique de l'interaction recueille toutes les applications que l'utilisateur a utilisées (Tableau 3). Ces observations sont estampillées d'informations temporelles situant l'événement dans le temps. Ces données sont utilisées pour réorganiser le menu Démarrer.

**Tableau 3. Historique des applications utilisées en mobilité**

Localisation	Jour	Moment	Application
CID40506LAC4354	Jeudi	matinée	Paramètres
CID40511LAC4354	Jeudi	matinée	Explorateur
CID40511LAC4354	Jeudi	matinée	Calendrier
CID40511LAC4354	Jeudi	matinée	Contacts
CID58063LAC4354	Jeudi	matinée	Appels
CID40511LAC4354	Jeudi	après-midi	Calendrier
CID40506LAC4354	Jeudi	après-midi	Messagerie
CID40511LAC4354	Jeudi	après-midi	Paramètres
CID58063LAC4354	Jeudi	après-midi	Jeux
CID58063LAC4354	Jeudi	soir	Paramètres
CID64457LAC4354	Jeudi	soir	Emma
CID22057LAC4354	Jeudi	soir	Messagerie

Des études sur la personnalisation des téléphones mobiles ont montré que les sonneries, les profils du téléphone, les images de fond et les thèmes graphiques sont personnalisés par plus de 85% des utilisateurs durant les premières semaines d'utilisation [11]. Par la suite, seul le changement de profil demeure une tâche répétitive. Les éléments de personnalisation ne rencontrent plus d'usage particulier mais continuent d'être changés ponctuellement, typiquement lorsque l'utilisateur télécharge une nouvelle sonnerie ou installe un nouveau thème. Ceci peut s'expliquer par le fait que les tâches de personnalisation restent fastidieuses. Dans EMMA, nous considérons ces éléments de personnalisation comme des opportunités pour renforcer la perception du contexte (partie gauche de la Figure 1 par rapport à

la partie droite de la Figure 3). En effet, pourquoi ne pas associer à chaque contexte clé une image de fond spécifique permettant de renforcer la perception du contexte par l'utilisateur final ? Ce constat motive le sondage des tâches de personnalisation.

L'apprentissage des changements de profil sur téléphone a fait l'objet d'autres travaux [3]. Nous collectons les changements de profil de la même façon que les précédents événements système. Chaque changement de profil donne lieu à une observation dans *l'historique de l'interaction*. Ces observations sont contextualisées (Tableau 4). Les tâches de personnalisation sont quant à elles collectées à chaque modification depuis l'application Paramètres. Les observations sont stockées de la même façon que les changements de profil ou d'application.

**Tableau 4. Changements de profil**

Localisation	Jour	Moment	Profil
CID22057LAC4354	Lundi	soir	Normal
CID40511LAC4354	Mardi	matinée	Silencieux
CID58063LAC4354	Mardi	après-midi	Normal
CID25550LAC4354	Mardi	soir	Normal
CID40506LAC4354	Mercredi	matinée	Silencieux
CID58063LAC4354	Mercredi	matinée	Normal
CID40511LAC4354	Mercredi	après-midi	Réunion
CID58063LAC4354	Mercredi	après-midi	Normal
CID40511LAC4354	Jeudi	matinée	Silencieux

### 3.2.3 Collecte des données de plasticité

Chaque fois qu'EMMA négocie un changement de contexte ou une adaptation avec l'utilisateur, les décisions de ce dernier sont collectées. De telles observations sont utiles pour améliorer les stratégies d'EMMA. Par exemple, elles peuvent motiver un passage du mode automatique au mode négocié. La proposition de changement sera alors négociée avec l'utilisateur qui reste libre de la rejeter si elle ne lui semble pas appropriée.

**Tableau 5. Décisions utilisateur sur changements de contexte**

Changement de contexte	Décision utilisateur
Maison	Oui
Travail	Oui
Maison	Non
Maison	Oui

**Tableau 6. Décisions utilisateur lors des adaptations**

Contexte	Cible de l'adaptation	Décision utilisateur
Maison	Profil	Oui
Maison	Image d'arrière-plan	Oui
Maison	Ecran d'accueil	Oui
Maison	Jeu de couleurs	Oui
Maison	Sonnerie	Non
Travail	Profil	Oui
Travail	Ecran d'accueil	Oui
Travail	Sonnerie	Non

Les Tableaux 5 et 6 donnent des exemples de décision collectées. Elles aident EMMA à améliorer à la fois l'identification du contexte (par exemple, si l'utilisateur rejette une proposition de changement de contexte) et l'adaptation (par exemple, si l'utilisateur rejette un changement de profil alors qu'il entre dans un contexte clé). D'autre part, elles peuvent aider le concepteur à déterminer quelles adaptations s'avèrent pertinentes en pratique.

