

Secondes Rencontres Jeunes Chercheurs en Interaction Homme-Machine



RJC-IHM'04
20-22 Octobre 2004
Lacanau



**Secondes Rencontres Jeunes Chercheurs
en Interaction Homme-Machine
RJC-IHM'04
20-22 octobre 2004**

Lacanau

<http://www.irit.fr/rjcihm04>



Avant-propos

Les RJC-IHM'04 se veulent une version inédite des Rencontres Jeunes Chercheurs (RJC) en Interaction Homme-Machine (IHM). Placées sous le thème de l'apprentissage, elles visent à induire une dynamique de groupe autour de la production et de la publication en IHM. Il s'agit d'apprendre aux jeunes chercheurs

- A mieux comprendre leur domaine, acquérir un recul, mûrir une vision de ce que sera l'IHM. Cette maturation est nécessaire pour un meilleur positionnement et une orientation réfléchie des travaux. Joëlle Coutaz, professeur à l'Université Joseph-Fourier de Grenoble, anime cet apprentissage à travers une rétrospective et perspectives de l'IHM ;
- A découvrir les acteurs en termes de laboratoires, thématiques développées et individus. Cette cartographie est précieuse pour identifier les complémentarités thématiques et établir les collaborations appropriées. Chaque jeune chercheur présente sa structure de recherche et établit les ponts avec d'autres laboratoires français ou internationaux ;
- A présenter sa problématique de recherche à l'écrit et à l'oral. L'écrit est travaillé en anglais sous la direction de Wendy Mackay, Directeur de Recherche à l'INRIA Futurs de Paris. Trois demi-journées y sont consacrées sur la base d'une publication d'une page apportée par chaque jeune chercheur. La publication française a fait l'objet d'une évaluation par le comité de lecture. L'oral est travaillé lors des présentations des jeunes chercheurs.

Au-delà de ces avancées individuelles, nous espérons avoir bâti un socle commun de connaissances et instigué une dynamique de groupe. Nous attendons une émulsion de personnes passionnées qui, dans un avenir proche, prendra corps par rapprochements géographiques, adhésions à des groupes de travail communs ou organisation de futures journées. C'est cette consolidation de la communauté IHM que nous souhaitons promouvoir, dans l'esprit de l'AFIHM.

Gaëlle Calvary et Emmanuel Dubois

Remerciements

Les RJC-IHM'04 ont pris corps grâce à un ensemble d'acteurs que nous remercions très chaleureusement....

En tout premier lieu nous remercions l'AFIHM sans qui elles n'auraient pas vu le jour. Elles sont une manifestation directe de la politique de l'association dont la priorité est de soutenir les actions en direction des jeunes chercheurs.

Les RJC-IHM'04 doivent leur succès scientifique à Joëlle Coutaz, professeur à l'Université Joseph-Fourier de Grenoble, et Wendy Mackay, Directeur de Recherche à l'INRIA Futurs de Paris qui ont consacré de leur temps précieux et de leurs compétences à la formation des jeunes chercheurs. Nous les remercions très chaleureusement, au titre du comité d'organisation et des jeunes chercheurs, d'avoir cru en ces journées. Nous remercions le comité de lecture pour l'évaluation des articles, préalable à la sélection des jeunes chercheurs.

D'un point de vue technique, nous remercions très vivement nos soutiens financiers qui, grâce à leur contribution, ont donné corps à la politique de l'AFIHM, rendant accessibles ces journées à une majorité de jeunes chercheurs : l'AFIHM, le CNRS via le RTP16, le GDR I3, l'IRIT et La Poste.

Enfin, nous pensons particulièrement à Louis Duroch, l'auteur du logo des RJC-IHM'04. Il a mis avec gentillesse et patience toutes ses compétences de graphiste à notre disposition : nous le remercions très chaleureusement.

Pour finir, nous nous tournons bien évidemment vers les jeunes chercheurs sans qui ces journées n'auraient pas non plus vu le jour. Nous vous souhaitons un bel envol !

Comités et Parrainages

Comité de lecture

Anastassova Margarita
Antoine Jean-Yves
Conversy Stéphane
D'Ausbourg Bruno
Delozanne Elisabeth
Derycke Alain
Dragicevic Pierre

Dubois Jean-Marc
Flury Thibaud
Girard Patrick
Haradji Yvon
Lenne Dominique
Martin Jean-Claude
Massou Luc

Pecci Isabelle
Renevier Philippe
Sire Stéphane
Skaf Ahmad
Truillet Philippe
Zouinar Moustafa

Comité d'organisation

Calvary Gaëlle
Dubois Emmanuel
Liberati Véronique

Parrainages scientifique et financier



RTP16 - IHM



Table des matières

Présentations longues

Un Dispositif de Visualisation 3D Immersif pour les Documents Numériques.....	3
<i>Rodrigo Almeida, Jérôme Dupire, CNAM – Paris</i>	
CIS : Décrire les Techniques d'Interaction et les Evaluer en Contexte	9
<i>Caroline Appert, LRI et INRIA Futurs – Orsay</i>	
Infrastructure Logicielle pour Interfaces Homme-Machine Plastiques.....	15
<i>Lionel Balme, CLIPS – Grenoble</i>	
Meta UI : vers un Desktop++	21
<i>Nicolas Barralon, CLIPS – Grenoble</i>	
Erreur Humaine, Modèles de Tâches et Description Formelle pour la Conception et l'Evaluation des Systèmes Critiques et Tolérant aux Erreurs	27
<i>Sandra Basnyat, IRIT – Toulouse</i>	
Pointage Sémantique et Distracteurs, la Dynamique du Pointage à la Rescousse	33
<i>Renaud Blanch, LRI et INRIA Futurs – Orsay</i>	
Interfaces Stylo pour la Saisie d'Ecriture Manuscrite sur Systèmes Mobiles de Petite Taille	39
<i>François Bouteruche, IRISA – Rennes</i>	
Les Interactions Multimodales et Multicanal dans UMR.....	45
<i>Vincent Chevrin, TRIGONE – Villeneuve d'Ascq</i>	
Le Problème de la Plasticité dans la Conception d'IHM	51
<i>Alexandre Demeure, CLIPS – Grenoble</i>	
AMUSING : Outil d'Assemblage et d'Adaptation d'IHM	57
<i>Anne-Marie Dery-Pinna, Jérémy Fierstone, Open-Plug, Rainbow et I3S – Sophia Antipolis</i>	
Modélisation et Vérification Formelle des IHM multimodales.....	63
<i>Nadjet Kamel, LRIA/USTHB – Alger et LISI/ENSMA – Futuroscope</i>	
La Conception Rationalisée pour les Systèmes Interactifs.....	69
<i>Xavier Lacaze, IRIT – Toulouse</i>	
Apports de l'Ergonomie à la Conception de Dispositifs Techniques de Lecture Numérique	75
<i>Angélica Léal, PARAGRAPHÉ – Saint Denis</i>	
Méthode de Conception pour les Systèmes de Réalité Augmentée	81
<i>Benoît Mansoux, CLIPS et TIMC – Grenoble</i>	

Validité d'une Evaluation In Situ via la Méthode du Magicien d'Oz	87
<i>Bruno Merlin, IRIT – Toulouse</i>	
KeyGlasses : des Touches Semi-transparentes pour Optimiser la Saisie de Texte	93
<i>Matthieu Raynal, IRIT – Toulouse</i>	
Introduire des Techniques de Programmation sur Exemple dans une boîte à outils : une étude de besoin.....	99
<i>Loé Sanou, Patrick Girard, Laurent Guittet, LISI-ENSMA - Futuroscope</i>	
Etude de l'Optimisation de Claviers Virtuels au Travers des Sujets Handicapés versus Valides.....	105
<i>Frédéric Vella, Nadine Vigouroux, IRIT – Toulouse</i>	

Présentations Courtes

Réalité Mixte et Co-Adaptation.....	113
<i>Pascal Costa-Cunha, LRI - Orsay</i>	
Modélisation de Tâches Contextualisées pour la Construction d'Applications Collaboratives.....	119
<i>Olivier Delotte, Bertrand David, ICTT – Lyon</i>	
Modélisation d'un Support de Communication pour des Personnes Agées à Domicile.....	125
<i>Abir Ghorayeb, Vincent Rialle, Norbert Nourry, Joelle Coutaz, TIMC et CLIPS – Grenoble</i>	
Variations pour une Nouvelle Méthode de Saisie de Données : Glyph	131
<i>Gurvan Uguen, Franck Poirier, VALORIA – Vannes</i>	
Architectures Logicielles Appliquées à l'Interaction Personne-Système	137
<i>Jérôme Lard, Célestin Sedogbo, THALES – Orsay</i>	
Le Projet SAI : Scénarios d'Animations Interactives	143
<i>Nicolas Pagès, Monique Noirhomme, FUNDP – Namur</i>	
Présentation de l'Harmonie Musicale par une Notation Spatiale et une Souris à Retour de Force	149
<i>Bertrand Tornil, Nadine Baptiste-Jessel, IRIT – Toulouse</i>	
Conduite de Projet Informatisé dans les PME/PMI : les Apports d'une Démarche Ergonomique	155
<i>Virginie Valiani, Gérard Vallery, ECCHAT – Amiens</i>	



Présentations longues

Un dispositif de visualisation 3D immersif pour les documents numériques

Rodrigo Almeida

CNAM / CEDRIC
292 rue St-Martin
F-75003 Paris
andrad_r@auditeur.cnam.fr

Jérôme Dupire

CNAM / CEDRIC
292 rue St-Martin
F-75003 Paris
dupire_j@cnam.fr

RESUME

L'omniprésence de la 3D dans les applications récentes illustre la tendance actuelle des développements informatiques. Mais si les logiciels ont migré du plan vers l'espace, les dispositifs de visualisation n'ont pas été transformés pour accueillir au mieux cette nouvelle dimension. Nous présentons dans cet article un dispositif de visualisation destiné aux environnements en 3D. Celui-ci se compose d'un écran hémisphérique immersif pour les tâches générales et d'un écran standard pour les travaux plus fins. L'intérêt de ce système réside dans l'optimisation des activités de recherche, d'organisation et de consultation de documents numériques dans un contexte 3D. Bien que notre prototype ait été développé dans le cadre des bibliothèques numériques, il peut facilement être étendu à n'importe quel environnement de travail informationnel.

MOTS CLES : Bibliothèques numériques, visualisation, multi-écran, vision périphérique, immersion, interaction 3D.

ABSTRACT

The omnipresence of 3D graphics in recent software shows the present trend in computer development. Even if computer software has migrated from the plane to the space, visualization hardware has not been transformed to better integrate this new dimension. In this paper, we present a visualization system intended for 3D environments. It is composed by an immersive hemispherical screen for general tasks and a standard monitor for more precise interaction. The goal of this system is to enhance research, organization, and consulting activities of digital documents within a 3D context. Although our prototype has been developed in a digital library research framework, it can be easily

extended to other kinds of information tasks.

CATEGORIES AND SUBJECT DESCRIPTORS: H.5.2 [User Interfaces]: Graphical User Interface, Interaction Styles, Ergonomics;

GENERAL TERMS: Design, Human Factor.

KEYWORDS : Digital libraries, visualization, multiple monitors, peripheral vision, immersion, 3d interaction.

INTRODUCTION

Depuis le « manifeste » d'Adrian van Dam [13] incitant les chercheurs et les industries à investir dans le développement des interfaces utilisateurs en 3D, des progrès considérables ont été réalisés dans ce domaine. L'explosion technologique des cartes graphiques, la baisse des coûts du matériel, la multiplication des langages de programmation et de script dédiés à la 3D ont banalisé l'accès à ce type de représentation. Que ce soit au travers des jeux vidéos ou des « bureaux 3D » [12], la tendance actuelle est résolument orientée vers l'abandon progressif des représentations en 2D. Pourtant les changements dans le domaine des dispositifs de visualisation ont été très réduits. La presque totalité des utilisateurs (professionnels, chercheurs ou particuliers) est toujours équipée aujourd'hui d'écrans traditionnels de 15 ou 17 pouces. La souris, l'écran, la tablette graphique sont autant de périphériques initialement destinés à explorer les environnements en 2D. Or, pour pouvoir tirer profit et développer de nouveaux paradigmes pour la 3D, il semble important de posséder des outils adaptés.

Nous présenterons tout d'abord le dispositif qui est au centre de nos expérimentations – la *VisionStation*. Nous évoquerons ensuite les points positifs et les obstacles rencontrés lors de son utilisation. Nous décrivons l'intégration de la *VisionStation* avec un second écran, ainsi que les interactions spatiales possibles pour un environnement de consultation, dans le contexte particulier des bibliothèques numériques. Nous proposerons enfin quelques orientations pour les travaux à venir.



Figure 1 : La disposition des deux écrans et des deux souris (à gauche la souris 3D).

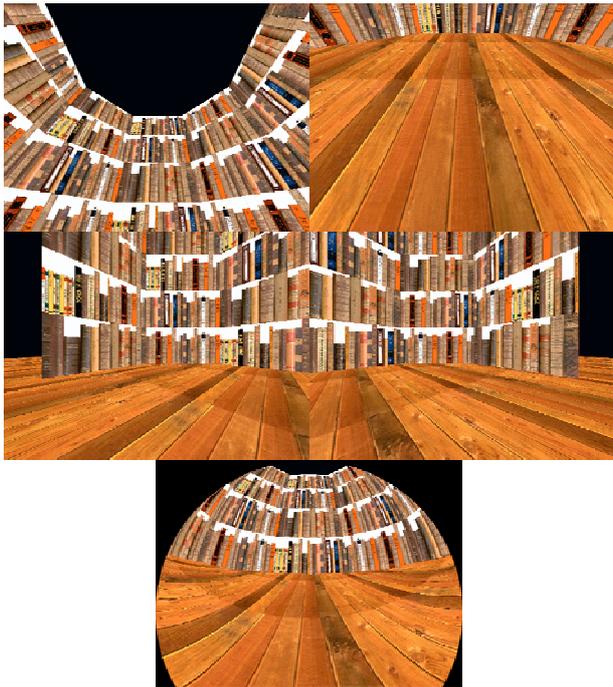


Figure 2 : Les 4 rendus initiaux et (en bas) l'image résultante de la fusion des précédentes, après déformation.

LA VISIONSTATION

Procurer à un utilisateur une meilleure immersion dans l'espace virtuel 3D nécessite de modifier les conditions de restitution visuelle de celui-ci. Patrick et al. [10] ont montré que la visualisation d'une scène qui occupe tout le champ de vision de l'utilisateur peut lui apporter une plus grande appropriation cognitive de l'espace virtuel. Dans cette perspective, seuls de rares périphériques, comme le casque de réalité virtuelle et les écrans larges, remplissaient les conditions préconisées. Notre choix s'est porté vers le système *VisionStation* (VS1024) produit par la société *Elumens* (figure 1). Ce dispositif est composé d'un écran semi-sphérique d'environ 1,5 mètre de diamètre, d'une tablette centrale sous laquelle est logé un projecteur EPSON (résolution maximale de 1024x768 pixels). Ce dernier est équipé d'une lentille spéciale de type *fish-eye* permettant de

projeter l'image sur la surface complète de l'écran [5]. Livrée avec le dispositif matériel, l'API SPI (*Spherical Projection for Images*) est destinée à traiter les images générées par l'application en vue de leur projection. Cette API est compatible avec *OpenGL* et *Direct3D*. Sans cette étape de transformation, l'image initialement destinée à un écran plat apparaîtrait déformée sur l'écran semi-sphérique. Le processus de traitement débute par le calcul du rendu de 4 points de vue de la scène 3D initiale (latéraux, supérieur et inférieur). L'étape suivante consiste à transformer chaque rendu en une texture, qui sera appliquée ensuite sur un maillage, préalablement déformé pour compenser la géométrie particulière de l'écran (figure 2). Enfin, cette image est projetée sur l'écran de la *VisionStation*, couvrant la totalité des 180° grâce à la lentille *fish-eye* du vidéoprojecteur.

IMMERSION

Dans la *VisionStation*, l'image occupe donc la totalité du champ de vision de l'utilisateur. Il est donc protégé du bruit visuel habituellement rencontré lors de l'utilisation d'écrans plats. Ceci permet d'accroître à la fois le confort d'utilisation mais aussi de faciliter la concentration sur les tâches en cours. En outre, l'augmentation de son espace virtuel de travail lui permet de rendre visible simultanément plusieurs documents. Il peut donc utiliser les régions de son champ de vision comme des éléments sémantiques pour un travail optimisé sur une information particulière [7, 9]. Cette propriété de spatialisation dans la visualisation offerte par le dôme donne à l'utilisateur une expérience plus réaliste dans la manipulation des documents. Un tel environnement procure un affichage fortement immersif, qui supprime chez l'utilisateur la sensation d'être un élément extérieur. Ainsi « plongé » dans cet espace, il peut mieux appréhender les différents objets, en transférant ses habitudes cognitivo-comportementales concernant leur manipulation, leur organisation et leur exploitation dans l'espace. Ce phénomène est d'autant plus important lorsqu'il s'agit d'utilisateurs novices qui ne sont pas habitués aux représentations de scènes en 3D.

OBSTACLES

Le principal inconvénient rencontré dans l'utilisation de la *VisionStation* comme outil de consultation de documents numériques est la résolution de l'image projetée. Elle est au maximum de 1024 x 768 pixels (limitation du vidéoprojecteur). Cette résolution reste très satisfaisante pour des applications comme les simulateurs de vol, les jeux vidéo et de manière plus générale, pour l'affichage d'images animées. Par contre, lors d'un travail plus précis sur des documents, textuels ou graphiques, la qualité de l'image perçue devient un critère essentiel pour le confort et la bonne compréhension des informations par l'utilisateur. Mutter et Maurutto montrent que la vitesse de lecture sur un écran

d'ordinateur est proportionnelle à la résolution de l'image affichée [8]. C'est dans ce contexte particulier que la *VisionStation* montre ses limites. On peut en effet discerner à l'affichage le détail des pixels projetés (figure 3). Ce phénomène constitue un obstacle important à l'utilisation de ce dispositif, dans cette configuration, pour ce type d'application.

Baudisch et al. [1] ont proposé un dispositif composé de matériel courant (un vidéoprojecteur, un écran de projection et un moniteur) permettant de projeter simultanément une vue globale de la scène et une partie de celle-ci à une résolution plus élevée. Ce système a été testé pour la visualisation et le travail informationnel en 2D tel que l'analyse de cartes ou d'images satellitaires. Ce dispositif ne semble pas approprié pour gérer simultanément des contextes différents, la vue détaillée étant inévitablement une partie de l'image globale. De plus, le problème du bruit visuel lié aux écrans traditionnels se retrouve encore un peu dans ce système.

Fort de ces constats, nous avons ajouté à notre dispositif un écran plat (TFT 17 pouces). Celui-ci est capable

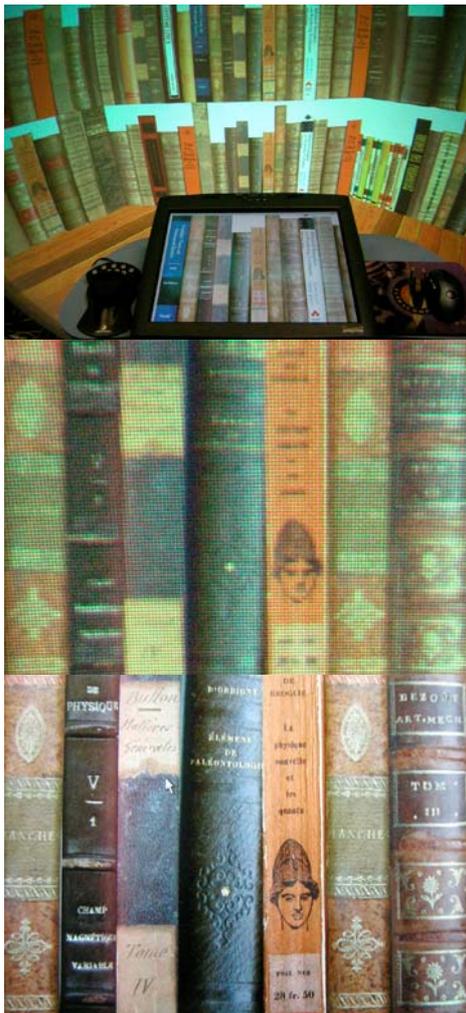


Figure 3: En haut, la vue globale de la scène sur la *VisionStation*. Au milieu, le détail de livres au centre de la projection. En bas, la même partie de l'image affichée sur l'écran plat. Noter les différences dans la qualité des images affichées.

d'afficher des résolutions plus fines que la *VisionStation* (jusqu'à 1280x1024). Nous avons donc envisagé l'utilisation de ce moniteur pour différentes tâches. Il pourrait permettre d'afficher une partie de la scène 3D, qui nécessiterait des conditions compatibles avec un travail sur les documents (confort visuel), ou d'isoler une partie de la scène, permettant une vision précise et plus détaillée de celle-ci. Nous pourrions, d'autre part, permettre à l'utilisateur d'accéder, par cet affichage complémentaire, à des éléments d'interaction non disponibles dans la vue globale.

FENETRES ET ECRANS

Pour les fonctions de fenêtrage et la gestion des interactions de notre prototype, nous avons utilisé la bibliothèque GLUT (OpenGL Utility Toolkit) [6]. Pour travailler dans un modèle bi-écran, nous avons créé deux contextes et associé chacun à une fenêtre. Pour chaque fenêtre, nous avons spécifié des fonctions de projection différentes. Malgré le fait que les deux fenêtres utilisent la même description de la scène, elles ont chacune leur champ de vision spécifique. La fenêtre destinée à la *VisionStation* utilise les fonctions d'affichage et de projection issues de l'API SPI alors que la fenêtre spécifiée pour l'écran normal utilise les fonctions de projection d'OpenGL et de la bibliothèque GLU (*OpenGL Utility Library*), pour un champ de vision de 30 degrés. Enfin, pour permettre l'affichage en bi-écran, nous avons utilisé une carte graphique *NVIDIA Quadro FX3000*, montée sur un PC (P4, 2,4 GHz, 512 RAM). Nous avons choisi de placer le deuxième écran sur la tablette de la *VisionStation*. L'avantage d'une telle configuration est de conserver à la fois l'immersion de l'utilisateur dans la scène 3D et la possibilité de travailler sur une image en haute résolution. Cela permet enfin d'associer ou de différencier le travail de l'utilisateur sur chaque visualisation.

IMPLEMENTATION-EXPERIMENTATION

Dans des travaux précédents, notre équipe a étudié de nouveaux modèles d'interaction en 3D, en particulier dans le domaine des bibliothèques numériques [3, 4]. Nous nous sommes servis de ces travaux comme support pour tester l'organisation des différentes fonctionnalités liées aux deux affichages. Le modèle d'organisation de la scène est cylindrique et correspond à celui décrit en [11]. L'affichage de cet espace se fait sur la *VisionStation*, et permet de visualiser une collection de livres. Ceux-ci sont identifiables par leur tranche, et organisés en « étagère ». L'affichage sur l'écran plat permet, après sélection d'un ouvrage, de le consulter avec un confort visuel adéquat (résolution élevée). Afin d'optimiser la navigation dans l'espace virtuel, nous avons décidé d'utiliser une « souris 3D » (*SpaceMouse Plus* de *3dConnexion* - figure 2). GLUT ne reconnaissant pas ce périphérique, nous avons utilisé une version modifiée de cette bibliothèque [2] afin de pouvoir intégrer ce périphérique. La *SpaceMouse*

permet d'accéder facilement aux mouvements de translation et de rotation selon et autour des 3 axes. Pour notre prototype, nous avons choisi de contraindre les mouvements afin de faciliter les « déplacements ». L'utilisateur a donc à sa disposition les translations selon les 3 axes et la rotation autour de l'axe vertical. Cette dernière offre la possibilité pour l'utilisateur de percevoir très rapidement tout l'environnement qui l'entoure. De telles contraintes sont nécessaires afin d'éviter que l'utilisateur se "perde" dans cet espace. Enfin, des techniques de gestion de collisions ont été utilisées pour restreindre son mouvement au volume interne du cylindre dans lequel se trouve notre bibliothèque.