## 3.3 Modèle Utilisateur

Cette sous-section décrit les fondements bayésiens à nos travaux puis développe les techniques d'apprentissage et de classification.

### 3.3.1 Modèle utilisateur bayésien

Les réseaux bayésiens sont des modèles graphiques probabilistes qui consistent en une partie qualitative et une partie quantitative. La partie qualitative constitue la structure du réseau : un graphe acyclique orienté où les nœuds correspondent aux variables et les arcs représentent les influences entre variables. La partie quantitative fournit les tables de probabilités conditionnelles qui constituent les paramètres du réseau. Les réseaux bayésiens sont des outils puissants qui tirent profit des algorithmes d'inférence et d'apprentissage associés. L'inférence s'appuie sur le théorème de Bayes pour propager la connaissance au sein du réseau. L'apprentissage s'applique autant à la structure du réseau qu'à ses paramètres et peut se faire à partir de données complètes ou incomplètes. Les réseaux bayésiens sont souvent utilisés pour le diagnostic, la prédiction, la modélisation et la surveillance. Les modèles utilisateur bayésiens ont fait l'objet de travaux antérieurs. [12] marque le point d'entrée de notre travail. Sur téléphone mobile, [3] explore un apprentissage bayésien pour identifier quand et comment l'utilisateur change de profil à travers le temps.

En pratique, les modèles bayésiens peuvent être construits soit à partir de la connaissance d'un expert soit automatiquement à partir de données, soit à partir des ces deux types d'information. En tant qu'experts, nous avons modélisé la structure du modèle utilisateur dans EMMA. Cette structure (Figure 8) dirige le format des données collectées (voir la sous-section précédente). Dans notre modèle utilisateur, chaque nœud assume une fonction particulière. Nous distinguons trois types de fonctions selon que le nœud est en charge d'observer et reconnaître le contexte d'usage (cas des nœuds *Day*, *Time*, *Location*, et *Context*), d'observer l'interaction de l'utilisateur pour calculer l'adaptation (cas des nœuds *Application*, *Profile*, *Background*, *Ringtone*, *Scheme*, et *ColorScheme*), ou de gérer les préférences de plasticité (cas des nœuds restants). Initialement, le modèle utilisateur ne contient aucune connaissance a priori.

D'un point de vue implémentatif, le modèle utilisateur bayésien est développé avec Netica™ [15]. Netica est une solution logicielle développée par Norsys Software Corp. Netica est un logiciel complet qui inclut un éditeur graphique et une API (Application Programming Interface). L'API est disponible sous plusieurs systèmes d'exploitation et utilisable dans plusieurs langages de programmation (C/C++/C#, Java, Visual Basic). Nous utilisons une version C de l'API spécialement développée pour s'exécuter sur des plates-formes équipées de Windows Mobile. Cette API se présente sous la forme d'une DLL (Dynamic-Link Library) compacte de fonctions C ultra rapides parmi lesquelles on trouve une procédure d'apprentissage de paramètres qui permet de réviser les tables de probabilités à partir de données.