CONCLUSION ET TRAVAUX FUTURS

Cette première étude sur l'utilisation de nouveaux périphériques d'affichage nous offre de nouvelles opportunités en termes de visualisation et d'interaction. Parfaitement adaptée au rendu des scènes 3D, ce dispositif doit cependant être complété sur différents aspects. Tout d'abord, nous souhaiterions optimiser la relation de l'utilisateur vis à vis des deux écrans. Quelles pourrait être la répartition la plus efficace des tâches associées aux différents modes d'affichage ? D'autre part, nous pensons remplacer l'écran plat par un écran tactile. Nous pourrions alors implémenter, grâce à ce dispositif, de nouvelles interactions (feuilletter d'un livre par exemple). Il serait alors possible de remplacer la souris traditionnelle par un stylet ou par la main de l'utilisateur. Ce nouveau dispositif renforcera le caractère intuitif des interactions puisqu'elles seront basées sur un vocabulaire gestuel soit déjà connu de l'utilisateur, soit facilement assimilable [9]. Enfin, afin de tirer pleinement profit de l'immersion procurée par la *VisionStation*, nous devons proposer de nouvelles fonctionnalités dans l'aspect organisationnel du travail dans la scène 3D. L'immersion procurée par ce dispositif permettrait à l'utilisateur de s'approprier la topologie de la scène 3D et de transférer naturellement ses connaissances et habitudes organisationnelles du monde physique vers cet environnement virtuel. La recherche d'information, la visualisation de grandes quantités de données, l'organisation de celles-ci relativement aux besoins et demandes de l'utilisateur sont des aspects devant être donc étudiés.

BIBLIOGRAPHIE

- [1] Baudisch, P., Good, N., and Stewart, P.
Focus Plus Context Screens: Combining Display Technology with Visualization Techniques.
In *Proceedings of UIST '01*, Orlando, FL, Novembre 2001, pp.31-40.
- [2] Chung, A. J.
<http://www.doc.ic.ac.uk/~ajchung>
- [3] Cubaud P., P. Stokowski and A. Topol
Binding Browsing and Reading Activities in a 3D Digital Library.
In *Proceedings Joint Conference on Digital Libraries 2002*, Portland, Oregon, USA, July 14-18, 2002.
- [4] Cubaud, P., C. Thiria and A. Topol
Experimenting a 3D Interface for the Access to a Digital Library.
In *Proceedings ACM DL*, June 23-26, Pittsburgh, PA, USA, 1998.
- [5] Elumens
The SPIClops API.
Août 2001.
- [6] GLUT - OpenGL Utility Toolkit
<http://www.opengl.org/resources/libraries/glut.html>
- [7] Mackinlay, J. D., Heer, J. and Royer, C.**Wideband Visual Interfaces: Sensemaking on Multiple Monitors.**
In *Parc Technical Report*, 2003.
- [8] Muter, P., Maurutto, P.
Reading and Skimming from computer screens: the paperless office revisited.
In *Behavior and Information Technology*, 10(4), 257-266, 1991.
- [9] O'Hara K., Sellen A.
A comparison of reading paper and on-line documents.
In *Proceedings CHI 1997*, 1997, pp. 335-342.
- [10] Patrick E., Cosgrove D. et al.
Using a Large Projection Screen as an Alternative to Head-Mounted Displays for Virtual Environments.
In *Proceedings of the SIGCHI 2000*, Avril 2000, pp. 478-485.
- [11] Topol, A.
Interfaces 3D pour les Bibliothèques Numériques.
In *Mémoire de D.E.A. Médias et Multimédia du C.N.A.M.*, Septembre 1998.
- [12] Topol, A.
Immersion of XWindow applications into a 3D workbench.
In *Proceedings ACM CHI'2000*, The Hague, Netherlands, Avril, 2000.
- [13] Van Dam, A.
Escaping flatland in user interface design.
In *Proceedings of the 1992 symposium on Interactive 3D graphics*, Cambridge, MA, Etats-Unis, 1992.

Un dispositif de visualisation 3D immersif pour les documents numériques

Rodrigo Almeida Jérôme Dupire
 andrad_r@auditeur.cnam.fr dupire_j@cnam.fr
 CNAM/CEDRIC
 292 rue St-Martin
 F-75003 Paris

Motivations

"A Comparison of Reading Paper and Online Documents" O'Hara et Sellen
 Les périphériques de visualisation et d'interaction traditionnels ne sont pas adaptés au travail dans un environnement 3D
 Utilisation d'un plus grand champ de vision:
 . plus grande immersion = concentration
 . plus grand espace de travail pour organiser l'information

Expérimentation

Expérimenter un environnement de consultation de documents numériques visualisé avec une VisionStation d'Elumens

Le déplacement dans l'espace sera fait à l'aide d'une souris 3D (SpaceMouse de 3D Connexion)

Obstacles

Faible résolution des images affichées sur l'écran
 Perte de lisibilité et confort

API pour la Projection Sphérique

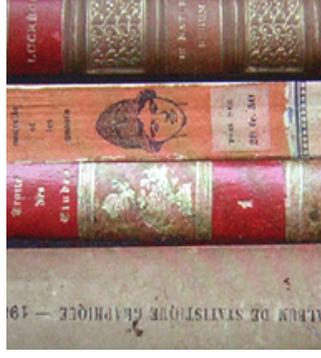
Définitions de la Scène 3D en OpenGL

Configuration des Canaux `spiInitialize`

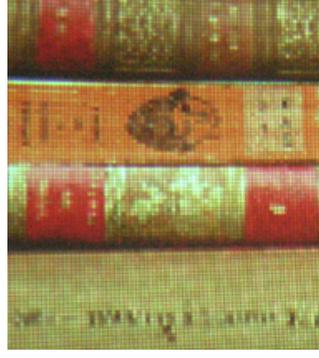
4 Canaux =
 4 rendus des points de vue latéraux, supérieur et inférieur
 Chacun avec une vue de 90°

Les 4 images générées sont stockées comme textures

Les 4 textures sont placées sur un maillage sphérique. `spiFlush` génère l'image finale



sur l'écran traditionnel



sur la VisionStation

Solution

"Focus Plus Context Screens" Baudisch et al.
 Deuxième Ecran:
 . Visualisation plus précise
 . Ajout de fonctions d'interaction

Conclusion et Travaux Futurs

Repartir les tâches de visualisation et d'interaction associées à chaque écran

Remplacer le deuxième écran par un écran tactile:
 procurer une interaction plus fine comme le feuilletage de documents

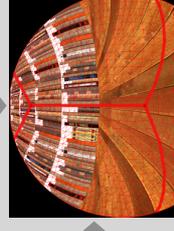
Faciliter l'appropriation de la topologie de la scène 3D par l'utilisateur pour qu'il puisse optimiser ses tâches

Implémentation

Scène: OpenGL
 Fonctions de Fenêtrage: GLUT
 Projection Sphérique: API SPIClops
 Interaction avec le SpaceMouse par une GLUT modifiée: `glutSpacemouseMotionFunc`
 Utilisation d'images de tranches de livres pour les représenter
 Distribution cylindrique des documents pour "entourer" l'utilisateur



Redessine la fenêtre: `glutDisplayFunc`
 Lors de la projection, l'image finale, avec une vue de 180°, est affichée sur la surface courbe



CIS : Décrire les techniques d'interaction et les évaluer en contexte

Caroline Appert

LRI & INRIA Futurs
Bâtiment 490
91405, Orsay, France
appert@lri.fr

RESUME

Cet article présente le modèle Complexity of Interaction Sequence (CIS). CIS décrit les techniques d'interaction et opérationnalise des contextes d'utilisation pour prédire l'efficacité d'une technique en contexte. CIS a été validé par une expérimentation contrôlée comparant les palettes, les palettes bimanuelles et les toolglasses.

MOTS CLES : Technique d'Interaction, Séquence d'Interaction, Complexité, Contexte, Palette, Palette Bimanuelle, Toolglass, Expérimentation.

ABSTRACT

This paper introduces the (CIS) model. CIS describes interaction techniques and operationalizes contexts of use in order to predict the technique efficiency in context. CIS has been validated by a controlled experiment comparing palettes, bimanual palettes and toolglasses.

KEYWORDS : Interaction Technique, Interaction Sequence, Complexity, Context, Palette, Bimanual Palette, Toolglass, Experimentation.

INTRODUCTION

Afin de rendre les applications graphiques plus utilisables, la recherche en IHM a créé de nombreuses techniques d'interaction. Cependant, très peu de logiciels les utilise. Une des raisons est la difficulté de choisir la technique plus performante parmi un ensemble de techniques. L'évaluation d'une technique d'interaction est souvent faite en comparant sa performance à celle d'autres techniques sur une tâche choisie dans le cadre d'une expérimentation contrôlée. Cependant, les résultats sont difficilement généralisables car ils dépendent de la tâche qui ne représente pas un contexte d'utilisation réel.

Nous présentons ici un nouveau modèle, Complexity of Interaction Sequences (CIS), qui permet de prédire l'efficacité d'une technique d'interaction dans un contexte particulier. Ce modèle permet non seulement de décrire la structure d'une technique d'interaction mais aussi un contexte à travers la notion de *séquence d'interaction*. Le but de CIS n'est pas de remplacer les

autres formes d'évaluation mais d'aider à mieux comprendre l'effet du contexte sur la performance.

Après un bref aperçu des travaux antérieurs, nous présentons, tout d'abord, comment CIS décrit l'interaction, puis, comment l'outil SimCIS donne une mesure de la performance d'une technique d'interaction. Nous présentons, enfin, l'expérimentation contrôlée que nous avons réalisée pour valider CIS. Nous comparons les données expérimentales aux prédictions de CIS sur les palettes, les palettes bimanuelles et les toolglasses.

TRAVAUX ANTERIEURS

Bien que correctement réalisées, la plupart des expérimentations contrôlées ne prennent pas explicitement en compte le contexte d'utilisation. Par exemple, Kabbash et al. [4] comparent les toolglasses aux palettes d'outil sur une tâche qui impose un changement d'outil à chaque interaction. Les toolglasses [1] sont des palettes d'outils semi-transparentes et déplaçables avec la main non dominante. Pour appliquer un outil sur un objet, il suffit de cliquer sur l'objet au travers de l'outil. Avec une palette, il faut quitter l'espace de travail pour aller sélectionner l'outil puis revenir l'appliquer sur l'objet sauf si l'outil est déjà sélectionné. L'expérimentation rapporte que les toolglasses sont plus rapides que les autres palettes d'outils. Cependant, la tâche qu'ils ont utilisée semble défavorable aux palettes : le sujet n'ayant jamais à utiliser deux fois de suite le même outil, il se voit contraint à de nombreux allers retours avec les palettes.

Mackay [6] a mené une expérimentation contrôlée démontrant que les utilisateurs organisent leurs interactions selon leur contexte cognitif. Lorsque les utilisateurs sont dans un contexte "copy", c'est-à-dire qu'ils doivent reproduire un dessin, ils ont tendance à regrouper leurs utilisations d'un même outil. Lorsqu'ils sont dans un contexte "problem solving", c'est-à-dire qu'ils doivent effectuer les bonnes modifications sur un dessin, ils ont plutôt tendance à suivre leur raisonnement de travail en créant les objets incrémentalement à partir d'un focus sans minimiser les changements d'outil. Ce résultat est accompagné des préférences des

utilisateurs qui sont différentes selon le contexte : les palettes flottantes pour un contexte "copy" et les toolglass dans un contexte "problem solving".

Ces expérimentations sont coûteuses et il est impossible de tester toutes les tâches possibles. Nous avons besoin d'un modèle capable de décrire l'interaction et de prédire sa performance. De nombreux modèles existent, nous ne présentons ici que ceux qui ont un niveau d'abstraction semblable à celui de CIS.

Le modèle de Card et al. [3] décrit les périphériques d'entrées comme des traducteurs de propriétés physiques en propriétés logiques. Ce modèle permet de comparer les périphériques sur un ensemble de propriétés comme leur vitesse de pointage, leur précision, etc. Goals, Operators, Methods and Selection rules (GOMS) [4] est une famille de modèles qui décrivent l'interaction au niveau de la tâche afin de prédire le temps requis. Une tâche est décrite par un ensemble de méthodes permettant d'atteindre un but. Une méthode est une suite d'opérateurs dont le temps est prédéterminé. Les prédictions sont donc obtenues en sommant les temps des opérateurs nécessaires pour mener à bien une tâche.

Alors que les concepteurs d'interface sont confrontés au problème de choisir et combiner des techniques d'interaction, les modèles existants décrivent l'interaction soit au niveau du périphérique d'entrée soit au niveau de la tâche. CIS propose d'aborder l'interaction au niveau de la technique.

DECRIRE L'INTERACTION AVEC CIS

CIS décrit une interface comme un ensemble d'objets graphiques que l'on peut créer, modifier et/ou effacer. L'état de l'interface est l'ensemble courant d'objets et la valeur de leurs attributs. L'espace des interactions est l'ensemble des manipulations (création, modification ou effacement) disponibles à l'utilisateur dans un état donné.

Une technique d'interaction est un ensemble d'étapes organisées sous la forme d'un graphe d'interaction décrivant comment activer une manipulation. Il y a deux types d'étape : les étapes d'*acquisition* qui permettent d'identifier un sous-ensemble de l'espace des interactions (par exemple, déplacer le curseur sur un outil) et les étapes de *validation* qui valident ces réductions (par exemple, cliquer sur l'outil pour le sélectionner). Les figures 1 et 2 correspondent aux graphes d'interaction de la palette et de la toolglass contenant chacune trois outils pour créer des triangles, des rectangles et des ellipses :   . Par exemple, le chemin le plus à gauche dans la figure 1 commence par réduire l'espace des interactions à $\{(Create_triangle, p) \mid p \in position\}$ en déplaçant le curseur sur l'outil pour créer des triangles (acquisition/arc) et en cliquant

(validation/nœud). Enfin, déplacer le curseur vers la position pos puis en cliquant, l'espace des interactions est alors réduit à l'unique élément ($Create_triangle, pos$), ce qui permet d'exécuter cette manipulation.

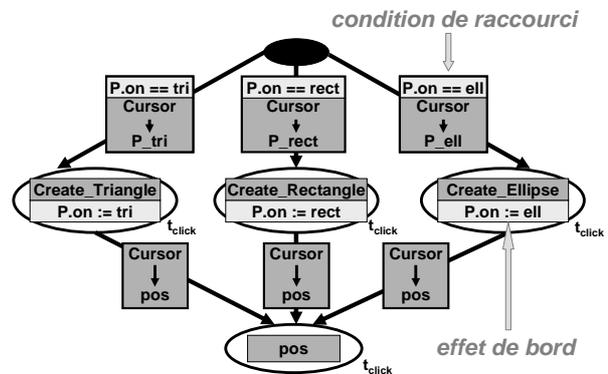


Figure 1: Graphe d'interaction de la palette d'outils.

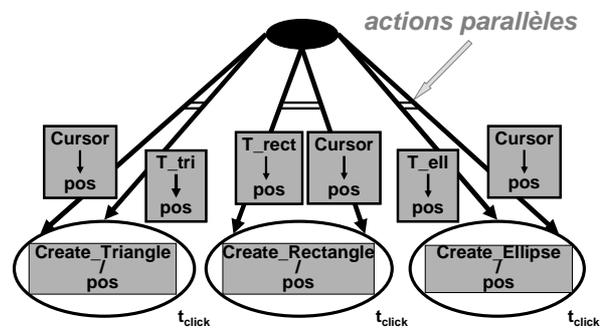


Figure 2: Graphe d'interaction de la toolglass.

CIS permet de décrire certaines propriétés des techniques d'interaction comme la *persistance* ou le *parallélisme*. Les nœuds peuvent avoir des effets de bord sur l'état de l'interface. Par exemple, la sélection d'un outil dans une palette est persistante : l'outil est actif pour les prochaines interactions. La condition de raccourci indique justement que si l'outil est actif, l'acquisition et la validation qui suivent peuvent être ignorées. Certaines étapes peuvent être *parallèles* : par exemple, le curseur et la toolglass peuvent être déplacés soit simultanément soit séquentiellement.

Les résultats rapportés par Mackay [6] mettent en avant la différence d'organisation du travail des utilisateurs et donc la différence de performance d'une technique d'interaction selon le contexte. Dans un contexte "copy", les utilisateurs créent ensemble les formes de même type alors que, dans un contexte "problem solving", ils travaillent incrémentalement autour d'un focus sans minimiser les changements d'outil. CIS opérationnalise la notion de contexte par la notion de *séquence d'interaction*. Par exemple, pour atteindre l'état S à partir de l'état S_{init} de la figure 3, un contexte "problem solving" est décrit par la séquence Seq_1 tandis qu'un contexte "copy" est décrit par la séquence Seq_2 .

Seq ₁	Seq ₂	
(Ellipse, (300, 100))	(Ellipse, (300, 100))	
(Triangle, (400, 100))	(Ellipse, (700, 100))	
(Rectangle, (500, 100))	(Ellipse, (300, 500))	
(Triangle, (300, 400))	(Rectangle, (500, 100))	
(Rectangle, (300, 300))	(Rectangle, (500, 300))	
(Ellipse, (700, 100))	(Rectangle, (300, 300))	
(Triangle, (600, 100))	(Triangle, (400, 100))	
(Triangle, (600, 200))	(Triangle, (600, 100))	
(Rectangle, (500, 300))	(Triangle, (600, 200))	
(Ellipse, (300, 500))	(Triangle, (400, 400))	
(Triangle, (400, 400))	(Triangle, (300, 400))	
(Triangle, (300, 200))	(Triangle, (300, 200))	

Figure 3: Deux séquences pour passer de S_{init} à S .

PREDIRE LA PERFORMANCE AVEC CIS

Nous avons développé l'outil SimCIS afin de prédire la performance d'une technique d'interaction pour une séquence d'interaction donnée. SimCIS prend en entrée l'état initial de l'interface, la séquence d'interaction et le graphe d'interaction de la technique pour fournir en sortie des mesures de performance : la complexité en nombre d'actions et en temps de la technique. Les complexités correspondent au nombre d'étapes (acquisition et validation) et au temps requis pour mener à bien la séquence. Pour produire ces résultats, SimCIS construit, dans le graphe d'interaction, le chemin représentant la séquence et évalue le temps de ce chemin en utilisant la loi de Hick pour les branchements, la loi de Fitts pour les arcs et les temps fixes pour les nœuds étiquetés. SimCIS fournit également une représentation graphique des actions correspondantes pour aider visuellement à comprendre les différences de résultats.

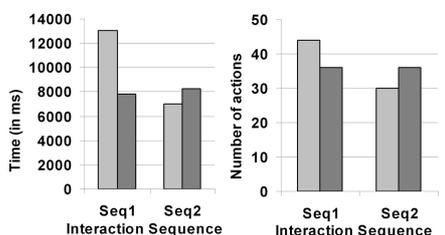


Figure 4: Comparaison des complexités [Palette - Toolglass].

CIS confirme les préférences des utilisateurs : la toolglass est plus efficace pour la séquence Seq1 ("problem solving") alors que la palette est plus efficace pour Seq2 ("copy"). Les représentations graphiques fournies par CIS montrent que la différence de complexité entre la palette et la toolglass dans un contexte "problem solving" est due aux nombreux aller-retour de l'espace de travail à la palette causés par les changements d'outil.

VALIDER CIS

Afin d'évaluer CIS, nous avons réalisé une expérimentation contrôlée pour comparer les trois techniques suivantes : Palettes (P), Palettes Bimanuelles (PB) et Toollasses (T). La Palette Bimanuelle est une technique implémentée dans le projet CPN2000 [2] : pour appliquer un outil sur un objet, l'utilisateur sélectionne l'outil à l'aide d'un curseur relié à la main

non dominante et l'objet à l'aide d'un deuxième curseur relié à la main dominante. Comme la palette, la sélection de l'outil est persistante et, comme la Toolglass, les déplacements des curseurs sont parallèles.

Cette expérience avait trois buts :

- Comparer les prédictions aux données empiriques.
- Evaluer la palette bimanuelle jamais testée jusqu'ici.
- Mieux comprendre le lien entre les propriétés d'une technique et son efficacité.

Douze adultes ont volontairement accepté non seulement de participer à l'expérimentation mais aussi de répondre à un questionnaire sur leurs préférences subjectives. Chaque sujet avait pour consigne d'effacer le plus rapidement possible un ensemble de formes dessinées à l'écran soit dans un ordre imposé par une ligne noire reliant les formes entre elles soit dans l'ordre de son choix. Chaque technique contient 3 outils correspondant aux trois types de forme (rectangle, triangle et ellipse) : pour effacer une forme, l'utilisateur doit appliquer le bon outil sur cette forme. La figure 5 montre les différents arrangements que nous avons utilisés afin d'opérationnaliser différents contextes d'utilisation.

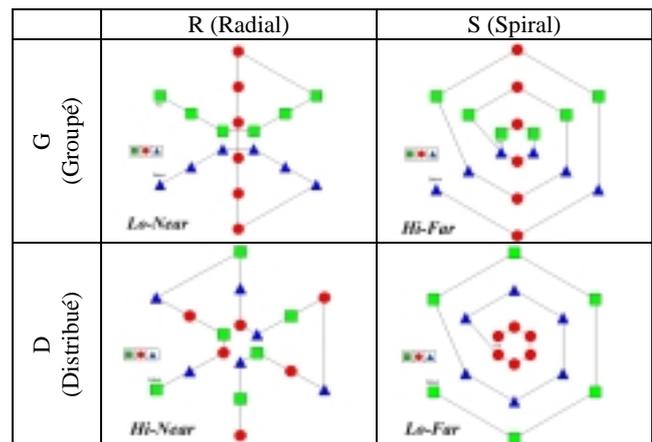


Figure 5: Les 4 types de séquence imposés (Longueur = 18).

Les conditions (3*2*2*3 inter-sujet) étaient :

- La technique : P, PB ou T
- Le groupement : G ou D
- L'ordre : R, S ou L
- La longueur : 6 ou 18 (la figure 4 ne montrent que les conditions où longueur = 18, c.a.d. 18 formes à effacer).

Pour analyser les résultats, nous avons utilisé deux types de regroupement sur la figure 5. D'une part, nous avons confronté les deux colonnes distinguant les contextes où les formes sont proches et ceux où elles sont éloignées (Near vs Far). D'autre part, nous avons confronté les diagonales distinguant les contextes demandant peu et beaucoup de changements d'outil (Lo vs Hi).

Les prédictions de CIS sur cette expérimentation sont les suivantes :

(P₁) P et PB sont sensibles au nombre de changements d'outil à cause de la propriété de persistance.

(P₂) PB est plus ou aussi efficace que les deux autres techniques car elle combine parallélisme et persistance.

(P₃) Les techniques sont sensibles aux distances entre objets à cause de l'effet de la loi de Fitts.

(P₄) Plus la séquence est longue, plus la différence est importante entre le pire et le meilleur cas car l'optimisation est fonction de l'anticipation.

La condition dans laquelle les utilisateurs sont libres de l'ordre dans lequel effacer les formes (L) et les préférences subjectives collectées lors du sondage vont nous permettre de vérifier l'hypothèse :

(H_{users}) Les utilisateurs sont capables d'optimiser l'utilisation d'une technique et de l'adapter au contexte.

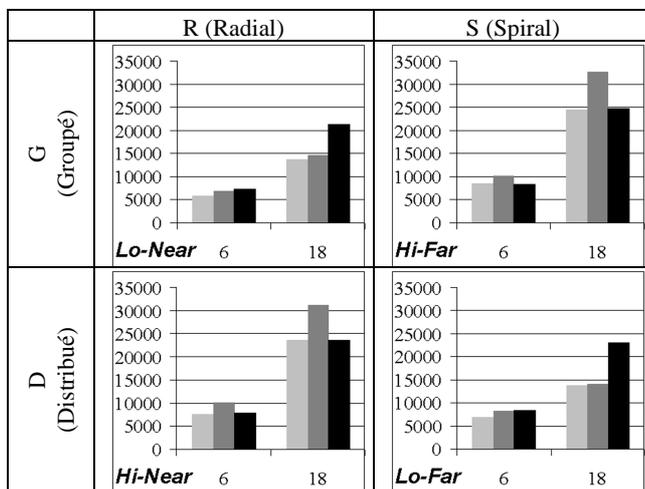


Figure 5: Données empiriques sur les séquences dont l'ordre est imposé [PB - P - T]

La figure 5 présente les données empiriques collectées :

- valide P₁ : Les palettes (P et PB) sont plus rapides dans les conditions Lo que dans les conditions Hi.
- valide P₂ : Le temps minimal pour un contexte donné est toujours celui de la palette bimanuelle.
- ne valide pas P₃ : Les histogrammes des contextes Near et Far sont très similaires, la différence n'est pas significative. CIS surestime certainement l'effet de la loi de Fitts.
- valide P₄ : Les différences entre le temps minimal et le temps maximal sont plus de trois fois supérieures lorsque longueur = 18 que lorsque longueur = 6.

Les temps réalisés sur les essais dans lesquels l'ordre n'est pas imposé sont très proches du meilleur temps

réalisé sur les essais dans lesquels l'ordre est imposé. De plus, les utilisateurs préfèrent PB pour effacer un ensemble de formes dans l'ordre de leur choix. S'ils doivent choisir entre P et T, ils préfèrent P lorsque les formes sont groupées et T lorsqu'elles sont distribuées. Ces résultats nous permettent de valider H_{users}.

CONCLUSION ET PERSPECTIVES

Cet article a présenté le modèle CIS pour décrire les techniques d'interaction et les évaluer en contexte. Nous l'avons utilisé pour montrer le lien entre contexte et efficacité. CIS est un outil pour aider à la conception d'évaluation de techniques d'interaction.

Le travail à venir peut prendre plusieurs directions. Premièrement, le modèle permet déjà la description de plusieurs techniques au sein d'une même interface mais il n'a pas été testé dans ces conditions. Deuxièmement, nous avons vu la sensibilité du modèle à l'effet de la loi de Fitts, le modèle doit donc être raffiné pour fournir de meilleures prédictions. Enfin, il faudrait automatiser la recherche de la pire et de la meilleure séquence pour atteindre un but. Ce dernier point doit aborder l'explosion combinatoire des séquences possibles.

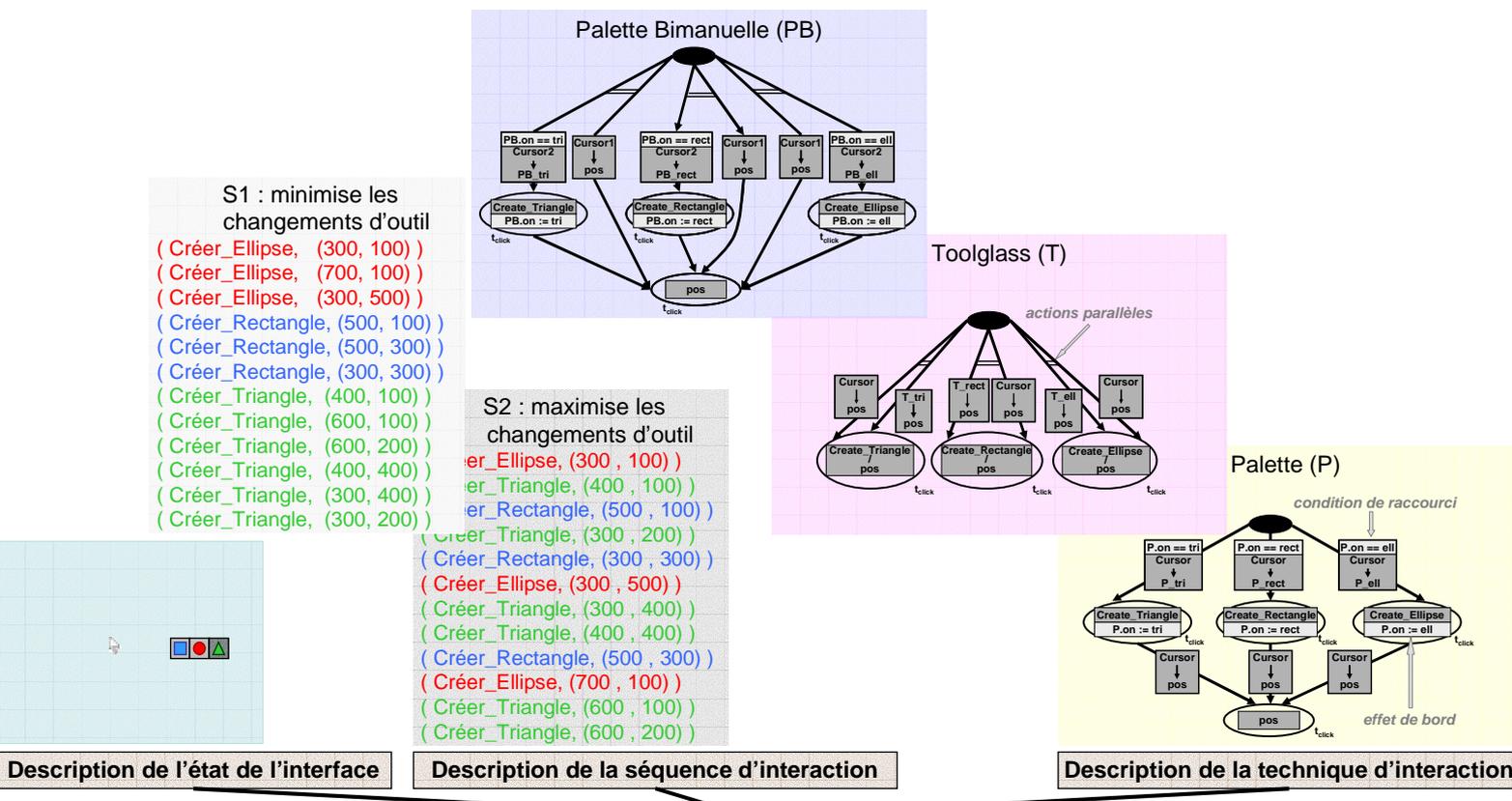
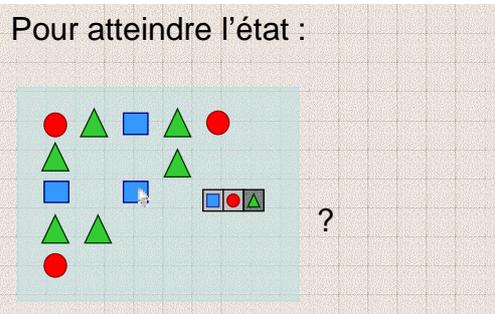
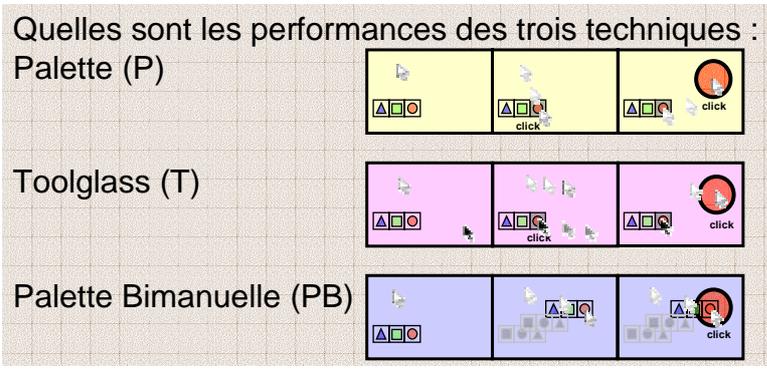
BIBLIOGRAPHIE

1. Bier, E.A., Stone, M.C., Pier, K. & Buxton, W. Toolglass and Magic Lenses: the See-Through Interface. *Proc. ACM Siggraph*. (1993). ACM Press, pp. 73-80.
2. Beaudouin-Lafon, M. & Lassen, H.M. (2000). CPN2000: A Post-WIMP Graphical Application.. *Proc. ACM Symposium on User Interface Software and Technology (UIST'00)*. p. 181-190
3. Card, S.K., Robertson, G. & Mackinlay, J. A. (1991). Morphological Analysis of the Design Space of Input Devices. *Proc. ACM Transactions on Information Systems*, 9(2), p. 99-122.
4. John, B. E. & Kieras, D. E. (1996). The GOMS Family of User Interface Analysis Techniques: Comparison and Contrast. *Proc. ACM Transactions on Computer-Human Interaction*. 3(4), p. 320-351.
5. Kabbash, P., Buxton, B. & Sellen, A. (1994). Two-handed Input in a Compound task. *Proc. ACM Human Factors in Computing Systems (CHI'94)*, pp. 417-423.
6. Mackay, W.E. (2002). Which Interaction Technique Works When? Floating Palettes, Marking Menus and Toolglasses support different task strategies. *Proc. ACM Conference on Advanced Visual Interfaces (AVI'02)*. pp. 203-208.

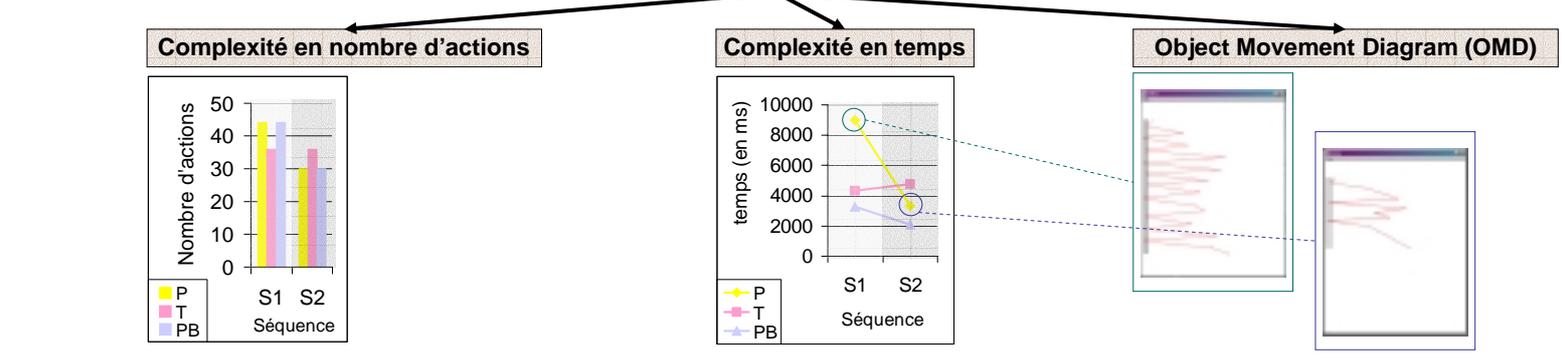
CIS : Décrire les Techniques d'Interaction et Prédire leur Efficacité en Contexte

Appert Caroline / Projet InSitu – LRI Université Paris-Sud & Inria Futurs

Complexity of Interaction Sequences (CIS) est un modèle pour **décrire** la structure des **techniques d'interaction** et opérationnaliser des contextes d'utilisation par la notion de **séquence d'interaction**. CIS peut alors **prédire** l'efficacité d'une technique d'interaction en contexte. CIS est un outil pour aider les évaluateurs à mieux comprendre l'effet du contexte sur la performance.



SimCIS



- Les palettes bimanuelles (PB) sont toujours plus efficaces quel que soit le contexte.
- Les palettes (P) sont plus sensibles au contexte que les deux autres techniques.
- Palette (P) VS Toolglass (T) :
 - P > T dans un contexte où les changements d'outils sont peu fréquents.
 - T > P dans un contexte où les changements d'outils sont fréquents.

Infrastructure Logicielle pour Interfaces Homme-Machine Plastiques

Lionel Balme

CLIPS-IMAG

385, rue de la bibliothèque
BP53, 38041, Grenoble cedex 9, France
lionel.balme@imag.fr

RESUME

Cet article traite de l'étude d'une infrastructure logicielle pour interfaces homme-machine plastiques. Après avoir précisé le contexte et le sujet de l'étude, les premiers résultats sont présentés.

MOTS CLES : Plasticité, infrastructure logicielle, décomposition fonctionnelle.

ABSTRACT

This paper describes a thesis subject about a software infrastructure to run plastic user interfaces. First, the context and the aim of the study are precised. Then, first results are presented.

KEYWORDS: Plasticity, software infrastructure, functional decomposition.

INTRODUCTION

Mon sujet de thèse concerne la plasticité des interfaces homme-machine des systèmes interactifs. Un système interactif est un assemblage de matériels et de logiciels susceptibles de fournir un ensemble cohérent de services sous le contrôle explicite d'utilisateurs. Par référence à la propriété de plasticité des matériaux qui se déforment sans se rompre, un système interactif est plastique lorsqu'il est capable de s'adapter à la diversité des contextes d'interaction¹ tout en préservant son utilisabilité [9].

Cette propriété se justifie par le fait que les progrès réalisés dans les technologies des ordinateurs, des réseaux sans fil et des capteurs permettent à l'individu de créer et de modifier lui-même son espace d'interaction. Ainsi, par exemple, la plate-forme matérielle sur laquelle s'exécute un système interactif peut-être constituées dynamiquement et de manière opportuniste par l'utilisateur au grès des ressources disponibles à l'endroit où il se trouve.

Ces nouvelles possibilités, offertes par l'informatique diffuse, induisent que les IHM ne sont plus confinées à une station de travail. Elles peuvent désormais être distribuées sur plusieurs dispositifs, migrer de l'un à l'autre

¹ Le contexte d'interaction se définit par le triplet constitué de la plate-forme, de l'environnement physique et de l'utilisateur.

et s'adapter dynamiquement au nouveau contexte de l'interaction. Elles deviennent plastiques.

Le cadre de référence CAMELEON, issu des travaux de [9], distingue de manière explicite le développement d'IHM plastiques de leur exploitation. Si les principes des phases amont y sont bien détaillés, peu de choses sont dites sur l'exploitation. Mon travail consiste à donner corps à ces phases en aval.

MOTIVATION

La phase d'exploitation d'un système interactif plastique, c'est à dire son exécution, pose plusieurs problèmes.

L'adaptation à la diversité des plates-formes dépasse le problème de la portabilité des logiciels telle que l'entend l'approche par machine virtuelle façon JVM. L'expérience montre que l'IHM d'un système ne peut être identique pour un téléphone portable, un calculateur de poche et une station de travail : les différences de ressources d'interaction, par exemple la variabilité de la taille de l'écran, voire l'absence de dispositif d'entrée comme le clavier, impliquent parfois un profond remodelage de l'IHM.

L'adaptation à la diversité des environnements d'interaction ajoute à la complexité de la diversité des plates-formes. Avec la miniaturisation, l'autonomie des batteries, et les progrès de la communication sans fil et des capteurs de toutes sortes, l'accès aux ressources de calcul devient possible en tout lieu et à toute heure. Il en résulte des environnements d'interaction extrêmement diversifiés où les frontières entre milieux familial et professionnel, privé et public, s'estompent.

La diversité des plates-formes et des environnements d'interaction engendre potentiellement une infinité de situations d'exécution. Il est alors impossible, pour un développeur de système interactif, de toutes les prévoir lors des phases de conception du logiciel.

Si l'ensemble des situations n'est pas prévisible lors du processus de conception, certaines d'entre elles sont néanmoins identifiables à ce stade. Pour ces situations d'exécution identifiées à l'avance, un système interactif plastique peut embarquer en son sein des mécanismes d'adaptation capable de les prendre en compte. Les mécanismes d'adaptation embarqués définissent le domaine

de plasticité du système interactif, c'est à dire sa capacité à s'adapter aux situations rencontrées.

Si une situation d'exécution rencontrée ne fait pas partie du domaine de plasticité du système interactif, celui-ci ne peut pas s'adapter seul. Il doit alors pouvoir se reposer sur des mécanismes extérieurs pour l'être, en fonction des ressources matérielles et logicielles disponibles et de la configuration de l'environnement physique de l'interaction.

OBJECTIFS

L'objectif général de cette thèse est de concevoir une infrastructure logicielle qui sous-tend l'exécution d'interfaces homme-machine plastiques. Une infrastructure logicielle s'entend ici comme un ensemble cohérent de services et de mécanismes généraux utiles à l'exécution de toute IHM plastique. A ce titre, cette infrastructure s'apparente à un intergiciel.

Afin de construire une infrastructure qui soit adaptée aux besoins humains, il convient au préalable d'identifier les propriétés et services généraux attendus par les utilisateurs vis-à-vis de la plasticité. Ce premier objectif atteint, je serai en mesure de proposer une taxonomie des systèmes interactifs plastiques. A son tour, cette taxonomie permettra d'analyser l'état de l'art et d'en identifier les lacunes.

Si les premiers objectifs sont centrés utilisateur (identification de propriétés, élaboration d'une taxonomie et analyse de l'état de l'art des IHM plastiques existantes), les objectifs suivants sont orientés système. Dans cette étape, l'objectif est de traduire les requis utilisateur en leur équivalent système. En regard de ces requis système, je propose comme dernier objectif, une architecture logicielle conceptuelle ainsi que des mécanismes génériques pour la mise en œuvre d'interfaces homme-machine plastiques.

L'intérêt essentiel de ce travail est d'obtenir des résultats réutilisables qui respectent le principe de la séparation fonctionnelle. En interaction homme-machine, cette séparation est appliquée de la manière suivante : Le noyau fonctionnel (l'ensemble des services du système) est un composant distinct de l'IHM. L'IHM se décompose en composants de niveaux d'abstraction distincts en sorte de favoriser la modifiabilité. Mes mécanismes et architecture doivent suivre ce même principe de décomposition, déjà éprouvé dans le domaine de l'IHM.

APPROCHE

L'approche adoptée reprend la trame de mes objectifs centrés sur l'utilisateur : des requis utilisateur, identification des requis système et de là, conception d'une architecture conceptuelle² en accord avec ces requis. Puis, mise en œuvre, évaluation et révision.

² Par opposition à une architecture implémentaire qui désigne un ensemble organisé de composants programmés exécutables [3].

Pour la solution technique, j'opte pour une approche centrée architecture que motivent les requis des systèmes répartis auto-adaptables et dynamiquement configurables (Nous avons vu en introduction que les IHM non seulement se remodelent, mais aussi peuvent migrer et se distribuer.) L'Interaction Homme-Machine, pour sa part, dispose de modèles d'architecture de référence [7]. Je retiens le modèle ARCH pour sa puissance conceptuelle et l'étendue de son application [2]. La recherche en plasticité des IHM propose, pour sa part, un cadre de référence orienté modèle [6].

PREMIERS RESULTATS

Les premiers résultats obtenus dans le cadre de ce travail sont de nature théorique et pratique. Sur le plan théorique, l'identification des requis utilisateur m'ont permis de caractériser précisément l'espace problème. De cet espace problème, j'ai déduis des requis système qui m'ont permis ensuite d'élaborer une architecture conceptuelle. Enfin, sur le plan pratique, une première implémentation de cette architecture est en cours de développement.

Espace problème

L'identification des requis utilisateur pour la plasticité fait apparaître un ensemble de propriétés orthogonales au domaine de l'application. Par nature ces propriétés sont génériques et ont vocation à être toutes ou parties présentes dans un système interactif plastique quel qu'il soit. Elles constituent des propriétés caractéristiques d'un système interactif plastique tel que le perçoit l'utilisateur.

La présence d'un sous-ensemble des ces propriétés dans une IHM peut permettre d'en évaluer le domaine de plasticité.

Ces propriétés s'organisent autour de dix axes. Faute de place ici, je ne détaillerai que sommairement ces axes. Une description plus précise de ceux-ci est disponible dans [5].

Le premier axe caractérise le type de la plate-forme. La plate-forme est l'ensemble des ressources matérielles et logicielles sur lequel s'exécute le système interactif. Elle peut être élémentaire ou une composition homogène ou hétérogène de plates-formes élémentaires [4].

Le deuxième axe concerne la sensibilité au contexte d'interaction. Je caractérise ici les différents canaux susceptibles de déclencher une adaptation du système interactif.

Quatre axes caractérisent les différentes facettes d'une adaptation. Ces axes permette d'exprimer les techniques d'adaptations employées par le système interactif, l'initiateur de l'adaptation, le grain de reprise (par exemple, est-ce que l'adaptation peut avoir lieu à la volée, ou doit elle s'effectuer entre deux sessions d'utilisation ?) et la précision de l'état de reprise (par exemple, est-ce que des données ou leur représentation ont été perdues ou dégradées lors de l'adaptation).

Les trois axes suivants caractérisent l'IHM elle-même. Ces axes permettent de décrire le type de l'IHM (est-elle distribuée, centralisée, mono ou multi modale ?), les types d'IHM de transitions employées et les types de migration possibles.

Enfin, un dernier axe décrit les différentes contraintes temporelles qui pourrait être associées à l'adaptation, notamment la latence du système.

Ces dix axes de caractérisation des systèmes interactifs plastiques constituent un outil nécessaire pour réaliser une revue critique comparative de l'état de l'art des techniques de mise en œuvre des IHM dynamiquement adaptables au contexte d'interaction.

Architecture conceptuelle

L'espace problème ainsi défini implique que, du point de vue logiciel, un système interactif plastique soit auto-reconfigurable et distribué sur un ensemble hétérogène de ressources d'interaction. Au cours de l'exécution, la composition de cet ensemble est susceptible d'être modifiée, ainsi que les conditions environnementales dans lesquelles se déroule l'interaction.

La littérature des systèmes autoreconfigurables montre qu'une approche par composants réflexifs et introspectifs semble la plus adaptée à notre problème.

[8] identifie deux classes de systèmes autoreconfigurables : les systèmes à reconfiguration interne (close-adaptive) et les systèmes à reconfiguration externe (open-adaptive). Une reconfiguration interne suppose qu'un composant embarque toutes les données et le code nécessaires pour s'adapter. Une reconfiguration externe fait l'hypothèse qu'un mécanisme extérieur au compo-

sant est en charge de mettre en œuvre l'adaptation.

En ce qui nous concerne, les deux approches sont complémentaires. En effet, un composant est caractérisé par un domaine de plasticité, c'est-à-dire un ensemble de variations du contexte qu'il est capable de gérer par lui-même. Si la variation du contexte sort de ce domaine, alors un mécanisme externe, qui embarque une vision plus globale du système est nécessaire pour mettre en œuvre l'adaptation. Ainsi, les notions d'introspection et de réflexivité doivent être présentes à deux niveaux : au niveau des composants, pour leur permettre de gérer leur domaine de plasticité, et au niveau de l'infrastructure générale, pour permettre une adaptation plus complexe. L'infrastructure d'exécution proposée (figure 1) est présentée en détail dans [1]. Je n'en présente ici qu'une vue d'ensemble.

Cette infrastructure se décompose en trois couches. Celle de bas niveau concerne la plate-forme d'exécution. La couche intermédiaire constitue l'intergiciel en charge de la plasticité. A ce niveau, on trouve des mécanismes dédiés à la détection du contexte, un gestionnaire de plate-forme et des boîtes à outils d'interacteurs.

La partie innovante de cet intergiciel est constitué du gestionnaire de reconfiguration externe chargé d'adapter les systèmes interactifs en cours d'exécution quand ceux-ci ne peuvent plus le faire eux-mêmes. Ce gestionnaire est composé de trois modules principaux : Un identificateur de situation qui fait la synthèse des informations issues du système interactif, de l'utilisateur, de la plate-forme et de l'environnement physique. Il détecte les changements de situation et notifie un moteur

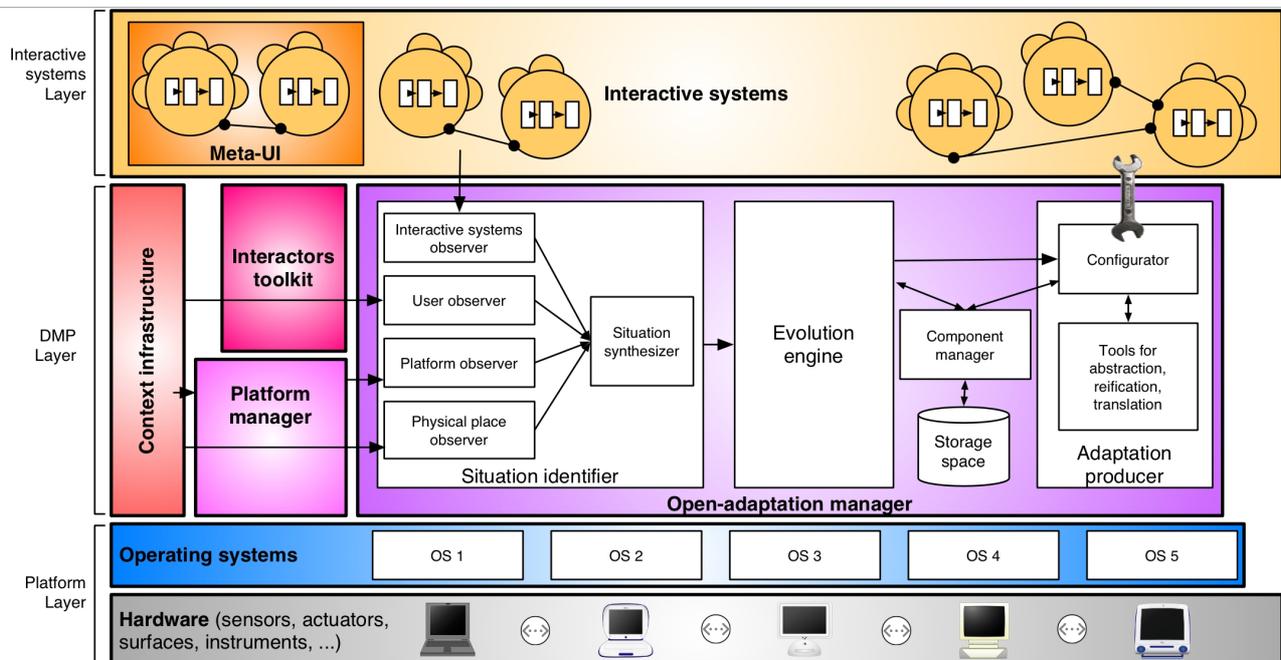


Figure 1 : Architecture conceptuelle pour l'exécution des systèmes interactifs plastiques. Les éléments en forme de fleur, , représentent des composants à reconfiguration externe. Les miniatures en forme de gestionnaire d'adaptation, , représentent les composants capables de reconfiguration interne. Les flèches dénotent des flux d'information, les traits des liens bidirectionnels.

d'évolution. Celui-ci calcule une solution d'adaptation et la propose au producteur de l'adaptation. Celui-ci est chargé de la mettre en œuvre.