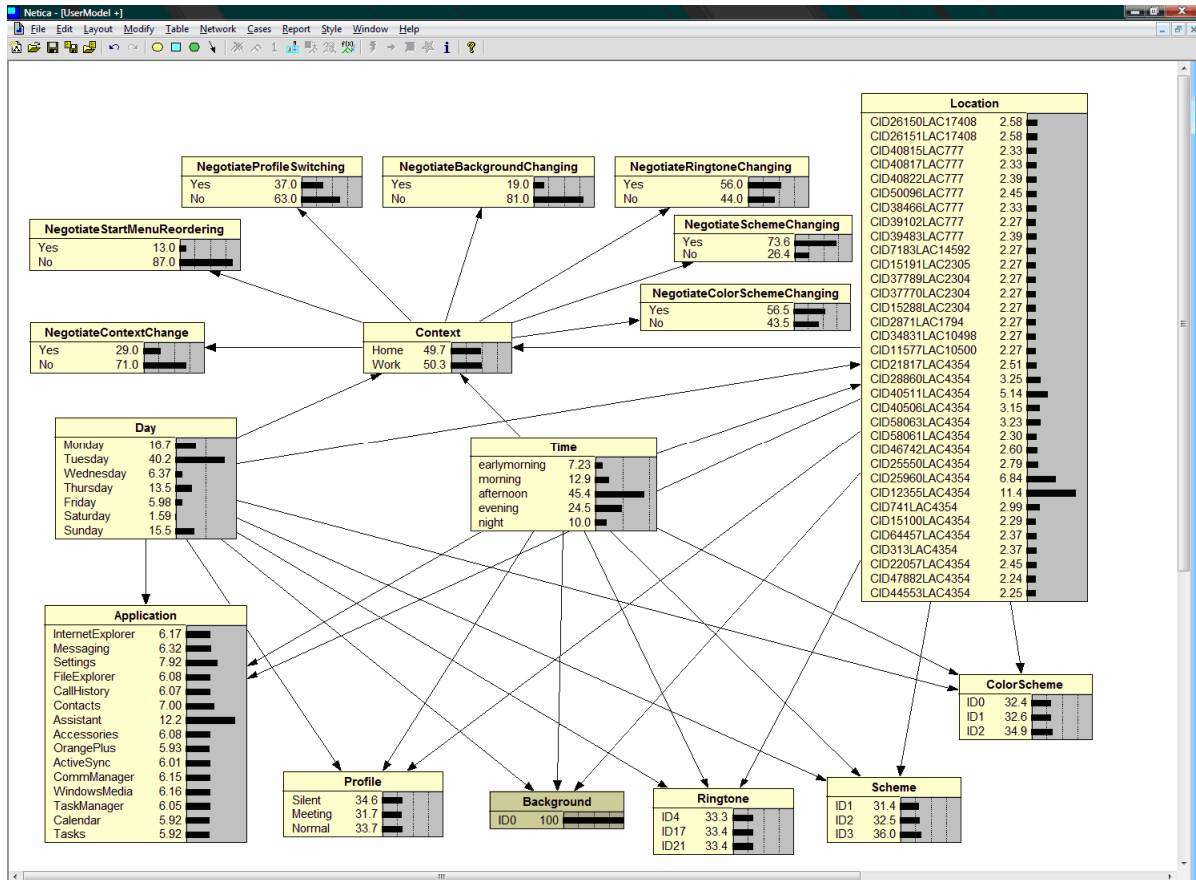


Figure 8. Le modèle utilisateur dans Netica une fois les paramètres appris sur la base de 9 jours de collecte.

### 3.3.2 Apprentissage de paramètres

Afin d'exploiter l'inférence bayésienne, il nous faut apprendre les états et les probabilités conditionnelles de chaque nœud du réseau. Pour cela, les données collectées précédemment alimentent l'algorithme d'apprentissage de paramètres mis à disposition par Netica. Une fois l'apprentissage de paramètres terminé, le modèle utilisateur est prêt à être utilisé pour inférer la connaissance (par exemple, le contexte d'usage courant, les adaptations pertinentes et la façon de la négocier). La Figure 8 montre une capture d'écran de Netica affichant le modèle utilisateur une fois les paramètres appris.

### 3.3.3 Classification hiérarchique

Chaque changement dans l'une des variables du contexte d'usage (localisation géographique, jour de la semaine ou moment de la journée) peut potentiellement conduire EMMA à proposer de changer de contexte et d'adapter l'IHM, ce qui pourrait entraîner un rejet massif par les utilisateurs. Aussi, afin de limiter l'adaptation, nous cherchons à identifier des contextes d'usage clés à l'aide d'une technique de classification hiérarchique. L'objectif est de rassembler au sein de ces contextes clés un ensemble de variables pour lesquelles les actions de l'utilisateur présentent une certaine similarité. L'algorithme commence par mettre chaque variable dans un cluster différent. Ensuite, à chaque pas, l'algorithme choisit la paire de clusters les plus proches et les rassemble au sein d'un