La couche supérieure est celle des systèmes interactifs en cours d'exécution. L'un d'eux, particulier, est appelé Méta-IHM. Il regroupe des éléments comme les interfaces de transition, chargées de rendre observable à l'utilisateur chaque adaptation du système, et les éléments d'IHM des composants de l'intergiciel en charge de la plasticité.

CONCLUSION

L'état de l'art des systèmes interactifs plastiques fait apparaître un manque de modèles et d'outils pour supporter les phases d'exécution de ces systèmes. Mon travail consiste à concevoir une infrastructure pour supporter l'exécution de systèmes interactifs plastiques. Pour cela, mon approche a été de proposer une définition précise de l'espace problème, d'identifier des requis systèmes à partir de cet espace problème, pour ensuite proposer une architecture conceptuelle d'une infrastructure d'exécution.

Il reste maintenant à éprouver cette architecture conceptuelle en fournissant un exemple d'implémentation. Actuellement, dans le cadre du projet européen CAMELEON, une implémentation de l'infrastructure proposée est en cours de réalisation. Elle permettra de supporter des applications web comme des applications dites classiques.

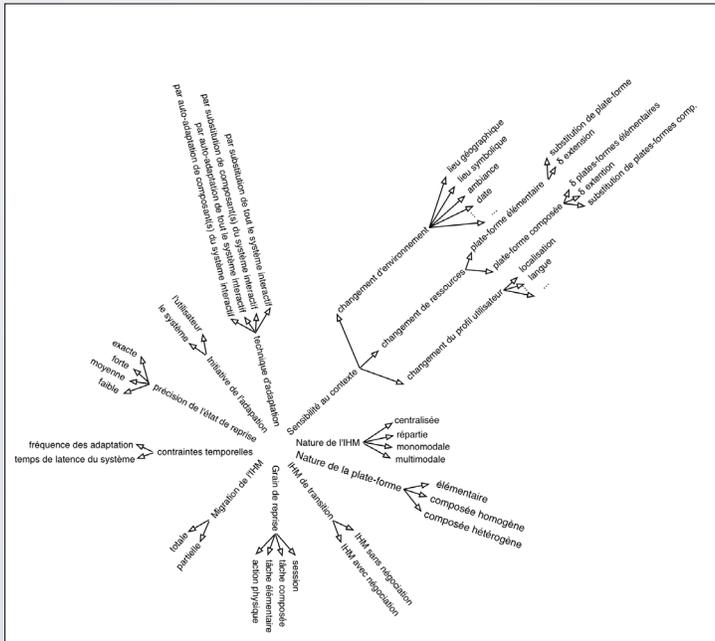
BIBLIOGRAPHIE

1. Balme L., Demeure A., Barralon N., Coutaz J., Calvary G., CAMELEON-RT : A Software Architecture Reference Model for Distributed, Migratable and Plastic User Interfaces, à paraître dans les actes de EUSAI 2004.
2. Bass L., Little R., Pellegrino R., Reed S., Seacord R., Sheppard S., The Arch Model : Seeheim Revisited (version 1.0). The UIMS Tool Developers Workshop (Avril 1991). in ACM SIGCHI Bulletin Vol.24, No 1, Janvier 1992.
3. Bass L., Clements P., Kazman R., *Software Architecture in Practice*. Addison Wesley Publ., ISBN 0-201-19930-0 (1998).
4. Calvary G., Bouillon L., Paternò et al. The CAMELEON Reference Framework, Deliverable D1.1, Décembre 2002.
5. Coutaz J., Balme L., Barralon N., Calvary G., Demeure A., Lachenal C., Rey G., Bandelloni, R., Paternò, F. The CAMELEON Reference Framework, Deliverable D2.2, September 2003.
6. Calvary G., Coutaz J., Thevenin D., Supporting Context Changes for Plastic User Interfaces: a Process and a Mechanism, in *Proc. HCI-IHM 2001*, A. Blandford, J. Vanderdonck, P. Gray Eds., BCS conference series, Springer Publ., pp. 349-363.
7. Coutaz J., Architectural Design for User Interfaces; *The Encyclopedia of Software Engineering*, J. Marciniak Ed., Wiley & Sons Publ., seconde édition, 2001.
8. Oreizy P., Taylor et al, An Architecture-Based Approach to Self-Adaptive Software. In IEEE Intelligent Systems. Pp. 54-62, May-June, 1999.
9. Thevenin D., Adaptation en Interaction Homme-Machine: Cas de la plasticité. Thèse de doctorat de l'université Joseph Fourier, Grenoble, 2001.



Infrastructure logicielle pour IHM plastiques

Espace problème de la plasticité des IHM à l'exécution



Flocon de l'espace problème pour l'adaptation des IHM plastiques dynamiques.

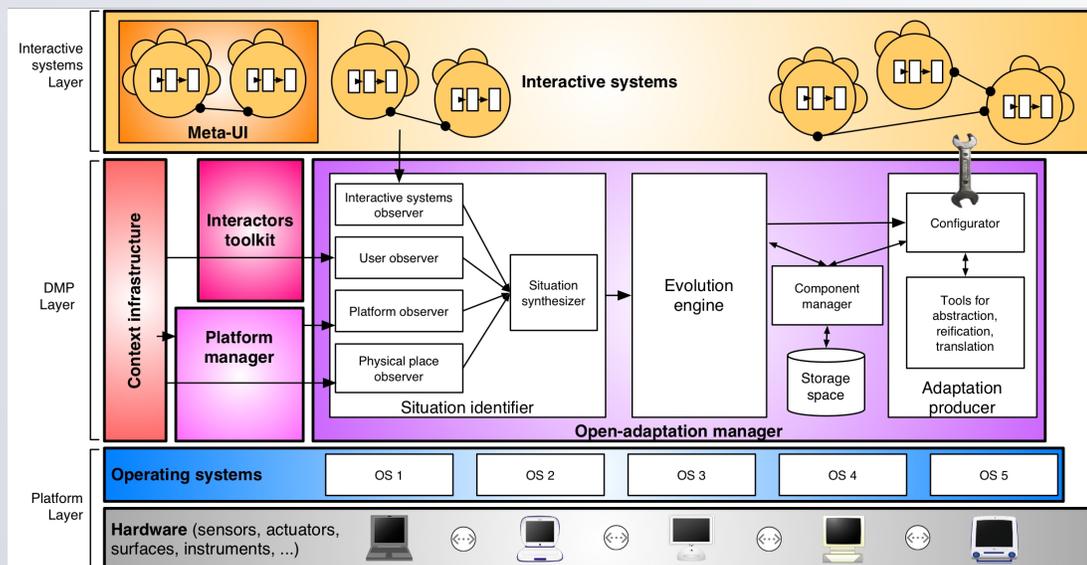
L'identification des requis utilisateur pour la plasticité fait apparaître un ensemble de propriétés orthogonales au domaine de l'application.

Par nature ces propriétés sont génériques et ont vocation à être toutes ou parties présentes dans un système interactif plastique quel qu'il soit.

Elles constituent des propriétés caractéristiques d'un système interactif plastique tel que le perçoit l'utilisateur.

La présence d'un sous-ensemble des ces propriétés dans une IHM peut permettre d'évaluer le domaine de plasticité.

Une décomposition fonctionnelle pour supporter l'exécution de systèmes interactifs plastiques



Les éléments en forme de fleur, , représentent des composants à reconfiguration externe.

Les miniatures en forme de gestionnaire d'adaptation, , représentent les composants capables de reconfiguration interne.

Les flèches dénotent des flux d'informations, les traits des liens bidirectionnels.



Meta UI : vers un Desktop++

Nicolas Barralon

CLIPS-IMAG, Université Joseph Fourier
38041, Grenoble Cedex 9, France
nicolas.barralon@imag.fr

RESUME

Cet article démontre le besoin d'un nouveau composant pour l'informatique ambiante : le Meta UI (Meta User Interface) ou super-desktop de demain. La création de ce nouveau composant part du constat que les services fournis par le bureau classique ne sont plus suffisants. Il convient dorénavant de prendre en charge : l'emprunt, le prêt et la composition de ressources d'interaction ; la migration et la distribution des interfaces Homme-Machine, voire la plasticité des IHM avec ses services de négociation et de transition.

MOTS CLES : Desktop, couplage de ressource d'interaction, interaction multisurface multi-instrument.

ABSTRACT

This paper demonstrates the need for a new component in the HCI development of interactive systems: the meta-UI (meta User Interface). The meta-UI is a kind of super desktop that provides users with the capacity to access new aspects of ambient computing. These include: borrowing, coupling, and lending interaction resources; migrating, distributing, and plastifying user interfaces; and means for negotiating and for understanding system transitions between relevant states.

KEYWORDS : Desktop, coupling interaction resources, multisurface multi-instrument interaction.

INTRODUCTION

Le domaine de l'informatique subit régulièrement de profonds bouleversements provoqués par l'émergence de nouvelles technologies. Aujourd'hui, une nouvelle évolution se prépare avec la convergence des réseaux ad hoc sans fil, des micro et nano systèmes et la maturation de trente années de recherche en systèmes répartis, en perception artificielle et en interaction homme-machine

(IHM). Cette convergence de technologies permet d'envisager le passage d'une informatique confinée et statique, à une informatique ambiante favorisant l'émergence opportuniste d'écosystèmes. Dans cette vision, les espaces interactifs prennent la forme d'îlots autonomes. Ces derniers sont sans cesse en mutation puisque l'utilisateur peut dynamiquement faire apparaître, composer, prêter ou emprunter les ressources d'interaction disponibles dans son espace. Je mets en exergue les services que ne couvre pas le bureau actuel, avant de proposer des éléments de solution.

LES LACUNES DU BUREAU (DESKTOP)

La métaphore du bureau supposant un utilisateur devant un ordinateur composé d'un clavier, d'un écran et d'une souris n'est plus suffisante. Cependant, il est nécessaire, de continuer à fournir les services d'interface. L'utilisateur doit, par exemple, pouvoir gérer son espace d'affichage et ses dispositifs d'entrée. De plus, la mobilité des utilisateurs rend la notion d'espace publique, semi privé, privé particulièrement importante. Plusieurs utilisateurs peuvent par exemple partager un mur interactif. Dans ces conditions, l'utilisateur doit pouvoir percevoir toutes ces nouvelles informations (partage, disponibilité des ressources d'interaction) ainsi que les modifier. Je traite ces problèmes dans les sections « couplage de surfaces », « couplage d'instruments ». Si le bureau actuel gère l'affichage des fenêtres des systèmes interactifs sur un écran (deux au maximum), il est incapable de prendre en charge l'affichage sur un ensemble de machines. Dans la section « Gestion de l'affichage », j'analyse les problèmes de distribution et de migration. Ces deux derniers points, imposent, pour garantir l'utilisabilité des IHM que celles-ci soient plastiques [1] afin de s'adapter aux changements de l'espace interactif. Je présente donc, le Meta UI comme réponse à ces problèmes.

META UI : NOUVEAUX SERVICES

Les IHM conventionnelles WIMP (Windows, Icons, Menus, Pointing) supposent, l'existence de dispositifs d'interaction dont le nombre et le type sont connus par avance. Avec la réalité mixte (RM), tout objet familier augmenté est un dispositif d'interaction donnant lieu à une grande diversité de types. Avec l'informatique ambiante, à la diversité des types s'ajoute la présence incertaine de dispositifs qui, nous l'avons vu en introduc-

tion, peuvent être empruntés, prêtés, composés et découverts dynamiquement. Ces dispositifs, moyens pour construire des espaces interactifs, prennent le statut de ressource. Parce qu'ils sont nécessaires à l'interaction, je les nomme *ressources d'interaction*.

La diversité des ressources d'interaction est une source de complexité. En réponse à ce problème, nous avons introduit la notion unificatrice de *rôle* [2]. Désormais, une entité physique de l'environnement est une ressource d'interaction si elle joue un rôle de *surface* et/ou d'*instrument*. Le rôle de surface confère à l'entité la capacité de montrer l'état du système, le rôle d'instrument, celle d'en modifier l'état. Par exemple, un mur peut jouer un rôle de surface et un stylo, celui d'instrument. L'écran d'un PDA sert de surface mais aussi d'instrument lorsque son orientation a un effet sur l'état du système [3].

Dans les IHM conventionnelles, la gestion des surfaces est prise en charge par le système d'exploitation et est configurable par l'utilisateur. Grâce aux cartes graphiques multi-écran un utilisateur peut étendre la surface d'affichage de son bureau. Ici il s'agit d'une solution ad hoc restreinte à deux écrans et à une connexion filaire. Reste à la charge de l'utilisateur de spécifier la configuration spatiale physique des surfaces (telle surface est à droite de la surface principale, ...).

La gestion des instruments est encore plus triviale : la souris et le clavier doivent être présents et les seuls degrés de liberté pour l'utilisateur sont de brancher ou non ces dispositifs et de configurer, par exemple, l'accélération de la souris et la vitesse de répétition de frappe du clavier. Le Meta UI va permettre à l'utilisateur d'aller plus loin dans l'interaction ambiante.

Pour modéliser la composition dynamique d'un espace interactif, j'utilise la notion de couplage [4] entre ressources d'interaction. Le *couplage de deux ressources d'interaction* est l'action de lier ces ressources de manière à ce qu'elles opèrent conjointement pour fournir une nouvelle fonction.

Couplage de surfaces

J'illustre ici, le couplage de surface par un exemple afin de mettre en exergue le rôle du Meta UI. Bob et son collègue se rencontrent dans la rue et vont de façon opportuniste coupler leur PDA (Figure 1) afin d'obtenir une surface d'affichage plus grande pour leur collaboration.

Dans ce scénario, comment les deux collègues peuvent-ils prédire les effets du couplage ? Cette question relève clairement des propriétés d'observabilité et de prévisibilité développées dans [7]. Du point de vue ergonomique

le développement du Meta UI devra prendre en compte ces propriétés afin de garantir l'utilisabilité des services de couplage.

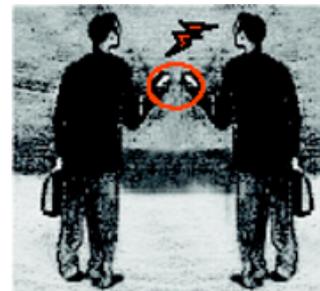


Figure 1. Couplage de PDA

D'autres propriétés de souplesse et de robustesse sont pertinentes pour notre problème. Dans l'exemple, l'atteignabilité se caractérise par la capacité de l'utilisateur de coupler à tout instant son PDA. Le Meta UI doit donc supporter des techniques d'interaction pour la réalisation du couplage. Par exemple Hincley[5] propose de coupler des tablettes graphiques en les heurtant. Une métaphore similaire, par contact, permet le couplage de ConnecTables dans [6]. Dans ce dernier exemple, le couplage réalise la fusion des deux espaces de travail des deux tablettes posant le problème de la gestion de l'espace privé. Dans le cas de notre exemple de couplage de PDA, les données privées de Bob vont-elles être partagées ? Comment exprime-t-il que telle ou telle information peut être communiquée à son collègue ? Qu'advient-il du contenu de leur travail collaboratif lors du découplage (propriété de curabilité) ?

Dans un environnement composé de plusieurs surfaces, quelles surfaces sont atteignables ? Quelle est la limite de l'espace d'affichage ? La figure 2 illustre cette dernière question en présentant trois surfaces A, B, C couplées pour fournir un espace d'affichage étendu. On représente par un trait épais les différentes « frontières » de l'espace d'affichage sur les surfaces et des fenêtres diversement positionnées pour montrer les avantages et inconvénients des différentes solutions. Les parties en pointillés sont les parties non visibles. À droite de chaque exemple se trouve une représentation des véritables frontières.

En haut de la figure 2, l'espace est maximal, mais on constate de nombreux « trous ». Les frontières affichées ne permettent certainement pas à l'utilisateur de comprendre les réelles limites de l'espace. Au centre de la figure 2, les frontières affichées permettent probablement à l'utilisateur de comprendre les limites de l'espace, mais contraignent le déplacement de la fenêtre trop grande de A vers B. En bas de la figure 3, une solution hybride est envisagée mêlant les deux premiers cas. Il ne s'agit pas ici de prendre parti pour l'une ou l'autre des solutions

mais de montrer sur un exemple simple la difficulté de rendre seulement observable le couplage de surfaces. Le choix pour telle ou telle solution ne pourra s'obtenir que par l'expérimentation, validant de fait l'affordance [11] des solutions proposées. En déclinant toutes les propriétés ergonomiques de [7] on se rend compte de la complexité des problèmes rencontrés par le Meta UI à ce sujet.

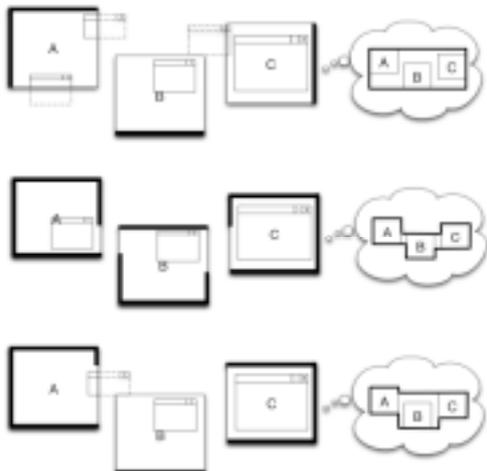


Figure 2. Frontières d'espace d'affichage

Après le couplage de surface, je m'intéresse maintenant au couplage d'instruments, en particulier le couplage clavier-souris.

Couplage d'instruments

Dans les IHM WIMP actuelles, l'instrument de pointage et l'instrument de saisie de caractères sont couplés dès lors qu'ils sont connectés au calculateur. La fonction résultant de ce couplage est immuable : l'instrument résultant de ce couplage est immuable : l'instrument de pointage détermine le "focus" de l'instrument de saisie. Il lui confère la capacité de multiplexage spatial. L'adjonction d'une deuxième souris au même calculateur permet (dans le meilleur des cas) de contrôler l'unique pointeur avec les deux dispositifs.

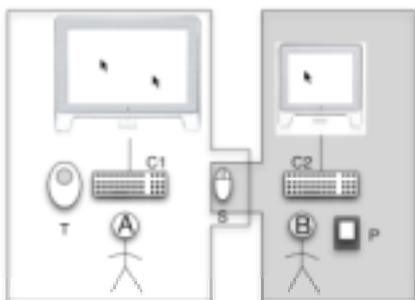


Figure 3. Couplage d'instruments

Dans le contexte de l'informatique ambiante, l'utilisateur peut interagir avec plusieurs instruments (ici le couple claviers-souris) et les partager avec d'autres utilisateurs. Dynamiquement, le Meta UI doit rendre possible le cou-

plage de n'importe quel clavier avec n'importe quelle souris. La figure 3 montre un exemple simple dans lequel l'utilisateur A a couplé son trackball T avec la souris S1 pour faire de l'interaction bi-manuelle. Le clavier est couplé avec S1 et T. De son côté, B a couplé S avec son clavier. Il peut dynamiquement coupler son PDA avec la souris S. Comment exprime-t-il qu'il veut coupler son PDA ? Par quelle métaphore ?

La Figure 4 montre un exemple un peu plus complexe de couplage clavier-souris. Les deux souris S1 et S2 sont couplées pour manipuler le même curseur ; les deux claviers C1 et C2 également couplés pour ne fournir « qu'une seule file d'événements ».

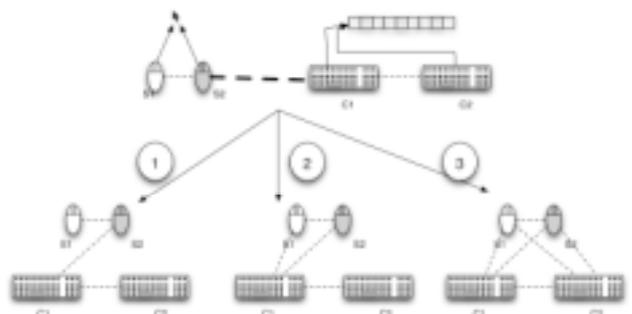


Figure 4. Couplage et transitivité

Le couplage de S2 avec C1 peut être interprété de diverses manières :

- Seuls S2 et C1 sont couplés. Dans ce cas, les événements clavier sont pris en compte que lorsque le curseur qui a donné le focus a été manipulé par S2
- Par transitivité, S1 est également couplé avec C1. Ainsi, quelle que soit la souris qui a piloté le curseur, les événements claviers de C1 sont pris en compte.
- La fermeture transitive est réalisée. Quelle que soit la souris qui pilote le curseur, n'importe quel événement clavier (de C1 ou C2) est pris en compte

L'intuition nous guide vers la troisième solution, mais elle ne correspond peut-être pas au souhait de l'utilisateur. Une négociation est toujours possible entre le Meta UI et l'utilisateur pour trancher, mais elle rallonge d'autant la trajectoire d'interaction du couplage. Pour être efficace, le Meta UI doit pouvoir prédire l'intention de l'utilisateur.

Gestion de l'affichage

Un autre rôle du bureau / window manager est la gestion de l'affichage des fenêtres. Avec un bureau classique, les fenêtres sont confinées au seul écran d'une machine. Grâce au couplage de surface, l'utilisateur va pouvoir déplacer ses pointeurs souris hors de son propre écran et

ainsi, de fait, pourvoir migrer et/ou distribuer ses IHM. Dans *Augmented Surfaces* [8] Rekimoto illustre ce point en migrant une partie de l'IHM de son portable sur une table augmentée. Le *DynaWall* [9] permet une interaction similaire, à l'aide de trois tableaux augmentés, à un grain de distribution différent. Dans *Augmented Surfaces*, l'IHM que l'utilisateur manipule n'est présente que sur une seule surface à la fois (soit le portable soit la table) alors que les IHM du *DynaWall* peuvent être « à cheval » sur deux tableaux différents, garantissant une continuité visuelle. Dans ces deux cas cependant, les configurations des machines sont fixes et leur agencement spatial pré-établi. Le *Meta UI* en tant que *window manager* se doit de prendre en compte ces aspects de distribution et migration de l'affichage puisqu'ils ont un impact direct sur l'interaction.

En analysant plus précisément *Augmented Surfaces*, on s'aperçoit que la représentation des concepts manipulés est différente suivant que l'on se trouve sur le portable et/ou sur la table. Ainsi Rekimoto tente de garantir l'utilisabilité de ses IHM, ce qui introduit le prochain aspect d'une fonction du *Meta UI* : les interfaces de contrôle pour la plasticité.

Plasticité

Le terme plasticité introduit dans [1] est inspiré de la plasticité des entités physiques, biologiques, comme les plantes ou le cerveau, à s'adapter aux contraintes extérieures pour préserver une continuité d'usage. Appliquée à l'interaction Homme-Machine, la plasticité est la capacité d'un système interactif à s'adapter aux changements de l'espace interactif en préservant l'utilisabilité. Par exemple, lorsque l'utilisateur va faire migrer son logiciel de mail de sa station de travail vers son PDA, parce que toutes les fonctionnalités ne sont pas accessibles sur le PDA, une version « élégamment dégradée » [10] (sans la visualisation des pièces jointes) sera lancée. Afin de garantir la propriété d'honnêteté, le *Meta UI* va entrer en dialogue avec l'utilisateur pour l'informer de la dégradation et après une éventuelle validation, mettre en place une transition [12] pour guider l'utilisateur.

CONCLUSION ET PERSPECTIVES

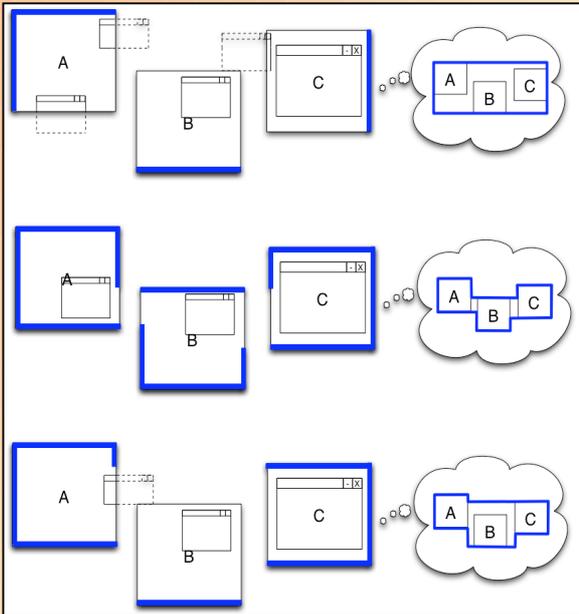
L'informatique ambiante présente de nouveaux challenges en termes d'interaction multisurface / multi-instrument et de plasticité. Les services offerts par le bureau classique ne permettent plus de garantir les propriétés d'ergonomie habituellement appliquées. Le couplage de ressources d'interaction présente un vaste terrain d'étude et une part non négligeable à l'implémentation. Cette remarque s'applique également à la gestion des fenêtres pour prendre en compte la distribution et la migration. Ceci nous amène naturellement sur le terrain de la plasticité avec la mise en place de négociations avec l'utilisateur et « d'effets » de transition.

L'objectif principal étant la conservation de l'utilisabilité.

BIBLIOGRAPHIE

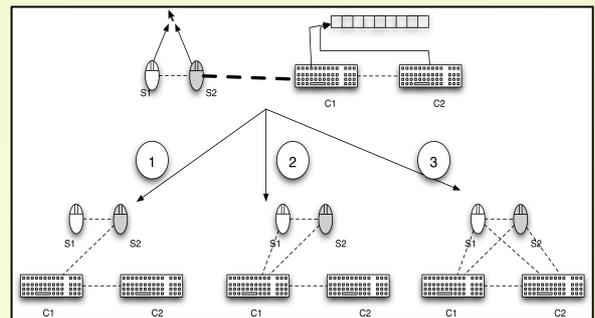
1. Calvary G., Coutaz J., Thevenin D., A Unifying Reference Framework for the Development of Plastic User Interfaces, EHCI01, Toronto, May 2001. pp.173-192.
2. Coutaz, J., Lachenal, C., Dupuy-Chessa, S. Ontology for Multi-Surface Interaction. In *Proc of Interact'03*, Zürich, 2003, pp.447-454.
3. Harrison, B. L., Fishkin, K. P., Gujar, A., Mochon C., Want, R. Squeeze me, Hold me, Tilt Me ! An exploration of Manipulative User Interface. In *Proc of ACM CHI'98*, Los angeles, 1998, p. 17-24.
4. Barralon N., Lachenal C., Coutaz J., Couplage de ressources d'interaction, IHM'04, Namur, Belgique, 2004, p13-20.
5. Hinckley, K. Synchronous Gestures for Multiple Persons. In *Proc of the ACM UIST'03*, Vancouver, Canada, 2003, pp.149-158.
6. Tandler, P., Prante, T., Müller-Tomfelde, C., Streitz, N., Steinmetz, R. ConnecTables: Dynamic Coupling of Displays for the Flexible Creation of Shared Workspaces. In *Proc of the ACM UIST'01*, Orlando, Florida, 2001, pp.11-20.
7. Gram, Ch., Cockton, G. (Eds.). *Design Principles for Interactive Software*. Chapman & Hall, London, 1996.
8. Rekimoto, J., Saitoh, M. Augmented Surfaces: A spatially continuous work space for hybrid computing environments. In *Proc of ACM CHI'99*, Pittsburgh, Pennsylvania, 1999, pp.378-385.
9. Streitz N. A., J. Geißler, T. Holmer, S. Konomi, C. MüllerTomfelde, W. Reischl, P. Rexroth, P. Seitz, R. Steinmetz i-LAND: An interactive Landscape for Creativity and Innovation. In: *ACM Conference on Human Factors in Computing Systems (CHI'99)*, Pittsburgh, Pennsylvania, USA, May 15-20, 1999. pp. 120-127.
10. Florins M., Vanderdonck J., Graceful degradation of user interfaces as a design method for multiplatform systems, 9th international conference on Intelligent user interface, 2004, pp. 140-147.
11. Norman, D. Affordance, Conventions, and Design. In the May 1999 issue of *Interactions*, 1999, pp.38-43.
12. Barralon N. Interfaces Homme-Machine de Transition. DEA Informatique Système et Communication, Institut National Polytechnique de Grenoble, Université Joseph Fourier, Juin 2002.

Meta UI : vers un desktop++



Couplage de surfaces

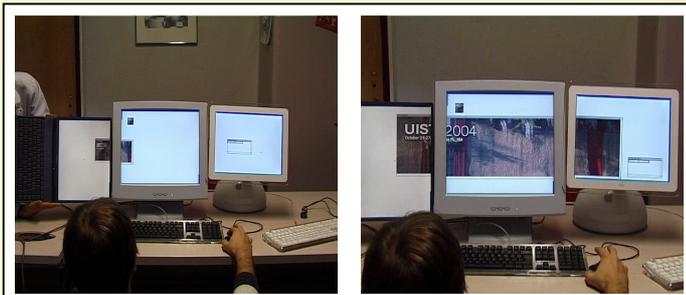
Le *couplage de deux ressources d'interaction* est l'action de lier ces ressources de manière à ce qu'elles opèrent conjointement pour fournir une nouvelle fonction.



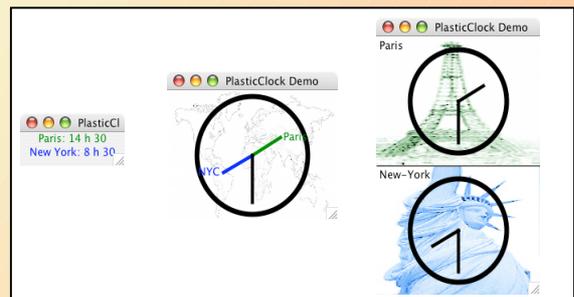
Couplage d'instruments

Nouveaux services du desktop
pour l'informatique ambiante

Distribution / migration d'IHM



Plasticité des IHM



Erreur humaine, modèles de tâches et description formelle pour la conception et l'évaluation des systèmes critiques et tolérant aux erreurs

Sandra Basnyat

LIHS – IRIT. 118, route de Narbonne, 31062, Toulouse, France
basnyat@irit.fr

RESUME

Cet article présente une façon d'apporter de la cohérence entre le modèle de tâche et le modèle du système en tenant compte de l'erreur humaine de façon à atténuer la possibilité de cas d'erreur dans les systèmes critiques interactifs. Nous proposons d'étendre un modèle de tâche classique en incorporant l'analyse des erreurs humaine. En raison de la complexité que cela introduit, nous présentons des « task patterns » réutilisables dans l'intention de réduire la quantité du travail. Le modèle de tâche étendu peut alors être systématiquement exploité sur le modèle du système correspondant pour déterminer toutes les fautes du système. Si nécessaire, le modèle du système peut être itérativement révisé pour assurer à la fois la conformité du système aux tâches des utilisateurs, et la tolérance du système aux erreurs des utilisateurs.

MOTS CLES Méthodes Formelles, Système Critique Interactif, Modèles de Tâches, Modèles de Système, Erreur Humaine.

ABSTRACT

This paper presents a way of bringing coherence between the task model and system model while taking into account human error in order to mitigate the occurrence of erroneous events in safety-critical interactive systems. We propose to extend a standard task model by incorporating human error analysis. Due to its complexity, we present re-usable task patterns as a means of reducing workload. The extended task model can then be systematically exploited on its corresponding system model to determine any system flaws. If necessary, the system model can be iteratively redesigned to ensure both system compliance to user tasks and system tolerance to user errors.

KEYWORDS Formal Methods, Interactive Safety-Critical Systems, Task & System Models, Human Error.

INTRODUCTION

Le contexte de ces travaux est à l'intersection de l'analyse, de la conception et de la validation des systèmes critiques interactifs et tolérants à l'erreur. Ceci est accompli en utilisant des techniques issues des méthodes formelles telles que la modélisation des tâches, du système et de l'erreur humaine. En conception, on utilise plusieurs types de modèles basés sur des besoins donnés, qui contribuent à la conception globale (ex. le modèle de tâche, le modèle de l'utilisateur, le modèle de

l'environnement, le modèle de la plateforme, le modèle du système, le modèle de présentation et le modèle de conception).

Nos travaux se concentrent principalement sur la modélisation des tâches et du système, en raison de l'orientation de notre équipe et de notre capacité à contribuer à ce domaine. La modélisation des tâches et du système est souvent accomplie par des experts de ces domaines. Il est improbable qu'un spécialiste en facteurs humains conçoive le modèle de tâche puis le modèle du système. Il peut ainsi exister des incohérences entre les deux. Par exemple, une tâche aurait pu être incorporée dans un modèle de tâche qui ne peut pas être accomplie avec le modèle du système. C'est pourquoi nous croyons qu'il est essentiel de garantir la cohérence entre les deux modèles, afin de tester la complétude du système et sa capacité à prendre en charge toutes les tâches de l'utilisateur (nécessaires pour atteindre un but donné) et pour réduire l'occurrence d'accident ou d'incident. Cependant, non seulement les tâches standard de l'utilisateur devraient être prises en compte, mais aussi, les cas erronés. La modélisation des tâches est normalement accomplie sans tenir compte des erreurs des utilisateurs. C'est seulement plus tard pendant le déroulement du cycle de vie, durant les phases essais, que les cas erronés sont découverts. Cette approche préventive permet de réduire la probabilité des cas erronés en concevant un système qui supporte de tels cas. Ceci peut être obtenu en concevant un système qui retournera dans un état sûr après qu'un cas problématique ait eu lieu. Les erreurs humaines jouent un rôle primordial dans l'occurrence d'accidents dans les systèmes critiques tels que l'aviation, les systèmes ferroviaires ou les centrales nucléaires [17]. Ces travaux sont donc nécessaires pour diminuer la possibilité et les effets d'un comportement erroné dans de tels environnements.

VUE D'ENSEMBLE DES TRAVAUX

Dans ce papier et en partie dans la thèse, nous proposons de rendre cohérent le modèle de tâche et le modèle du système, tout en introduisant l'erreur humaine. Jusqu'à présent, nous avons proposé une façon de prendre systématiquement en considération le comportement erroné de l'utilisateur. Ces travaux sont basés sur des résultats précédents dans les domaines de l'analyse de tâche, de la modélisation de tâche et de l'analyse et de l'identification des erreurs humaines.

Nous proposons de définir et d'utiliser les « task patterns » pour traiter la complexité et les répétitions qui apparaissent fréquemment lors de la modélisation des comportements erronés des utilisateurs. Le modèle de tâche peut ensuite être joué sur le modèle du système pour vérifier tous les scénarii possibles et, si nécessaire, dans un processus itératif, de re-concevoir le modèle du système. La Figure 1 montre une version modifiée du processus de maintien de cohérence du modèle du système et du modèle de tâche [10], destiné à démontrer nos idées et l'état d'avancement de nos travaux. La modification est l'addition de l'analyse de la tâche et de l'erreur humaine, ainsi qu'une partie fusion, illustrée par la ligne pointillée. Les quatre phases de modélisation peuvent être identifiées par les flèches circulaires contenant des processus itératifs. Pour plus d'information voir [10].

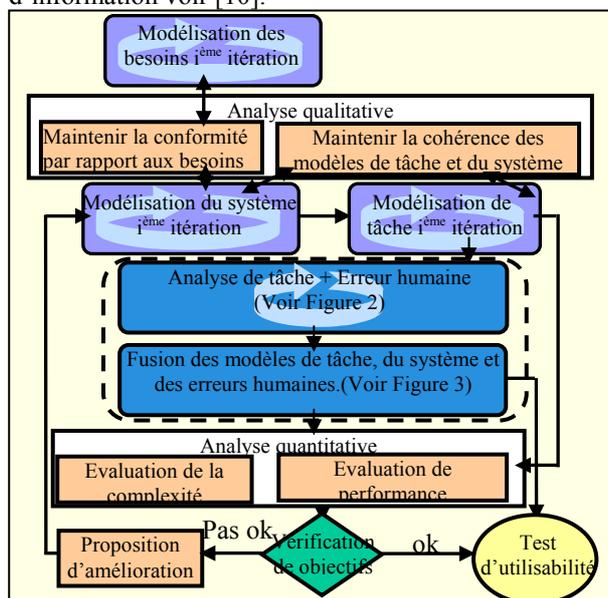


Figure 1: Cohérence diagramme

Dans les sections suivantes, nous fournissons plus de détails sur le diagramme ci-dessus et de ses modifications concernant la modélisation des tâches, l'analyse d'erreur, les « task patterns » et la modélisation du système. Ensuite, nous examinons la fusion de l'information mentionnée auparavant.

LA MODELISATION DES TACHES

L'analyse des tâches et leur modélisation sont largement acceptées en tant qu'élément central pour les approches de conception centrée utilisateur. Le modèle de tâche et la représentation des tâches utilisateur impliquent une forme d'interaction avec un système influencé par son contexte. Un résumé des techniques reconnues d'analyse de tâche est présentée dans le Tableau 1. Les utilisateurs accomplissent des tâches et ils sont structurés en groupe d'activités [15] de façon à obtenir des buts de plus haut niveau.

Acronym	Full Name
HTA	Hierarchical Task Analysis [1]
TKS	Task Knowledge Structure [8]
MAD	Méthode Analytique de Description de tâches [18]
UAN	User Action Notation [5]
GTA	GroupWare Task Analysis [19]
CTT	ConcurTaskTrees [14]
GOMS	Goals, Operators, Methods and Selection rules [3]

Tableau 1: Techniques d'analyse de tâche

Des tâches peuvent être décomposées davantage, correspondant au sous buts de bas niveau. De cette notion de décomposition résulte naturellement un structure en forme d'arbre et ainsi en une représentation hiérarchique du modèle. Cependant jusqu'à présent, la modélisation des tâches ne permet pas la description, la représentation et l'analyse des éventualités inattendues qui pourraient arriver, comprenant l'erreur humaine. Comme le modèle de tâche influence la conception du system, il est important de comprendre comment gérer et maîtriser les événements erronés possibles.

CTT

Au sein de ce travail, nous considérons que ConcurrentTaskTree de F. Paternò [14] est la notation la plus appropriée grâce à son apparence graphique et l'outil associé. CTT est une notation graphique utilisée pour spécifier les modèles de tâche pour des applications coopératives par une structure hiérarchique en indiquant les relations temporelles par des opérateurs. Les modèles de tâche peuvent ensuite être simulés pour étudier les différents chemins d'interaction possibles. La notation est basée sur quatre types de tâche (les tâches abstraites, les tâches utilisateur, les tâches système et les tâches d'interaction), ainsi que sur plusieurs opérateurs temporels (voir [14] pour des détails supplémentaires). Cependant, CTT ne permet pas de prendre en compte le contexte et les conditions d'environnement, les circonstances qui peuvent affecter le procédé, les artefacts manipulés pendant la tâche, la charge de travail cognitive et l'état actuel des utilisateurs (i.e. stress, fatigue), le détail du type d'interaction de bas niveau ou le changement de point d'intérêt de l'utilisateur.

TASK PATTERNS

Les « task patterns » pour la conception des systèmes interactifs est un concept relativement nouveau dont le but est de résoudre les problèmes de conception en utilisant les « task patterns » identifiés auparavant. Introduits à l'origine par Paternò [4] et [14] en tant que structures réutilisables pour les modèles de tâche, les « patterns » sont décrits comme des fragment de tâches structurés et hiérarchiques qui peuvent être réutilisés pour construire le modèle de tâche. Nous introduisons ici l'utilisation des « task patterns » comme moyens d'incorporer les comportements erronés dans les modèles de tâche standard et comme méthode pour expliciter les erreurs.

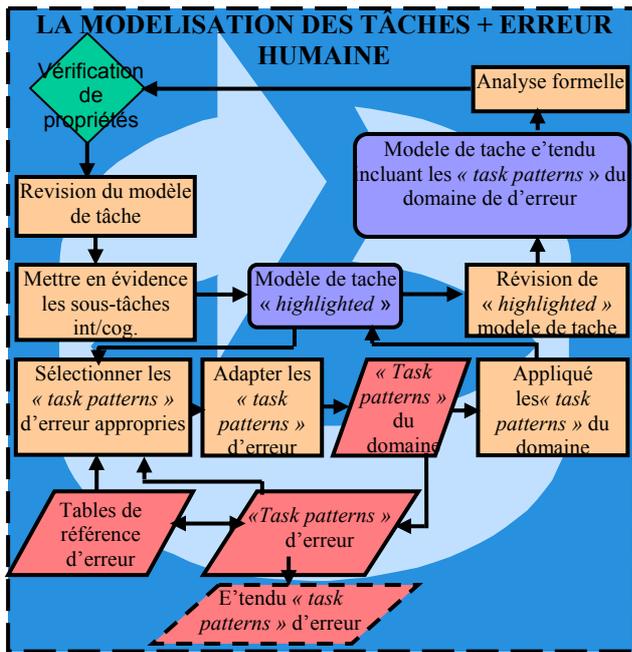


Figure 2: Modélisation Des Tâches + Erreur Humaine

ERREUR HUMAINE

80% de tous les accidents d'aviation sont attribués à « l'erreur humaine » [7]. Les systèmes interactifs, et en particulier les systèmes critiques, ont besoin d'être produits dans l'éventualité d'une erreur humaine, pour éviter les catastrophes. Ceci signifie que cela doit être pris en compte au début du processus de conception ainsi que durant la phase de test. Bien que le terme « erreur humaine » apparaisse très discutable, les théories sur les erreurs humaines telles que Rasmussen's [16] SRK, Hollnagel's [6] Phénotypes et Génotypes et Norman's [11] la classification des « slips » peuvent être considérées comme largement acceptables. En utilisant les classifications mentionnées ci-dessus, basées sur la théorie SRK [16], nous avons produit des tables de référence de l'erreur humaine pour l'analyse de l'erreur humaine potentielle dans les modèles de tâche. L'avantage de produire de tels tableaux de référence permet l'identification exacte de type d'erreur très précis durant l'analyse du comportement humain associé à des tâches particulières du modèle de tâche.

LA MODELISATION DES TÂCHES + ERREUR HUMAINE: EXTENSION DU PROCESSUS

La Figure 2 montre en détails le processus interne de la phase optionnelle de « Modélisation des tâches + erreur humaine » ajoutée au diagramme de la Figure 1. En se basant sur le processus existant, le processus étendu commence avec une analyse formelle, la vérification des propriétés et une révision du modèle de tâches. A ce moment, les sous tâches interactives et cognitives sont identifiées à partir du modèle de tâches (puisque ce sont les étapes durant lesquelles les erreurs humaines peuvent être faites) créant un modèle de tâche « highlighted ».

En fonction des nœuds retenus, les « tasks patterns » d'erreur explicite possible peuvent être choisis. En employant les tables de référence de l'erreur humaine (et les « tasks patterns » existants), l'interaction choisie ou les sous-tâches cognitives du modèle de tâche original sont analysées. Par élimination, ces erreurs applicables peuvent être analysées davantage, afin de déterminer les effets de l'erreur sur la tâche. Par ex, si un utilisateur, face un distributeur de billets, entre un code différent de celui attendu, le système va juger que le code est incorrect. Une fois le « task pattern » adapté au domaine, celui-ci vient enrichir la base des « task patterns » déjà existants. Les « task patterns » d'erreur explicite adaptés au modèle de tâche peuvent être connectés aux endroits mis en avant du modèle de tâche original. Ceci permet une amélioration plus importante du modèle résultant en un modèle de tâches étendu qui inclut les événements à la fois standard et erronés en employant les « tasks patterns ».

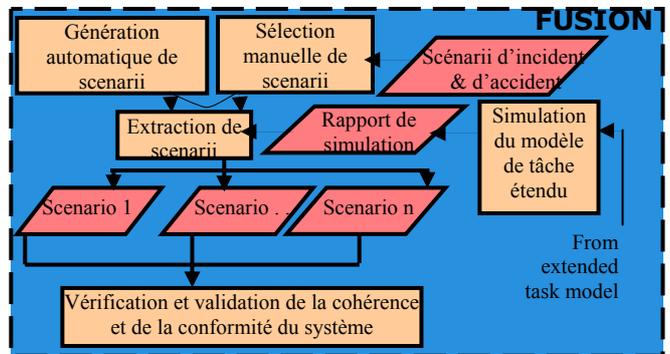


Figure 3: La phase de fusion

LA MODELISATION DU SYSTEME

Pour la modélisation du système, nous utilisons l'environnement « PetShop » [10] et la technique de description formelle ICO [2]. Les Objets Coopératifs Interactifs sont un formalisme orienté objet consacré à la modélisation des systèmes interactifs. Ce formalisme est dédié à la construction des applications réparties hautement interactives. Il doit être utilisé par des programmeurs spécialisés, qualifiés en techniques de description formelles, en approches orientées objet et systèmes interactifs répartis [13].

LA FUSION DU MODELE DE TACHE, DE L'ERREUR HUMAINE, DES « TASKS PATTERNS » ET DU MODELE DE SYSTEME

Une fois que le modèle de tâche étendu a été créé, celui-ci peut être simulé pour obtenir des scénarii au sein de l'environnement support de CTT (CTTE). Les scénarii extraits peuvent ensuite être appliqués et testés dans l'environnement PetShop sur le modèle du système décrit en réseau de Petri. Si à cette étape du processus il existe déjà un système, celui-ci devra être pris en compte. Ce processus assure que lors de la modélisation du système, ou future système, le modèle de tâches peut être testé sur le modèle du système produit en incluant les erreurs humaines déjà identifiées. Ceci limite la

découverte de problèmes durant les phases de test ultérieures au cours du cycle de vie du système. Si nécessaire, le modèle du système peut être re-conçu pour supporter les événements erronés identifiés. Une étude de cas détail décrivant le procédé de l'analyse de l'erreur humaine peut être trouvée dans [12].

TRAVAIL EN COURS : VERS DES SYSTEMES CRITIQUES TOLERANTS A L'ERREUR

Nous étudions actuellement les méthodes pour combiner l'erreur humaine et l'analyse de tâches. Ceci implique d'envisager soit d'englober les erreurs au sein du modèle de tâches (comme exposé dans ce papier), soit de les analyser en dehors, en les séparant du modèle de tâches et en les appliquant indépendamment sur le modèle du système. Le processus modifié peut devenir plus utile distinguant les divers groupes de flèches. L'idéal serait que la simulation et l'extraction de tous les scénarii éventuels comprenant toutes les erreurs pour la validation systématique du système soient rendues automatique ; c'est pourquoi nous dirigeons nos travaux dans cette direction. L'analyse de l'erreur ne devrait pas être retrainte à des erreurs basées sur la compétence mais elle devrait incorporer les erreurs basées sur la connaissance et les règles.