nouveau cluster. Les distances entre clusters peuvent être calculées à partir de différentes méthodes en se basant sur les probabilités conditionnelles inférées pour chaque variable considérée. La méthode hiérarchique produit tous les nombres possibles de clusters. Dans nos expérimentations, nous avons fixé le nombre de clusters à trois, comme le nombre de contextes clés retenus dans le projet Adamos [16]. La présence de l'utilisateur dans un contexte clé est négociée : si l'utilisateur accepte, la méta-IHM est affichée, permettant à l'utilisateur de personnaliser le nom et les caractéristiques de ce contexte (partie droite de la Figure 6).

## 4. CONCLUSION

En synthèse, l'article présente EMMA, un prototype opérationnel s'attaquant à deux verrous de l'adaptation au contexte : la découverte sur le terrain de contextes clés et l'identification par l'usage d'adaptations pertinentes. Pour ce faire, EMMA explore plusieurs domaines de recherche dont la programmation par l'exemple, les modèles utilisateur et la plasticité des IHM. EMMA embarque un modèle utilisateur fondé sur un réseau bayésien, support à l'apprentissage et à la classification sous l'éventuel contrôle de l'utilisateur final. EMMA est actuellement en phase de pré-test auprès de six utilisateurs pour une durée de deux semaines. Une campagne d'évaluation d'un mois sera menée en avril sur un panel de douze utilisateurs en conditions réelles d'utilisation.