BIBLIOGRAPHIE

- Annett, J. and Duncan, K. *Task Analysis and Training Design, Occupational Psychology*. 41, 1967, pp.211-227.
- Bastide R. and Palanque P. A Visual and Formal Glue between Application and Interaction. *International Journal of Visual Language and Computing*, Academic Press Vol. 10, No. 5, pp. 481-507. 1999.
- Baumeister, L.K., John, B.E., Byrne, M.D.A *Comparison of Tools for Building GOMS Models Tools for Design*. In: Proc. of ACM Conf. on Human Factors in Computing Systems CHI'2000, ACM Press, New York, 502-509 2000.
- Breedvelt, I., Paternò, F., Sereriins, C. *Reusable Structures in Task Models, Proceedings Design, Specification, Verification of Interactive Systems*. Springer Verlag, pp.251-265 1997.
- Hix, D. and Hartson, H. R. *Developing User Interfaces*. 1993
- Hollnagel, E., *The Phenotype of Erroneous Actions: Implications for HCI Design*. In: Weir, G.R.S. and Alty, J.L., (Eds.), *Human-Computer Interaction and Complex Systems*, Academic Press. 1991
- Johnson C.W. *Failure in Safety-Critical Systems: A Handbook of Accident and Incident Reporting*. University of Glasgow Press, Glasgow, Scotland, October 2003. ISBN 0-85261-784-4. 2003
- Johnson, P., and Johnson, H. *Knowledge Analysis of Task: Task Analysis and Specification for Human-Computer Systems*. In Downton, A. (ed.): *Engineering the Human Computer Interface*. McGraw-Hill, Maidenhead 1989 119-144
- Navarre, D., Palanque, P., Bastide, R. *Notations en Interaction Homme-Machine pour une Modélisation Synergique des Tâches et du Système*. Chapitre du livre *Ingénierie cognitive: IHM et cognition (Traité des Sciences Cognitives)*. Editeur Guy Boy. Hermès éditions, 2003. ISBN 2-7462-0571-8
- Navarre, D, Palanque, P, Bastide, R, & Sy, O A *Model-Based Tool for Interactive Prototyping of Highly Interactive Applications*. 12th IEEE, International Workshop on Rapid System Prototyping ; Monterey (USA). IEEE ; 2001.
- Norman D.A. *The design of everyday things*. New York: Currency-Doubleday, 1988.
- Palanque, P and Basnyat. S. *Task Patterns for Taking into account in an efficient and systematic way both standard and erroneous user behaviours*. HESSD 2004. 6th International Working Conference on Human Error, Safety and System Development, 22-27 August 2004, Toulouse, France (within the IFIP World Computing Congress WCC 04).
- Palanque, P and Bastide, R. UAHCI 2003. *User-Centered Point of View to End-User Development. Universal Access for Human-Computer Interaction* Heracklion, Crete, June 2003.
- Paternò, F. *Model Based Design and Evaluation of Interactive Applications*. Springer Verlag, Berlin 1999
- Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S. & Carey, T. (1994) *Human-Computer Interaction*. Wokingham, UK: Addison-Wesley.
- Rasmussen, J. *Skills, rules, knowledge: Signals, signs, and symbols and other distinctions in human performance models*. *IEEE Transactions on Systems, Man, and Cybernetics*, 13(3):257-267 1983
- Reason, J. *Human Error*. Cambridge University Press, 1990.
- Scapin, D., Pierret-Golbreich, C. *Towards a Method for Task Description: MAD*. In Berlinguet, L., Berthelette, D. (eds.): *Proc. of Conf. Work with Display Units WWU'89*, Elsevier Science Publishers, Amsterdam (189) 27-34
- van der Veer, G.C., ad van der Lenting, B.F., Bergevoet, B.A.J.: *GTA: Groupware Task Analysis - Modeling Complexity*. *Acta Psychologica* 91 1996 297-322.

Cohérence entres modèles de tâche et du système avec analyse d'erreurs

RESUME

Ce travail fournit une méthode permettant de maintenir la cohérence entre les modèles de tâche et du système. Celle-ci permet de prendre en compte les erreurs humaines de façon à réduire le nombre d'événements erronés dans un système interactif critique et tolérant aux erreurs.

1 MODELE DU SYSTEME

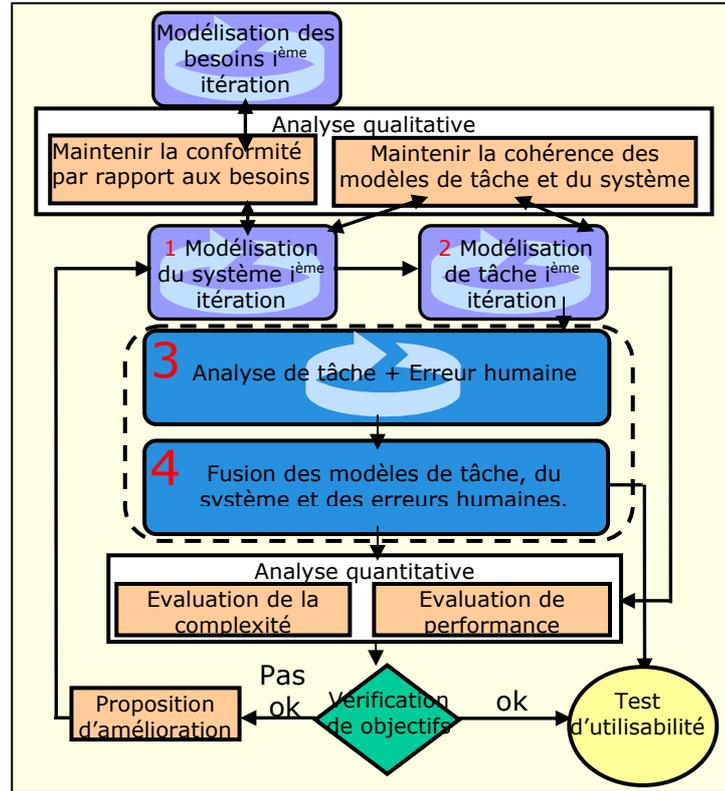
Le processus itératif consistant à modéliser un système est mis en œuvre grâce à l'environnement Petshop. [Navarre et al, 2001] et au formalisme ICO [Bastide & Palanque, 1999].

3 MODELE DE TACHE + ERREUR HUMAINE

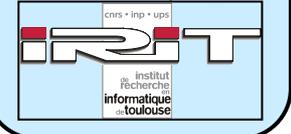
Cette phase, optionnelle et itérative, est basée sur des diagrammes de processus préexistants [Navarre et al., 2001]. Elle consiste en une analyse formelle du modèle de tâche original, une vérification de ses propriétés et une révision de ce modèle, comme on l'a fait pour les modèles de tâches et de système lors de la phase précédente. C'est à ce moment là que les sous-tâches cognitives et interactives sont identifiées (car c'est le moment où les erreurs humaines peuvent avoir lieu). Un modèle de tâche « highlighted » est ainsi produit.

4 FUSION

Le modèle de tâche étendu peut être simulé et des scénarii peuvent être exécutés dans CTTE. Les scénarii extraits peuvent être testés et exécutés dans Petshop sur le modèle du système. Cela permet de détecter des problèmes que l'on découvre habituellement lors des phases de test suivantes du processus de développements. Si nécessaire, le modèle du système peut être redéfini de manière à prendre en compte des événements non-conformes qui ont été mis en évidence précédemment.

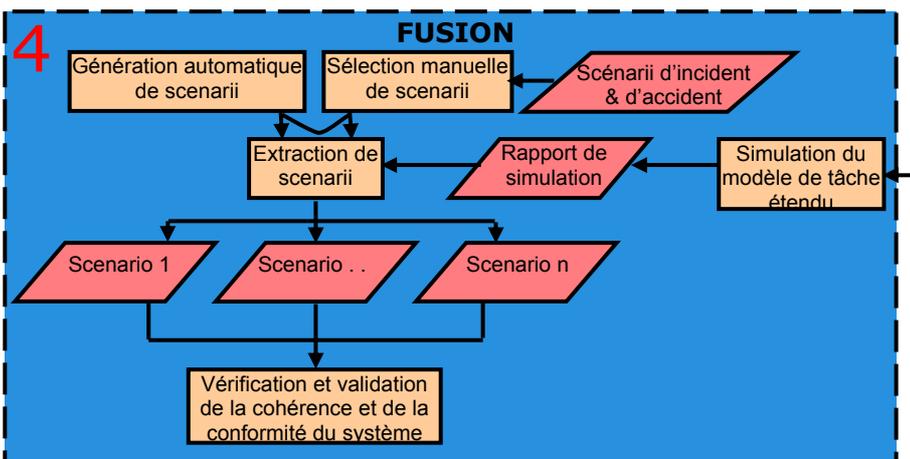
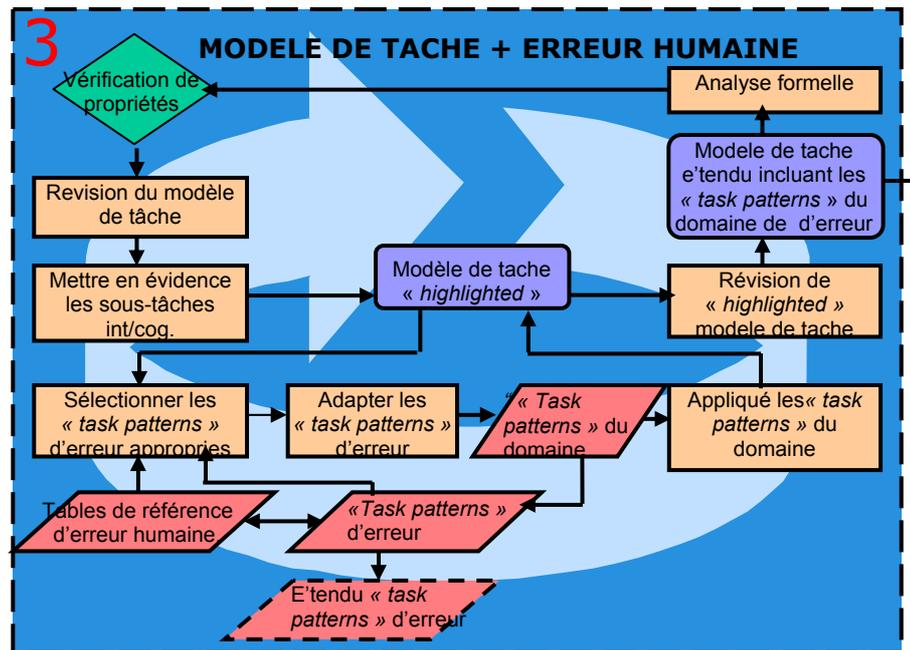


Sandra BASNYAT
 Université Paul Sabatier
 LIHS - IRIT
 118, route de Narbonne,
 31062 Toulouse Cedex 4
 basnyat@irit.fr



2 MODELE DE TACHE

Nous avons choisi CTT pour la modélisation des tâches car c'est une notation graphique qui possède un éditeur. Le modèle de tâche peut être simulé dans le but d'explorer les différents chemins d'interaction possibles.



Pointage Sémantique et Distracteurs, la Dynamique du Pointage à la Rescousse

Renaud Blanch

LRI & INRIA Futurs
Université Paris-Sud XI
91 405, Orsay CEDEX, France
renaud.blanch@lri.fr

RESUME

Le pointage sémantique permet de faciliter le pointage en attribuant indépendamment des tailles visuelle et effective différentes à tous les objets graphiques d'une interface. Ces tailles sont choisies en fonction d'une information jusqu'à maintenant négligée par le système : l'importance pour l'interaction de chaque zone de l'écran. Nous améliorons cette technique en prenant en compte la trajectoire du curseur, qui comporte aussi des informations sous-exploitées, pour atténuer l'impact négatif des cibles potentielles présentes sur la trajectoire d'un pointage : les distracteurs.

MOTS CLES : distracteur, pointage, pointage sémantique, ratio *control/display*.

INTRODUCTION

La tâche de pointage est l'interaction élémentaire des interfaces graphiques. L'augmentation de la complexité ou, au mieux, du nombre des fonctionnalités des logiciels, se traduit par un nombre sans cesse croissant de cibles potentielles alors que la taille des écrans n'évolue guère [2]. On assiste donc à une prolifération de cibles de plus en plus nombreuses et de plus en plus petites. La loi de Fitts [6] nous rappelle que plus une cible est loin du curseur et petite, plus le temps pour l'atteindre est long.

C'est dans ce contexte que des travaux récents ont tenté, par différentes approches, de faciliter les tâches de pointage. Nous donnons ici un aperçu de ces techniques, et en particulier du pointage sémantique que nous avons développé dans un travail antérieur [3]. Nous montrons ensuite que cette technique peut être enrichie en utilisant des informations extraites de la trajectoire du curseur, ce qui la rend encore plus efficace.

TRAVAUX ANTERIEURS

La loi de Fitts [6] nous enseigne que le temps pour atteindre une cible croît avec la distance D à cette cible et diminue lorsque sa taille W , elle, augmente. Partant de ce simple constat, de nombreuses méthodes ayant pour objectif de faciliter le pointage ont été proposées. En voici quelques unes.

Faciliter le Pointage en Réduisant D

Réduire la distance des cibles est sans doute la première idée qu'on peut avoir. Cette idée a donné le jour aux menus contextuels qui apparaissent au plus près du curseur, réduisant ainsi la distance des entrées du menu au curseur au minimum. Les *pie menus* de Hopkins [4] améliorent encore cette technique en disposant les éléments du menu en cercle autour du curseur, les mettant ainsi à une distance quasiment nulle de celui-ci (figure 1).



Figure 1 : Les entrées d'un *Pie menu* disposées en cercle autour du curseur.

Le *drag-and-pop* de Baudisch [1] est une technique qui consiste à rapprocher les cibles potentielles d'un *drag-and-drop* vers l'objet déplacé. Ces cibles potentielles sont déterminées par la direction du déplacement et la compatibilité des objets (figure 2).

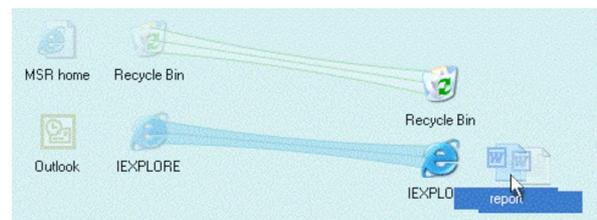


Figure 2 : Un document est déplacé et le *drag-and-pop* approche les cibles susceptibles de lui convenir.

Enfin, le pointage d'objets [7] est une technique récente qui pousse ce genre d'idée à l'extrême en "supprimant" artificiellement l'espace vide entre les objets du bureau, réduisant par là-même D le plus possible. Cette suppression artificielle est obtenue en faisant sauter le curseur vers l'objet le plus proche dans la direction de son mouvement dès qu'il se retrouve dans le vide.

Faciliter le Pointage en Augmentant W

L'autre variable sur laquelle il est possible de jouer est la taille de la cible. Kabbash et Buxton montrent qu'en utilisant pour la sélection une zone (*area cursor*) plutôt qu'un point (pointe du curseur), cela revient pour les cibles qui sont plus petites que la zone de sélection choisie, à leur donner une taille égale à cette zone [8]. Cette technique permet donc de faciliter la sélection de petites cibles en leur donnant artificiellement une taille supérieure, mais pose des problèmes lorsque plusieurs cibles entrent simultanément dans la zone de sélection.

Une autre approche consiste à dilater les cibles lorsque le curseur s'en rapproche par un effet de *fish-eye* ou en leur adjoignant des poignées ou "*bubble*" [9, 5]. Une telle technique est utilisée dans le Dock de Mac OS X (figure 3).



Figure 3 : Les cibles du Dock de Mac OS X se dilatent à l'approche du curseur.

Des comparaisons avec d'autres techniques montrent que la dilatation des cibles est efficace même si elle est mise en jeu tard dans le mouvement, comme l'ont montré McGuffin et Balakrishnan [9], et que l'on en tire parti même si elle n'est pas anticipée, comme l'ont montré Zhai et al. [10]. Cependant, ces techniques posent aussi problème lorsque plusieurs cibles sont proches. Il faut alors en effet les déplacer pour faire de la place à la cible dilatée. Dans le cas du Dock, par exemple, Zhai et Beaudouin-Lafon ont montré que si l'expansion permet de mieux repérer les cibles, elle n'offre par contre aucune aide au pointage puisque le mouvement des cibles généré par le déplacement du centre de la dilatation annule le bénéfice de celle-ci [10].

Une autre voie explorée récemment ne s'attaque plus de front à D et à W , mais cherche à aboutir aux mêmes effets sans modifier le placement et la taille des objets à l'écran. Il s'agit de l'adaptation du ratio *control/display*.

L'ADAPTATION DU RATIO CONTROL/DISPLAY

L'idée est de faciliter le pointage en modifiant le ratio *control/display* au cours du mouvement. Le ratio *control/display* est la constante qui lie les déplacements du curseur à ceux de la souris. Le même déplacement du curseur est provoqué par un mouvement de la souris de plus en plus grande amplitude à mesure que le ratio augmente. En adaptant cette constante à divers paramètres au cours du pointage, on peut artificiellement modifier, pour le mouvement à accomplir avec la souris et non pour le curseur à l'écran, la taille et la distance des cibles.

L'“Accélération” de la Souris

En fait, une telle adaptation existe depuis longtemps dans ce qui est nommé improprement l'“accélération” de la souris. Cette technique change le ratio en fonction de la vitesse de la souris, faisant en sorte que lorsque la souris se déplace vite, la distance couverte par le curseur pour un déplacement donné de la souris soit plus grande que lorsque celle-ci se déplace lentement. Cela permet par exemple d'aller d'un bout à l'autre de l'écran sans avoir besoin de beaucoup de place sur la table, tout en conservant une précision élevée pour des mouvements de faible amplitude réalisés à basse vitesse.

Le Pointage Sémantique

Nous avons montré par ailleurs que le ratio *control/display* peut s'interpréter comme un facteur d'échelle entre l'espace moteur (la table sur laquelle évolue la souris) et l'espace visuel (l'écran où sont affichés les objets de l'interface) [3]. En adaptant le ratio en fonction, cette fois, de la position du curseur et non de sa vitesse, on peut alors modifier les tailles relatives des différents objets dans l'espace moteur tout en préservant leur disposition à l'écran.

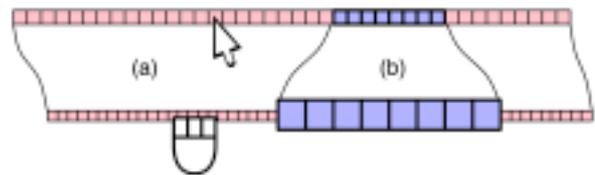


Figure 4 : Le ratio *control/display* comme facteur d'échelle entre l'écran (ligne de pixels en haut) et l'espace de la souris (“images” des pixels sur la table après un changement d'échelle en bas).

La figure 4 illustre cette idée mise en œuvre par le pointage sémantique. La ligne du haut représente une rangée de pixels à l'écran, et celle du bas, la position qu'ils occupent pour la souris, sur la table, après avoir subi un changement d'échelle. La zone (a) représente une partie non intéressante pour l'interaction (fond de l'écran par exemple). Elle est contractée pour être traversée rapidement. La zone (b) représente une cible importante. Elle est donc dilatée pour permettre à la souris de s'arrêter facilement dedans.

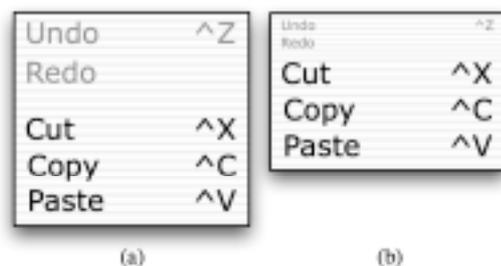


Figure 5 : Le menu redesigné.

(a) version à l'écran, et (b) version dans l'espace moteur qui permet de “sauter” les entrées inactives.

On peut ainsi contracter les zones de vide (ou celles qui ne sont simplement pas importantes pour l'interaction comme c'est le cas des entrées inactives du menu de la figure 5), on peut aussi réduire à l'écran des objets dont l'encombrement ne sert qu'à faciliter la manipulation, tout en préservant leur taille initiale pour une interaction aussi aisée (la figure 6 présente une telle évolution de la barre de défilement classique).

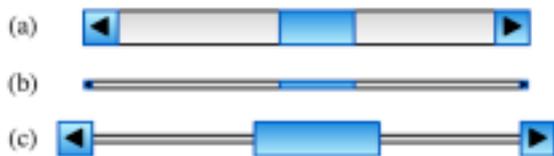


Figure 6 : La barre de défilement redesignée.

(a) version originale, (b) version à l'écran qui réduit l'encombrement en présentant la même information, et (c) version dans l'espace moteur qui préserve l'utilisabilité du design original.

Pour mieux appréhender le pointage sémantique, on en trouvera à l'adresse suivante une démonstration interactive en ligne sous la forme d'une *applet* Java :

<http://insitu.lri.fr/~blanch/project/SemanticPointing/demo/>

Bénéfices du Pointage Sémantique

L'intérêt du pointage sémantique tient au fait que l'on peut, en choisissant pour un objet une taille à l'écran et un ratio, lui attribuer en fait deux tailles distinctes et indépendantes : sa taille dans l'espace visuel et sa taille dans l'espace moteur. Cette dichotomie permet de séparer les contraintes qui pèsent sur le choix de la taille des objets et de les faire peser sur l'un ou l'autre des aspects. La taille à l'écran peut ainsi n'être choisie qu'en fonction de l'information que doit présenter l'objet à l'utilisateur. Par contre, la taille dans l'espace moteur n'est régie que par la contrainte de l'utilisabilité, et on peut donc rendre gros dans cet espace les objets interactifs peu porteurs d'information comme les poignées de manipulation ou les boutons. L'exemple de la barre de défilement (figure 6) illustre ce principe : la barre donne peu d'information, on lui attribue peu de pixels (figure 6b). Par contre, les flèches aux extrémités doivent être faciles à atteindre, elles sont donc grosses dans l'espace moteur (figure 6c).

Cette séparation en deux tailles n'a d'intérêt que parce que le temps de pointage donné par la loi de Fitts pour un objet est déterminé par sa taille et sa distance, mais celles-ci, comme nous l'avons montré par une expérimentation contrôlée dans [3], sont comprises dans l'espace *moteur*. On peut donc, étant donnée une tâche spécifiée visuellement, la rendre plus facile à réaliser par simple manipulation du ratio *control/display*, et ceci sans modifier le placement à l'écran des objets.

PRISE EN COMPTE DES DISTRACTEURS

Un problème inhérent à la plupart des techniques présentées jusqu'à présent, est que le système ne peut prédire la cible précise d'un mouvement. Il facilite également toutes les cibles considérées comme potentielles du mouvement. Les objets considérés comme cibles possibles par le système qui diffèrent de la cible réelle, celle que l'utilisateur a en tête, sont autant de distracteurs pour la réalisation de la tâche. Il s'agit, dans le cas d'un *pie menu* par exemple, de toutes les entrées du menu exceptée celle qui sera choisie. Dans le cas du pointage d'objet, comme dans celui du pointage sémantique, il s'agit des objets qui sont croisés par la trajectoire du curseur, et qui le ralentissent au passage. Dans le pointage normal, il s'agit de chaque pixel qui sépare le curseur de la cible puisque tous les pixels sont traités de la même manière par le système.

Un Ratio *Control/Display* Optimisé

Le système de pointage idéal ressemble à l'interface souvent présente dans les films de science-fiction : le bouton unique qui effectue précisément l'action que l'utilisateur a en tête. Si le système ne peut prédire la cible que l'utilisateur veut atteindre, certaines informations peuvent être exploitées simplement et de manière complètement déterministe pour régler le compromis vitesse de déplacement / précision des mouvements du curseur que représente aussi le ratio *control/display*, et ce de manière à faciliter le pointage.

Prise en Compte de la Position. La position du curseur à l'écran donne une information complètement négligée par la plupart des systèmes : le curseur est-il dans une cible ou dans une zone inintéressante pour l'interaction ? Cela revient à considérer le pointage d'une cible comme la sélection d'un des pixels qui la compose parmi tous ceux de l'écran et non comme la simple sélection d'une cible parmi un ensemble de cibles. Les pointages sémantique et d'objets [3, 7] s'appliquent justement à réintroduire cette information pour optimiser le ratio *control/display*.

Prise en compte de la vitesse. La vitesse de déplacement du curseur fournit elle aussi une information intéressante facile à exploiter. En effet, lorsque le curseur se déplace rapidement, il y a peu de chance pour que l'utilisateur ait l'intention de s'arrêter. En fait, plus il déplace la souris rapidement, plus il recherche la vitesse, et moins il est préoccupé par la précision. A contrario, pour sélectionner précisément une cible, les déplacements se feront à faible vitesse. C'est exactement de cette manière qu'est adapté le ratio *control/display* par l'"accélération" de la souris sur la plupart des systèmes existants.

Ces deux optimisations (position et vitesse) ne permettent cependant pas de lever une ambiguïté résiduelle. En

effet, les données de position et de vitesse ne discriminent pas un début de mouvement de la fin d'un pointage. Dans les deux cas, le curseur est en général à proximité d'une cible (but du mouvement précédent ou de celui en cours), sa vitesse est faible (le curseur est en train de démarrer ou en train de s'arrêter). Pour déterminer dans quel contexte le mouvement se situe, il faut donc prendre en compte une variable qui ne présente pas de similitude entre le début et la fin d'un mouvement.

Prise en compte de l'accélération. L'accélération du curseur est positive lorsque le début d'un mouvement est entamé, alors qu'elle devient négative lorsque l'on freine pour s'arrêter dans une cible. C'est donc le candidat idéal qui permet de discriminer entre le début d'un mouvement et sa fin.

En résumé, la prise en compte de la position du curseur permet de pointer des cibles et non plus de simples pixels sur l'écran, et ce faisant, de faciliter arbitrairement le pointage (figure 7, à gauche, une trajectoire typique sans aide au pointage, à droite, l'effet du pointage sémantique qui réduit significativement le temps de pointage pour la même tâche).

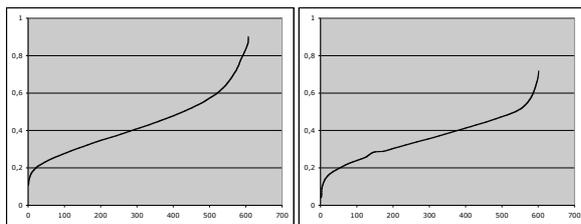


Figure 7 : Trajectoires typiques d'un pointage. (abscisse : position du curseur, ordonnée : temps, cible en 600) à gauche, sans le pointage sémantique, à droite, avec la cible ayant une échelle de 4.

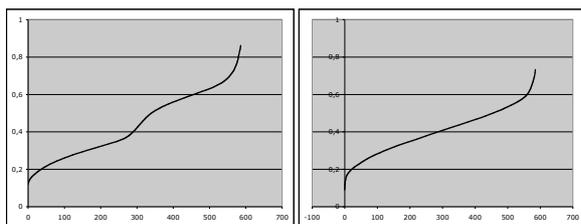


Figure 8 : Pointage sémantique avec un distracteur en 300. à gauche, le pointage sémantique classique, à droite, en tenant compte de la vitesse et de l'accélération.

Par contre, un problème se pose alors : celui des distracteurs qui freinent le curseur le long de sa trajectoire et déstabilisent l'utilisateur par un comportement non souhaité du curseur. Le bénéfice du pointage sémantique est alors minimisé (figure 8, à gauche on note la discontinuité dans la trajectoire causée par un distracteur à mi-chemin). En utilisant l'information de vitesse et d'accélération pour moduler le pointage sémantique, on arrive à atténuer l'effet des distracteurs et à revenir au

gain initial sans distracteur (figure 8, à droite, le distracteur à mi-course n'a pas d'influence sur la trajectoire et on retrouve le gain observé sans le distracteur).

CONCLUSION

On a montré que l'utilisation d'informations issues de la trajectoire du curseur et de sa dynamique permettait d'aider le pointage. L'utilisation de la position a fait l'objet d'une étude statistique approfondie [3]. Celle de la vitesse et de l'accélération est prometteuse comme le montrent les données présentées ici (issue d'un prototype visant à valider l'idée, figure 7 et 8) mais elle mérite une étude contrôlée pour être pleinement validée.

BIBLIOGRAPHIE

1. P. Baudisch, E. Cutrell, D. Robbins, M. Czerwinski, P. Tandler, B. Bederson, and A. Zierlinger. Drag-and-pop and drag-and-pick: techniques for accessing remote screen content on touch- and pen-operated systems. In *Proc. Interact 2003*, pages 57–64, 2003.
2. M. Beaudouin-Lafon. Instrumental interaction: an interaction model for designing post-WIMP user interfaces. In *Proc. CHI 2000*, pages 446–443. ACM Press, 2000.
3. R. Blanch, Y. Guiard, and M. Beaudouin-Lafon. Semantic pointing: improving target acquisition with control-display ratio adaptation. In *Proc. CHI 2004*, pages 519–526. ACM Press, 2004.
4. J. Callahan, D. Hopkins, M. Weiser, and B. Shneiderman. An empirical comparison of pie vs. linear menus. In *Proc. CHI 1988*, pages 95–100. ACM Press, 1988.
5. A. Cockburn and A. Firth. Improving the acquisition of small targets. In *Proc. HCI 2003*, pages 181–196, 2003.
6. P. M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, 47:381–391, 1954.
7. Y. Guiard, R. Blanch, and M. Beaudouin-Lafon. Object pointing: a complement to bitmap pointing in GUIs. In *Proc. GI 2004*, 2004.
8. P. Kabbash, and W. Buxton. The “Prince” technique: Fitts’law and selection using area cursors. In *Proc. CHI 1995*, pages 273–279. ACM Press, 1995.
9. M. McGuffin and R. Balakrishnan. Acquisition of expanding targets. In *Proc. CHI 2002*, pages 57–64. ACM Press, 2002.
10. S. Zhai, S. Conversy, M. Beaudouin-Lafon, and Y. Guiard. Human on-line response to target expansion. In *Proc. CHI 2003*, pages 177–184. ACM Press, 2003.

Pointage sémantique et distracteurs, la dynamique du pointage à la rescousse

Renaud Blanch

renaud.blanch@lri.fr
LRI & INRIA Futurs - projet InSitu
Université Paris-Sud XI

Le pointage sémantique

le but : **Faciliter la sélection de cibles** par un curseur à l'écran.

l'idée : **Adapter le ratio *control-display*** [3], pour faire en sorte que certaines portions de l'écran soient dilatées ou contractées dans l'espace moteur.

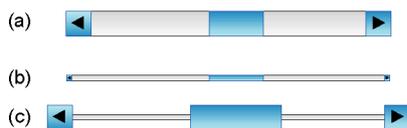
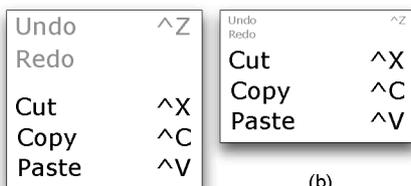
En fonction de la **sémantique du pixel** le curseur "accélère" dans les espaces vides et "freine" près des cibles potentielles.

l'hypothèse : Le temps de mouvement pour la réalisation d'un pointage suit bien une loi de Fitts [2] mais sa **difficulté** est celle de la tâche dans l'**espace moteur** [1].

l'application : **Choisir indépendamment deux tailles** pour chaque objet de l'interface :

- dans l'**espace visuel** adaptée à l'information présentée à l'utilisateur,
- dans l'**espace moteur** adaptée à l'importance de l'objet pour la manipulation.

Exemples : menu et barre de défilement

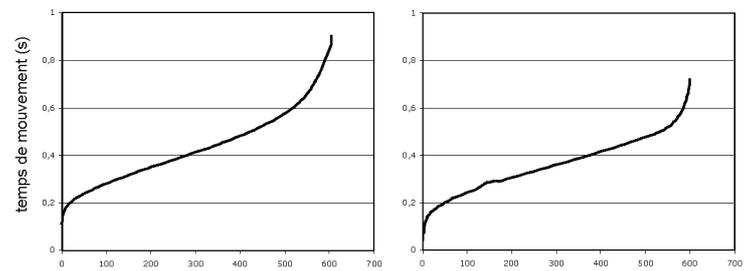


(a) design actuel : occupe toute une bande de pixels à l'écran mais n'apporte qu'une position dans le document et la proportion visible.

(b) redesign - écran : moins de pixels mais autant d'information.

(c) redesign - moteur : manipulation aussi aisée qu'avec la

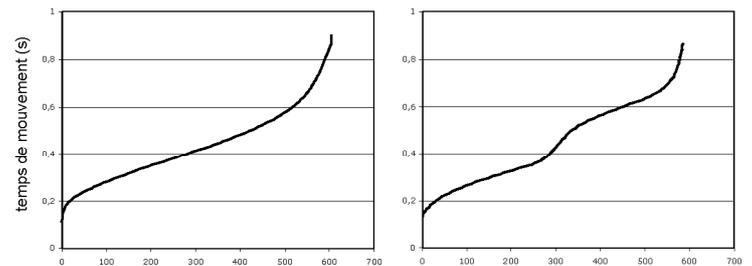
Effet du pointage sémantique



Le graphique de gauche représente une trajectoire typique pour la sélection d'une cible située en 600. Le graphique de droite représente la même tâche effectuée en présence de l'adaptation du ratio *control-display* par le **pointage sémantique**.

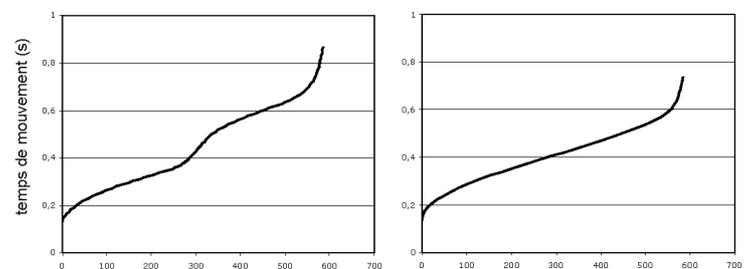
On observe une réduction significative du temps de pointage.

Effet d'un distracteur



En présence d'un **distracteur** à mi-chemin, alors que le pointage normal (à gauche) ne se dégrade pas, le pointage sémantique (à droite) est perturbé, et on perd quasiment tout le bénéfice potentiel de la technique.

Prise en compte de la dynamique du pointage



En considérant que quand la **vitesse** est supérieure à un seuil (zone intermédiaire du mouvement) et quand l'**accélération** est positive (début du mouvement), il n'y a pas lieu de ralentir, on restaure la performance initiale du pointage sémantique, même en présence d'un distracteur (graphique de droite).

Références

1. R. Blanch, Y. Guiard, M. Beaudouin-Lafon, Semantic Pointing: Improving Target Acquisition with Control-Display Adaptation, In *Proc. CHI 2004*, pages 519-526. ACM Press, 2004.
2. P. M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, 47:381-391, 1954.
3. A. Worden, N. Walker, K. Bharat, and S. Hidson. Making computers easier for older adults to use: area cursors and sticky icons. In *Proc. CHI 1997*, pages 266-271. ACM Press, 1997.

Interfaces stylo pour la saisie d'écriture manuscrite sur systèmes mobiles de petite taille

François Bouteruche

IRISA – INSA de Rennes
Campus Universitaire de Beaulieu
35042, Rennes Cedex, France
francois.bouteruche@irisa.fr

Éric Anquetil

IRISA – INSA de Rennes
Campus Universitaire de Beaulieu
35042, Rennes Cedex, France
eric.anquetil@irisa.fr

RESUME

Dans cet article, nous présentons les principes de conception d'interfaces de saisie d'écriture manuscrite pour périphériques mobiles de petite taille que nous avons établis à partir de travaux en psychologie ergonomique et de notre retour d'expérience suite à l'embarquement de notre système de reconnaissance de caractères manuscrits isolés RESIFCar sur des téléphones mobiles de nouvelle génération. Ces téléphones sont aujourd'hui commercialisés en Europe. Nous décrivons ensuite deux interfaces de saisie que nous avons conçues pour illustrer ces principes. Enfin, nous abordons l'évaluation de la qualité ergonomique de ces interfaces que nous effectuons en collaboration avec le Centre de Recherche en Psychologie, Cognition et Communication de l'Université de Rennes 2.

MOTS CLES : Système de reconnaissance d'écriture manuscrite, Interface homme machine stylo, téléphone mobile, ergonomie.

ABSTRACT

In this paper, we present design principles of handwriting input interfaces for small-size mobile devices. We have made up these principles from researches in ergonomic psychology and from our experience coming from the integration in smartphones of our isolated handwritten characters recognition system (RESIFCar). These smartphones are today commercialized in Europe. Then, we describe two input interfaces that we have developed to illustrate this principles. Finally, we discuss about the evaluation of the ergonomic quality of these interfaces. We lead this evaluation in collaboration with the Research Center in Psychology, Cognition and Communication of the University of Rennes 2.

CATEGORIES AND SUBJECT DESCRIPTORS: H.5.2 [Information Interfaces and Presentation]: User Interfaces | Ergonomics, Graphical user interfaces, Interaction styles, User-centered design; I.5.4 [Pattern Recognition]: Applications; I.7.1 [Document and Text Processing]: Document and Text Editing.

GENERAL TERMS: Design, Human Factors.

KEYWORDS: Handwriting Recognition System, Pen-based Interface, Smart phone, Ergonomics.

INTRODUCTION

Les interfaces homme machine basées sur une communication homme machine axée sur l'emploi d'un stylo

connaissent une forte expansion. Celui-ci permet d'interagir avec la machine via un écran tactile. Ce type d'interaction est, *a priori*, très intuitive puisqu'elle correspond à un mode de communication très utilisé par l'homme (prise de notes, croquis, *etc.*).

Deux éléments principaux encouragent le développement des interfaces stylo. Le premier est l'informatique nomade qui met en valeur la transportabilité des machines (tablette PC, assistant numérique personnel, téléphone mobile de nouvelle génération, *etc.*). Afin de diminuer l'encombrement et de s'adapter aux situations de mobilité, ces systèmes ont abandonné le traditionnel couple clavier – souris pour une interaction stylo. Le deuxième élément est la nécessité de développer de nouvelles ergonomies de communication homme machine, plus intuitives et mieux adaptées à certaines applications, notamment pour faciliter l'accès de l'outil informatique au plus grand nombre. Par exemple, l'interaction stylo se révèle particulièrement intéressante pour des éditeurs de partitions musicales, de graphes, de formules mathématiques, ou pour la prise de notes manuscrites.

Dans ce contexte, nos travaux concernent la conception de systèmes de reconnaissance d'écriture manuscrite, principalement à destination de périphériques mobiles de petite taille (téléphones mobiles de nouvelle génération et PDA). Ces travaux ont abouti à la conception du système RESIFCar [1][2] et son embarquement sur des téléphones de nouvelle génération (cf. figure 1) en collaboration avec la société PurpleLabs. Ces téléphones sont actuellement commercialisés en Europe.



Figure 1: modèle de téléphone équipé du système RESIFCar
Les contraintes matérielles de ce type de périphériques nous ont fait opter pour un système de reconnaissance de

caractères manuscrits isolés (RESIFCar). En effet, la puissance des processeurs et la taille des mémoires équipant à l'heure actuelle les téléphones mobiles de nouvelle génération ne sont pas suffisantes pour mettre en œuvre une reconnaissance de mots cursifs liés avec des temps de réponse acceptables du point de vue de l'utilisateur.

Dans la suite, nous expliquons l'importance de l'ergonomie des interfaces de saisie associées aux systèmes de reconnaissance d'écriture et présentons les caractéristiques des interfaces existantes, puis nous en dégageons un ensemble de principes de conception. Ensuite nous présentons une interface conçue à partir de ces principes et sa récente évolution. Enfin, nous abordons l'évaluation de la qualité ergonomique de ces interfaces en collaboration avec le Centre de Recherche en Psychologie, Communication et Cognition (CRPCC) de l'Université de Rennes 2.

ETABLISSEMENT DE PRINCIPES DE CONCEPTION D'INTERFACES DE SAISIE STYLO

Les systèmes de reconnaissance d'écriture manuscrite sont généralement évalués en dehors de tout contexte applicatif sur des bases de caractères manuscrits. Cette évaluation est nécessaire pour mesurer les performances du système en lui-même. Cependant, le retour d'expérience associé à l'embarquement du système RESIFCar dans un produit grand public montre que les performances de ce système en termes de taux d'erreur de reconnaissance, bien que proche des seuils de tolérance admis par l'utilisateur [6], ne permettent pas à elles seules l'acceptation d'un tel système par l'utilisateur. Il est nécessaire que l'ergonomie de l'interface de saisie associée à ce système soit intuitive et rendent l'utilisation du système la plus efficace possible. Cette constatation nous a guidé vers l'étude et la conception d'interfaces adaptées à ce type de systèmes.

Interfaces de saisie d'écriture manuscrite existantes

Actuellement, les interfaces de saisie associées à des systèmes de reconnaissance de caractères isolés commercialisés peuvent être divisées en trois groupes. Le premier correspond aux interfaces de type *Graphiti 1*. A chaque caractère ou symbole correspond une forme spécifique à tracer en un seul trait. La saisie s'effectue caractère par caractère, c'est-à-dire qu'une fois le caractère reconnu, celui-ci disparaît de la zone de saisie. Ce type d'interfaces, d'utilisation efficace pour l'utilisateur expert, est peu intuitive puisqu'elle nécessite un apprentissage et ne permet pas une édition et une correction aisée du texte saisi.

Le deuxième groupe correspond aux interfaces de type *Jot*. La saisie s'effectue aussi caractère par caractère avec les mêmes inconvénients. En revanche, leur saisie est plus naturelle, notamment grâce à la possibilité de tracer les caractères en plusieurs traits si ceux-ci sont ef-

fectués dans un intervalle de temps réduit (d'une à deux secondes).

Les interfaces du troisième groupe permettent une saisie de caractères en contexte de mot, c'est-à-dire que les caractères précédemment tracés restent affichés dans la zone de saisie, jusqu'à validation par l'utilisateur. Il est alors possible, selon les interfaces, de corriger et d'éditer ces caractères directement dans la zone de saisie. Nos travaux s'inscrivent dans cette philosophie de saisie de caractères en contexte de mots. Les interfaces de saisie commerciales les plus récentes s'intègrent aussi dans ce groupe. On peut notamment citer le système de reconnaissance de caractères isolés *Decuma OnSpot* de la société Decuma.

Principes de conception d'interfaces de saisie stylo

Pour commencer, il faut rappeler que les périphériques auxquels est destiné le système RESIFCar présentent deux contraintes matérielles. Tout d'abord, ils disposent de faibles ressources. Ensuite, leurs écrans sont de petites tailles. La zone de saisie d'un téléphone mobile de nouvelle génération peut être assimilée au mieux à la taille de la zone d'affichage occupée par les méthodes de saisie sur les assistants numériques personnels, notamment, si le téléphone est utilisé en mode paysage (cf. figure 1).

En étudiant le fonctionnement des interfaces de saisie des deux premiers groupes précédemment cités, on peut dégager plusieurs problèmes dans leur modalité d'interaction, mis en évidence par des travaux réalisés en psychologie ergonomique [3][5]. Un des principaux est le respect de la contiguïté spatiale des sources d'information. En effet, pour ces deux groupes, l'utilisateur doit effectuer des allers et retours visuels entre la zone de saisie et la zone d'application pour vérifier quels sont les caractères déjà saisis et s'ils ont été bien reconnus. Or la mémoire de travail d'un utilisateur dispose de ressources limitées et ces allers et retours visuels provoquent un partage de l'attention néfaste pour l'utilisateur [4]. I.S MacKenzie et R.W. Soukoreff [5] ont montré que le nombre de zones d'attention et la distance entre elles influaient sur les performances d'une interface de saisie (nombre de mots saisis par minute).

Ayant opté pour un système de reconnaissance de caractères isolés, il faut proposer une interface permettant de connaître les caractères qui ont été saisis et de savoir s'ils ont été correctement reconnus sans changer de zone d'attention visuelle. Ainsi, nous nous sommes orientés vers une saisie de caractères en contexte de mots complets, tout en essayant de respecter la contiguïté spatiale entre les zones de saisie et de retour visuel (zone permettant de connaître les caractères reconnus).

Cependant, comme la taille des écrans est réduite, l'utilisateur n'aura pas nécessairement la place de tracer l'intégralité de son mot dans la zone de saisie. Il est donc

nécessaire de prévoir des mécanismes de défilement lui permettant d'écrire un mot dans sa totalité sans avoir à se préoccuper de libérer de la place dans la zone de saisie. De plus, la nécessité de proposer une interaction intuitive, liée à la petite taille de l'écran implique d'utiliser les possibilités offertes par le stylo pour éviter l'utilisation de menus et de boutons, notamment en proposant des gestes graphiques à réaliser à l'aide du stylo auxquels sont associés des commandes d'édition.

En s'appuyant sur toutes ces contraintes de conception et l'expérience acquise par l'embarquement du système, nous avons dégagé un ensemble de principes guidant la conception d'interfaces de saisie stylo :

- proposer une saisie s'approchant le plus possible d'une saisie de mots complets tout en étant dans le cadre de saisie de caractères isolés ;
- proposer un nombre de zones d'attention minimal avec des distances réduites entre ces zones ;
- proposer un retour visuel des caractères reconnus proche de la zone de saisie (contiguïté spatiale) ;
- gérer l'espace libre de la zone de saisie afin de faciliter celle-ci ;
- utiliser les possibilités offertes par l'interaction stylo afin de réduire l'utilisation de menus et de boutons ;
- choisir des gestes d'édition simples, intuitifs et peu nombreux en s'inspirant des situations réelles de prises de notes sur papier.

INTERFACES DE SAISIE PROPOSEES

Dans un précédent article [2], nous avons présenté une interface issue de nos premiers travaux sur ce sujet que nous appelons « micro éditeur d'encre électronique ». Nous en rappelons ici les principales caractéristiques avant de décrire les évolutions récemment proposées.

Micro éditeur d'encre électronique

Saisie de lettres en contexte de mots complets.

L'utilisateur trace les lettres et symboles dans le sens de l'écriture latine (de la gauche vers la droite) avec l'obligation de lever son stylo entre deux caractères. Il peut cependant tracer un caractère en deux traits. Cette possibilité est indispensable pour ne pas contraindre l'écriture de lettres comme le « i », le « t » ou encore le « x ». les tracés restent affichés dans la zone de saisie, hormis ceux non identifiés ou correspondant à une commande d'édition. La segmentation automatique des tracés se base sur le repérage des poser et lever de crayon ainsi que sur la contiguïté spatiale entre les divers tracés.

Retour visuel. Les tracés composant un même caractère sont envoyés au système RESIFCar qui retourne en réponse le caractère reconnu. Les caractères reconnus sont affichés dans une zone de retour visuel située au dessus de la zone de saisie (cf. figure 2). Elle correspond au mot en cours de saisie et permet à l'utilisateur de conserver son attention dans un même contexte spatial.

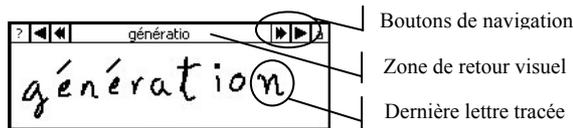


Figure 2 : exemple de saisie de caractères en contexte de mots

Navigation dans le micro éditeur. Pour pallier aux problèmes de la taille réduite de la zone de saisie, l'éditeur est doté d'un mécanisme de défilement automatique (fenêtre glissante) : la zone de saisie est donc considérée comme une feuille de papier virtuelle. Ce défilement est automatiquement déclenché lorsque l'utilisateur termine un tracé en fin de zone de saisie. Il peut naviguer dans le contenu de la feuille de papier virtuelle à l'aide de bouton de navigation (cf. figure 2), lui permettant de revoir et appliquer des commandes d'édition sur ce qu'il a saisi.

Gestion des caractères accentués. Les accents sont effectués directement sur l'encre électronique par des gestes graphiques correspondant à la forme de ces signes diacritiques. Toutes les lettres accentuables présentent sur la feuille de papier virtuelle peuvent l'être. La cédille est gérée de façon analogue aux accents.

Gestes associés aux commandes d'édition. Afin d'éviter l'utilisation de menus et boutons et ainsi rendre l'utilisation du micro éditeur plus conviviale, celui-ci offre un ensemble de commandes d'édition réalisables par des gestes graphiques (tracés) effectués directement sur l'encre électronique. Ainsi, il est possible de supprimer un ou plusieurs caractères en rayant l'encre électronique représentant ces caractères d'un trait de la droite vers la gauche (cf. figure 3). Les accents et cédille sont supprimables de la même manière. Il est aussi possible d'insérer un espace entre deux caractères, de changer la casse d'un caractère, ou de remplacer un caractère en écrivant directement sur celui-ci le caractère de remplacement.



Figure 3 : exemple de suppression de caractères

Nouvelle version du micro éditeur

Cette première interface répond aux principes de conception que nous nous étions imposés. Les choix effectués ont été guidés de manière intuitive par notre propre expérience de l'utilisation d'interfaces de saisie. Afin de pouvoir mener des expériences comparatives auprès d'utilisateurs pour évaluer la pertinence des solutions proposées du point de vue ergonomique, nous avons conçu une seconde interface respectant ces principes en y apportant des solutions différentes.

Afin de réduire le nombre de zones d'attention, nous proposons d'intégrer le retour visuel à la zone de saisie elle-même. Pour cela, nous avons couplé la saisie de caractères en contexte de mots avec le retour visuel. Une fois le tracé d'un caractère reconnu, l'encre électronique correspondant à ce caractère est remplacé dans la zone de saisie par le caractère d'imprimerie équivalent (cf. figure 4). Ainsi, l'utilisateur peut continuer sa saisie tout en sachant si son tracé a correctement été reconnu.



Figure 4 : exemple de conversion en caractères d'imprimerie

La navigation dans le micro éditeur et les gestes d'accentuation et d'édition restent identiques à la première interface. En revanche, ils s'effectuent directement sur les caractères d'imprimerie contenus dans la zone de saisie. Les modifications sont immédiatement répercutés sur ces caractères. Ceci permet à l'utilisateur de se rendre compte du résultat de sa saisie sans l'interrompre.

Ce nouveau mode de fonctionnement permet de libérer l'espace d'affichage dédié au retour visuel (cf. figure 4). Celui-ci peut ainsi être utilisé pour fournir une aide supplémentaire à l'utilisateur dans la saisie de texte comme par exemple la complétion de mots.

EVALUATION DE LA QUALITE ERGONOMIQUE DES SOLUTIONS PROPOSEES

Les principes de conception d'interfaces de saisie qui ont guidé nos travaux sont basés sur l'expérience acquise lors de l'embarquement du système RESIFCar et sur des travaux en psychologie ergonomique. Pour aller plus loin dans la validation de la qualité ergonomique des interfaces proposées, nous travaillons en collaboration avec le Centre de Recherche en Psychologie, Cognition et Communication de l'Université de Rennes 2.