## 5. REFERENCES

- [1] Abascal, J. and Civit, A. 2001. Universal Access to mobile Telephony as a Way to Enhance the Autonomy of Elderly People. In Proceedings of the 2001 EC/NSF Workshop on Universal Accessibility of Ubiquitous Computing: Providing For the Elderly (Alcácer do Sal, Portugal, May 22 – 25, 2001). WUAUC'01. ACM, New York, NY, 93-99. DOI= <http://doi.acm.org/10.1145/564526.564551>
- [2] Balme, L., Demeure, A., Barralon, N., Coutaz, J., and Calvary, G. 2004. CAMELEON-RT: a Software Architecture Reference Model for Distributed, Migratable, and Plastic User Interfaces. In Proceedings of the 2<sup>nd</sup> European Symposium on Ambient Intelligence (Eindhoven, The Netherlands, November 2004). EUSAI'04. Markopoulos and al. Eds. LNCS, vol. 3295. Springer-Verlag Heidelberg Publ., 291-302.
- [3] Bridle, R. and McCreath, E. 2005. Improving the Mobile Phone Habitat – Learning Changes in User's Profiles. In Proceedings of the 18<sup>th</sup> Australian Joint Conference on Artificial Intelligence (Sydney, Australia, December 5 – 9, 2005). AI 2005: Advances in Artificial Intelligence, Springer LNCS Publ., 970-974.
- [4] Bridle, R. and McCreath, E. 2006. Inducing Shortcuts on a Mobile Phone Interface. In Proceedings of the 2006 International Conference on Intelligent User Interfaces (Sydney, Australia, January 29 – February 1, 2006). IUI'06. ACM, New York, NY, 327-329. DOI= <http://doi.acm.org/10.1145/1111449.1111526>.
- [5] Coutaz, J. 2006. Meta-User Interfaces for Ambient Spaces. Invited talk, 6<sup>th</sup> workshop on Task Models Diagrams for UI Design (Hasselt, Belgium, October 23 – 24, 2006). TAMODIA'06. Springer LNCS Publ., 1-15.
- [6] Cypher, A. 1993. Eager: Programming Repetitive Tasks by Demonstration. In "Watch What I Do: Programming By Demonstration", A. Cypher, D. C. Halbert, D. Kurlander, H. Lieberman, D. Maulsby, B. A. Myers, and A. Turransky, Eds. MIT Press, Cambridge, MA, 205-217.
- [7] Demumieux, R. and Losquin, P. 2005. Collecter les usages réels des clients de téléphonie mobile (un outil embarqué). Actes de la 17<sup>ème</sup> Conférence Francophone sur l'Interaction Homme-Machine (Toulouse, France, September 27 – 30, 2005). IHM'05, vol. 264. ACM, New York, NY, 293-298. DOI= <http://doi.acm.org/10.1145/1148550.1148599>
- [8] Dey, A. K. 2000. Providing Architectural Support for Building Context-Aware Applications. Ph. D. thesis, December 2000, Georgia Institute of Technology.
- [9] Dey, A. K., Hamid, R., Beckmann, C., Li, I., and Hsu, D. 2004. a CAPpella: Programming by Demonstration of Context-Aware Applications. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (Vienna, Austria, April 24 – 29, 2004). CHI '04. ACM, New York, NY, 33-40. DOI= <http://doi.acm.org/10.1145/985692.985697>
- [10] Fischer, G. 2001. User Modeling in Human-Computer Interaction. In User Modeling and User-Adapted Interaction Journal, 11 (1-2) (March 2001). 65-86.
- [11] Häkkinä, J. and Chatfield, C. 2006. Personal Customisation of Mobile Phones – A Case Study. In Proceedings of the 4<sup>th</sup> Nordic Conference on Human-Computer Interaction: Changing Roles (Oslo, Norway, October 14 – 18, 2006). A. Mørch, K. Morgan, T. Bratteteig, G. Ghosh, and D. Svanaes, Eds. NordiCHI'06, vol. 189. ACM, New York, NY, 409-412. DOI= <http://doi.acm.org/10.1145/1182475.1182524>
- [12] Horvitz, E., Breese, J., Heckerman, D., Hovel, D., and Rommelse, K. 1998. The Lumière Project: Bayesian User Modeling for Inferring the Goals and Needs of Software Users. In Proceedings of the 14<sup>th</sup> Annual Conference on Uncertainty in Artificial Intelligence (Madison, WI, July 1998). UAI'98. Morgan Kaufmann, San Francisco, CA, 256-265.
- [13] Laasonen, K. 2004. Where Are You Going? Predicting user movement from cellular data. In Proceedings of the Proactive Computing Workshop (Helsinki, Finland, November 25 – 26, 2004). PROW'04. 121-124.
- [14] Mitchell, T. M., Caruana, R., Freitag, D., McDermott, J., and Zabowski, D. 1994. Experience With a Learning Personal Assistant. Communications of the ACM 37, 7 (July 1994), 80-91. DOI= <http://doi.acm.org/10.1145/176789.176798>
- [15] Netica, <http://www.norsys.com/>
- [16] Rantakokko, T. and Arhipainen, L. 2004. Adamos Menu: Towards Adaptive Service Selection. In Proceedings of the Proactive Computing Workshop (Helsinki, Finland, November 25 – 26, 2004). PROW'04. 9-13.
- [17] Ruvini, J. and Dony, C. 2000. APE: Learning User's Habits to Automate Repetitive Tasks. In Proceedings of the 5<sup>th</sup> International Conference on Intelligent User Interfaces (New Orleans, LA, January 09 – 12, 2000). IUI '00. ACM, New York, NY, 229-232. DOI= <http://doi.acm.org/10.1145/325737.325854>
- [18] Schilit, B.N., Adams, N., and Want, R. 1994. Context-Aware Computing Applications. In Proceedings of the Workshop on Mobile Computing Systems and Applications (Santa Cruz, CA, December 1994). WMCSA'94. IEEE Computer Society Press, 85-90.
- [19] Siewiorek, D., Smailagic, A., Furukawa, J., Moraveji, N., Rieger, K., and Shaffer, J. 2003. SenSay: A Context-Aware Mobile Phone. In Proceedings of the 7<sup>th</sup> IEEE International Symposium on Wearable Computers. 248-249.
- [20] St. Amant, R., Horton, T. E., and Ritter, F. E. 2004. Model-based Evaluation of Cell Phone Menu Interaction. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (Vienna, Austria, April 24 – 29, 2004). CHI '04. ACM, New York, NY, 343-350. DOI= <http://doi.acm.org/10.1145/985692.985736>
- [21] Thevenin, D. and Coutaz, J. 1999. Plasticity of User Interfaces: Framework and Research Agenda. In Proceedings of the 7<sup>th</sup> IFIP International Conference on Human-Computer Interaction (Edinburgh, Scotland, August 30 – September 3, 1999). INTERACT'99. A. Sasse and C. Johnson Eds. IOS Press Publ., Amsterdam, the Netherlands, 110-117.