Les expériences menées actuellement ont pour but de valider expérimentalement deux des hypothèses sous-tendues par les principes de conception :

- le respect de la contiguïté spatiale entre le retour visuel et la zone de saisie améliore les performances de l'utilisateur (nombre de mots saisis par minute);
- la saisie en contexte de mots soulage la mémoire de travail de l'utilisateur d'un effort de rétention des lettres déjà saisies et ainsi, a un effet sur l'efficacité mais aussi sur la satisfaction de l'utilisateur.

Les résultats d'une première série d'expériences auxquelles ont participé 87 sujets sont actuellement en cours d'analyse. Ces expériences ont consisté à saisir une liste de mots sur un assistant numérique personnel via différentes variantes de nos interfaces de saisie afin de mettre en lumière les mécanismes cognitifs mis en œuvre dans

ce type de tâche et d'étudier l'impact des deux hypothèses énoncées sur ces mécanismes. Les conclusions dégagées feront prochainement l'objet d'un article et vont permettre, aux vues du caractère exploratoire et vaste de cette étude, d'affiner les protocoles expérimentaux des tests suivants.

CONCLUSIONS ET PERSPECTIVES

Cet article présente une nouvelle orientation de nos travaux de recherche sur l'interaction homme machine orientée stylo à destination des périphériques mobiles. Après avoir conçu et embarqué le système RESIFCar sur des téléphones mobiles de nouvelle génération, nous nous attachons maintenant à améliorer l'ergonomie des interfaces de saisie s'appuyant sur ce système. Le retour d'expérience acquis lors de la commercialisation des téléphones mobiles équipés de ce système a mis en évidence l'importance de l'ergonomie de l'interface dans l'appréciation globale de l'application par l'utilisateur.

A partir de ce retour d'expérience et de travaux en psychologie ergonomique, nous avons dégagé un ensemble de principes de conception d'interface de saisie. Nous avons ensuite mis au point deux interfaces de saisie apportant des solutions différentes à ces principes.

L'ergonomie de ces interfaces doit être évaluée expérimentalement auprès d'utilisateur. Cette évaluation doit permettre notamment d'évaluer deux hypothèses que sous-tendent les principes que nous avons énoncés.

REMERCIEMENTS

Les auteurs remercient Eric Jamet, Maître de Conférence de l'Université de Rennes 2, et Guillaume Deconde, doctorant au CRPCC de l'Université de Rennes 2, pour leur collaboration dans l'étude de l'ergonomie des interfaces homme machine orientées stylo, ainsi que Guy Lorette, Professeur à l'Université de Rennes 1, pour avoir relu cet article.

BIBLIOGRAPHIE

1. Anquetil, E. et Bouchereau, H. Integration of an On-line Handwriting Recognition System in a Smart Phone Device. In *Proceedings of the 16th IAPR International Conference on Pattern Recognition (ICPR 2002)*, Quebec, 2002, pp. 192-195.
2. Anquetil, E. et Bouteruche, F. Conception d'un micro éditeur d'encre électronique et embarquement d'un système de reconnaissance d'écriture manuscrite sur téléphone mobile. Dans les Actes des Premières Journées Francophones : Mobilité et Ubiquité, Nice, 2004, pp. 151-157.
3. Jamet, E. *La recherche autour des applications et interfaces stylo : quelques pistes en psychologie ergonomique*, rencontres Irisa-Tech, Interfaces et ordinateurs stylo, octobre 2003. <http://www.irisa.fr/cdri/Irisatech/rencontres.html>.
4. Jamet, E. *L'intégration spatiale d'éléments textuels et illustratifs améliore-t-elle la performance ?* Revue d'intelligence artificielle : Les Interactions Homme-Système : perspectives et recherches psycho-ergonomiques, Vol.14, 2000, pp.167-188.
5. MacKenzie, I.S. et Soukoreff, R.W. *Text entry for mobile computing: Models and methods, theory and practice*. Human-Computer Interaction, Vol. 17, No 2&3, 2002, pp.147-198.
6. MacKenzie, I.S. et Chang, L. *A Performance Comparison of Two Handwriting Recognizers*. Interacting With Computers, Vol. 11, No 3, 1999, pp. 283-297.

Interfaces stylo pour la saisie d'écriture manuscrite sur systèmes mobiles de petite taille

F. Bouteruche

E. Anquetil

Résumé : ce poster présente nos travaux sur l'étude de l'ergonomie des interfaces homme machine stylo pour la saisie d'écriture manuscrite sur systèmes mobiles de petite taille. Il illustre les principes de conception que nous avons énoncés à partir de travaux en psychologie ergonomique et le retour d'expérience acquis lors de l'embarquement du système de reconnaissance de caractères manuscrits isolés RESIFCar sur téléphones mobiles de nouvelle génération.

Interaction homme machine orientée stylo

- Informatique nomade : transportabilité des machines
- Recherche de nouvelles ergonomies de communication : interactions intuitives

Contraintes matérielles des systèmes mobiles de petite taille

- Faible puissance de calcul
- Faible ressources mémoires
- Ecran de petite taille à faible résolution

Système de reconnaissance de caractères manuscrits : RESIFCar

- Reconnaissance de caractères manuscrits **isolés**
- Embarquement dans des téléphones mobiles de nouvelle génération (collaboration avec la société PurpleLabs)



Retour d'expérience

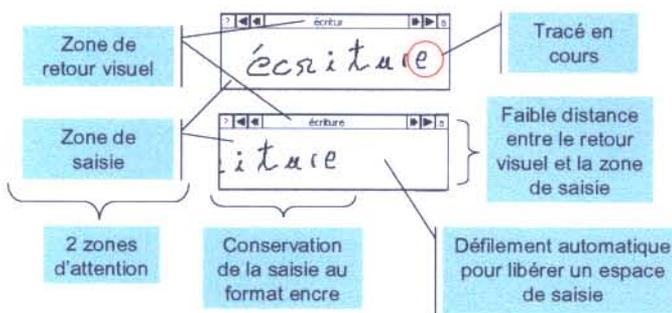
- La **qualité de l'ergonomie** de l'interface de saisie **influe** sur la satisfaction de l'utilisateur **indépendamment** des **performances** en termes de taux de reconnaissance

Interface stylo de saisie d'écriture pour système de petite taille : principes de conception

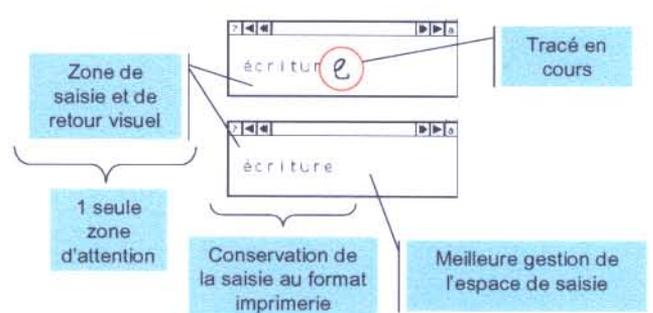
- Respect de la **contiguïté spatiale** et **minimisation des zones d'attention** pour soulager la mémoire de travail de l'utilisateur
- Saisie de caractères en **contexte de mots** pour s'approcher d'une saisie de mots cursifs
- **Gestion dynamique de l'espace de saisie** pour prendre en compte la taille réduite des écrans
- **Gestes d'édition intuitifs** pour profiter des possibilités d'interaction offertes par le stylo

Interface stylo de saisie d'écriture pour système de petite taille : exemples de réalisation

Micro éditeur d'encre électronique



Évolution du micro éditeur : fusion des zones d'attention



Interface stylo de saisie d'écriture pour système de petite taille : exemples de gestes graphiques d'édition associés

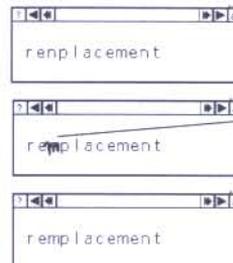
Geste de suppression



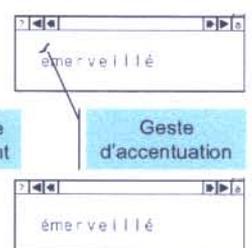
Geste d'insertion



Remplacement d'un caractère



Accentuation



Laboratoire



Structure Universitaire



Équipe de recherche



Les interactions multimodales et multicanal dans UMR

Vincent Chevrin

Laboratoire Trigone
Université de Lille 1
V.chevrin@ed.univ-lille1.fr
59655 Villeneuve d'Ascq cedex

RESUME

Le but de cette contribution est de faire un comparatif entre multimodalité et multicanal dans le cadre du marketing direct et de détailler la facette 1 de notre architecture pour le projet UMR (Ubiquitous Marketing Relationship) [3]. Dans un premier temps nous nous appuyons sur les approches CASE et CARE [1] pour effectuer la comparaison, puis nous passons aux aspects de fission et de fusion des flux d'interaction. Enfin, nous présentons un prototype de simulation de vente de produit via différents canaux (WAP, Web, téléphone).

MOTS CLES : Interaction, Multicanal, Multimodalité.

ABSTRACT

The goal of this contribution is to compare multimodality and multichannel within the framework of direct marketing and to detail facet 1 of our architecture for the UMR project (Ubiquitous Marketing Relationship) [3]. Initially we are based on the approaches CASE and CARE [1] to carry out the comparison, then we pass to the aspects of fission and fusion of flows of interaction. Lastly, we present a prototype of simulation of sale of product via various channels (WAP, Web, telephone).

CATEGORIES AND SUBJECT DESCRIPTORS: H.5.2 [User Interfaces]: User interface management system (UIMS);

GENERAL TERMS: Design, Human Factors.

KEYWORDS: Interaction, Multichannel, Multimodality.

INTRODUCTION

Le contexte de ces travaux s'intègre dans une collaboration avec la Cité Numérique, filiale du groupe 3 Suisses international.

Notre étude se situe à l'intersection entre les IHM d'une part et le marketing direct d'autre part.

Ainsi, la problématique de nos travaux se localise autour de la question générale du marketing relationnel

multicanal. Les entreprises veulent de plus en plus se tourner vers des solutions de type multicanal, mais cela pose certains problèmes. En fait, ces entreprises souhaitent intégrer plusieurs canaux de communication et ne pas se contenter de les juxtaposer au fur et à mesure de leur apparition. De plus, une refonte des systèmes d'information est incontournable [9].

Mais les systèmes multi-canaux posent d'autres problèmes. En effet, l'un des principaux enjeux est de minimiser les risques de « rupture » lors du couplage, du découplage ou tout simplement d'un changement de canal. Cela peut alors se solder par la perte d'une occasion de vente, ce qui est préjudiciable pour l'organisation [4].

Dans de précédents travaux [3] nous présentions le projet UMR qui vise à concevoir une architecture supportant les interactions multimodales et multicanal, basée, en partie, sur notre cadre théorique [2]. Dans [3] nous étudions rapidement l'ensemble de l'architecture en nous focalisant plus précisément sur l'aspect composition de e-Services. Le but de cette nouvelle contribution est de nous intéresser plus précisément à la Facette 2 de cette architecture, c'est-à-dire au modèle d'adaptation aux communications multicanal et multimodales (MACMM).

MULTIMODAL ET MULTICANAL : RAPPROCHEMENT SUR UN CADRE DE REFERENCE

Pour étudier conjointement le multicanal et le multimodal, nous avons choisi de les comparer sur un cadre de référence. Nous utiliserons donc CASE et CARE [1] pour faire ces comparaisons afin d'identifier les liens entre ces deux domaines.

Le point de vue système : CASE

La *Figure 1* représente les propriétés CASE de la multimodalité en entrée. Par translation, nous pouvons imaginer les mêmes propriétés pour le multicanal. Cependant, tous les cas de multimodalité sont-ils équivalents en multicanal et en multimodal ? C'est à cette question que nous allons essayer de répondre, à présent. Les modalités sont dites *alternées* quand il y a un entrelacement temporel pour effectuer une tâche, en coréférence de modalité. Par exemple, je fais un cercle avec ma souris et puis, ensuite, je prononce « trace ». Un cercle est alors dessiné dans ma fenêtre. Il y a bien ici un entrelacement séquentiel de mes modalités, gestuelle et vocale. C'est le même cas lorsqu'un client se connecte

sur un site de commerce électronique, qu'il remplit son caddie virtuel, qu'il termine sa tâche en validant. Ensuite, il appelle une conseillère par téléphone afin de payer, à l'aide de sa carte de crédit, et enfin, il consulte son e-mail pour voir si sa commande a bien été enregistrée. La différence que l'on remarque ici réside dans l'entrelacement temporel des modalités/canaux. Si dans le premier cas, le grain temporel doit être très fin, sous peine de perdre toute cohérence dans l'interaction, en multicanal, ce grain peut être très épais. Cela est dû aux exigences du marketing direct, la persistance des données doit être assurée, afin de garder une cohérence de l'interaction, quelle que soit sa durée.

La modalité employée est de type *synergique* lorsqu'une tâche est effectuée en parallèle, en coréférence de modalité. Dans l'exemple précédent, je prononcerais « trace » en même temps que je ferais le geste à l'aide de la souris. En multicanal, c'est le cas lorsqu'un client prononce au téléphone « je veux cette chemise, en taille 38 », et qui parallèlement, sur la page du catalogue, clique sur l'article souhaité. On ne retrouve pas cette différence de temporalité, le grain temporel est forcément très fin. Il est inutile de différencier multicanal et multimodal, les deux domaines seront équivalents dans ce cas.

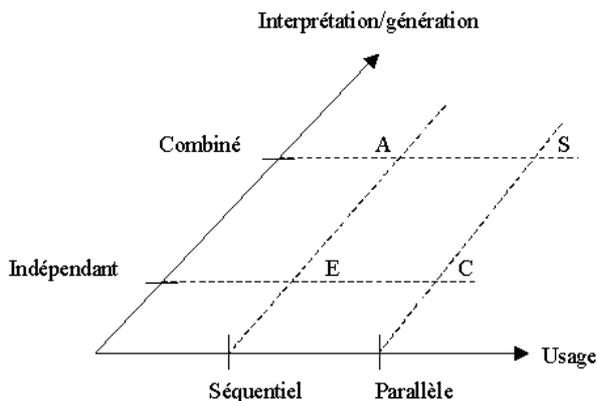


Figure 1 : Propriétés CASE de la multimodalité [1]

Les modalités sont dites *concurrentes* lorsque sont exécutées deux tâches distinctes en parallèle, sans coréférence. Par exemple, un utilisateur dicte des phrases à son logiciel de reconnaissance vocale pendant qu'il joue au démineur à l'aide de sa souris. On peut imaginer des exemples en multicanal. Un client regarde le catalogue numérique d'un groupe de VAD (Vente à Distance) pendant qu'il paie sa dernière commande par téléphone via une conseillère avec sa carte bancaire. Il est clair que dans ce cas encore, l'entrelacement temporel est étroit, car on se place là encore dans une utilisation des modalités/canaux parallèle. On ne peut donc pas trouver ici non plus de différence entre les deux domaines.

Enfin, la modalité employée est de type *exclusive*, lorsqu'une tâche est exécutée à la fois, sans usage de la multimodalité. C'est donc, ici, du monocanal et non du multicanal. On ne relèvera pas de différence là non plus.

Le point de vue utilisateur : CARE

Notre argumentation va se structurer de la même manière que précédemment en nous basant sur la Figure 2.

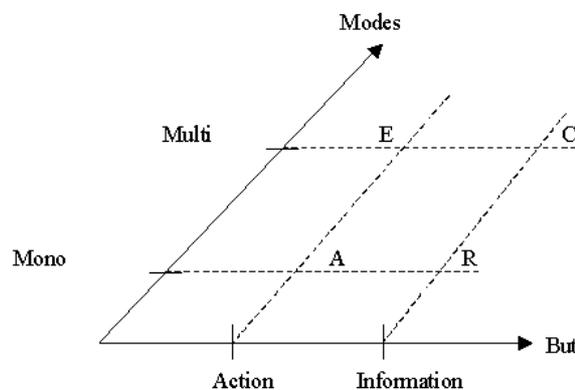


Figure 2 : Propriétés CARE de la multimodalité [1]

La notion d'*équivalence* fait référence au fait que l'utilisateur ou la machine peut avoir le choix entre plusieurs modalités pour formuler un énoncé particulier. Un utilisateur peut avoir le choix entre prononcer le mot « suivant » ou cliquer sur le bouton « suivant » à l'aide de sa souris. Pour un client, cela peut se traduire par exemple par un choix lors du paiement. Il peut appeler une conseillère, envoyer un chèque par la poste, ou simplement entrer son numéro de carte bancaire via le Web. Nous ne notons pas de différence à ce niveau.

La *complémentarité* consiste à transmettre différents messages représentant les constituants d'un même énoncé sur plusieurs modalités. La compréhension de l'énoncé nécessite ici une fusion entre les différents messages transmis à travers les différentes modalités. Par exemple, le système dit vocalement : « le résultat de votre requête est : » et liste un ensemble de réponses sur l'écran. Dans le cas du multicanal, on peut par exemple considérer un client qui reçoit un SMS contenant une URL qu'il peut aller visiter, via son navigateur Web, s'il veut savoir de quelles promotions il peut bénéficier ce mois-ci. Dans ce cas, on retrouve la même différence que dans le cas alterné de CASE. En effet, la granularité temporelle en multimodalité doit être fine pour garder une cohérence de l'interaction, tandis qu'en multicanal, le grain peut être très épais sans souci de cohérence.

L'*assignation* consiste à utiliser toujours la même modalité pour un type d'énoncé particulier et à ne pas l'utiliser pour d'autres types d'énoncés (spécialisation exclusive). C'est le cas d'un utilisateur qui utiliserait toujours la voix ou du texte pour effectuer une tâche définie, ou un client qui utilise toujours le web pour remplir son panier (caddie virtuel) et toujours le téléphone (via une conseillère) pour payer par carte bancaire. Nous ne notons pas de différence dans les deux domaines ici.

La *redondance* consiste à transmettre un même message via différentes modalités. En principe, l'analyse d'un énoncé transmis à travers une des modalités est

suffisante pour dégager toutes les informations sémantiques véhiculées par cet énoncé sans avoir besoin d'analyser les autres énoncés transmis sur les autres modalités. Par exemple, un utilisateur reçoit le résultat d'une requête vocalement au travers d'enceintes et textuellement au travers de l'écran. Cela peut se traduire par la réception d'un SMS et d'un e-mail de confirmation d'une commande (contenant les mêmes informations). Nous ne notons, là non plus, pas de différences.

Multicanal / Multimodal : Une continuité

On peut constater que les deux domaines sont très proches et que c'est au niveau des temporalités que les différences apparaissent. Effectivement, en multimodalité, l'interaction a un cycle de vie très court, de l'ordre de la seconde, voire de la milliseconde. En multicanal, il en est tout autrement : une interaction peut durer un petit laps de temps (une seconde) ou plus longtemps (une minute, une journée, un mois, etc.). On peut s'apercevoir de cela lorsque l'on a un entrelacement temporel des modalités/canaux non parallèle (séquentiel). Il n'y a donc pas de frontière figée entre multicanal et multimodal, mais plutôt une continuité. On va alors considérer que les interactions dont le grain de synchronisation entre les canaux est faible seront dites multimodales, et progressivement, avec l'épaississement du grain temporel, on va glisser vers le multicanal. Il ne faut pas oublier que le multicanal est issu du domaine du marketing direct (pour notre étude) et que, ce concept a pour intérêt de toucher plus facilement, plus durablement et plus efficacement le client. Le multicanal a donc, ici, pour vocation, de construire une « bonne » relation avec le client. En bref, le multimodal peut être qualifié à l'échelle d'une application, tandis que le multicanal apparaît à l'échelle d'un domaine d'application.

ASPECT GENERAL DE LA FACETTE 1 MACMM

Si l'on revient sur l'architecture du projet UMR [3], c'est la facette 1 qui a pour but de gérer le couplage et le découplage des canaux et des modalités. Cela entraîne un certain nombre de contraintes. Tout d'abord, il faut une ontologie des canaux, et des heuristiques de choix pour coupler ou découpler tels ou tels canaux ou modalités. Cela est traité par notre cadre théorique [2] et nous ne reviendrons pas sur ces points dans cet article.

Ensuite, si nous couplons et découplons des canaux et/ou des modalités, nous devons assurer la fission et la fusion du flux d'interaction. En effet, le flux de données et le flux de contrôle doivent être adaptés en fonction du ou des canaux/modalités utilisés. Ce sont les deux points sur lesquels nous allons discuter maintenant.

La fusion du flux d'interaction

La fusion du flux d'interaction ne peut intervenir qu'en entrée, c'est-à-dire que celui-ci va être composé du ou des canaux utilisés vers le système. En fait, les flux d'interaction provenant de l'utilisateur, et donc de tous les canaux (et/ou modalités) qu'il utilise, que ce soit un

flux de données ou un flux de contrôle ([3]), vont être fusionnés. Par ailleurs, cette opération demande une grande synchronisation des informations pour pouvoir donner du sens à cette interaction. La *Figure 3* schématise cela. Au niveau de la multimodalité, la fusion du flux d'interaction se fera sur un laps de temps très court. En revanche, en multicanal, ce laps de temps pourra être très long. La persistance des flux d'interaction devra donc être assurée.

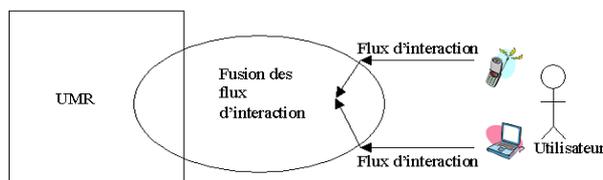


Figure 3 : Fusion du flux d'interaction

La fission du flux d'interaction

La fission du flux d'interaction ne peut intervenir qu'en sortie, c'est-à-dire que celui-ci va être composé du système vers le ou les canaux. Le mécanisme est similaire à la fusion du flux d'interaction, à la différence que cette fois-ci le flux est décomposé en autant de canaux (et/ou modalités) que le client en utilise, et de manière « intelligente ». Evidemment, là encore, une synchronisation précise est nécessaire. La *Figure 4* schématise cela. De plus, comme dans le cas de la fusion du flux d'interaction, les temporalités ne seront pas les mêmes dans les deux domaines.

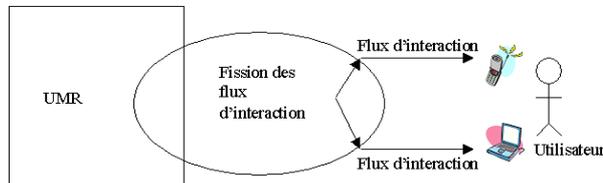


Figure 4 : Fission du flux d'interaction

Pour gérer la fusion et la fission du flux d'interaction, nous devons mettre en place des « opérateurs » spécifiques. Nous allons en ce sens en nous basant sur des travaux comme ceux de Moran et al. [6] qui tentent de résoudre ces problèmes à l'aide d'agents. Pour le moment, nous avons mis un prototype en place gérant la fusion du flux d'interaction (nous revenons rapidement sur celui-ci un peu plus loin dans cette contribution), mais sans l'apport de la technologie multi-agents.

Enfin, la présentation des données devra être adaptée au(x) périphérique(s) utilisé(s) (contraintes physiques, réseaux d'accès, etc.). De récents travaux de notre équipe [8] vont en ce sens avec PlasticML.

UN PROTOTYPE

Présentation

Le prototype que nous avons implémenté est un simulateur de vente de produit via le web, le téléphone ou le WAP. Nous pouvons présenter deux concepts importants dans notre démarche :

- la session : contient les opérations (tâches) inachevées.
- les tâches (e-Services [3]) : un utilisateur peut réaliser plusieurs tâches dans sa session. En Marketing Direct une tâche peut être une commande, en E-learning, un exercice en ligne... Tant que la session contient des tâches non terminées et que l'utilisateur est connecté, on ne la ferme pas.

Ainsi, ce prototype permet de gérer plusieurs types de problèmes. Tout d'abord, un mécanisme de persistance des données (mémoire) assure la reprise d'une interaction là où elle s'est arrêtée lors d'une rupture volontaire ou pas. Ensuite, le système gère la fusion du flux d'interaction, c'est-à-dire que le client peut utiliser plusieurs canaux (modalités) pour accomplir ses tâches, et c'est le système qui se charge de synchroniser et « d'assembler » les données pour lui, de manière transparente.

La Figure 5 montre un exemple de ce que l'on peut faire (en terme de multicanal) avec ce prototype.

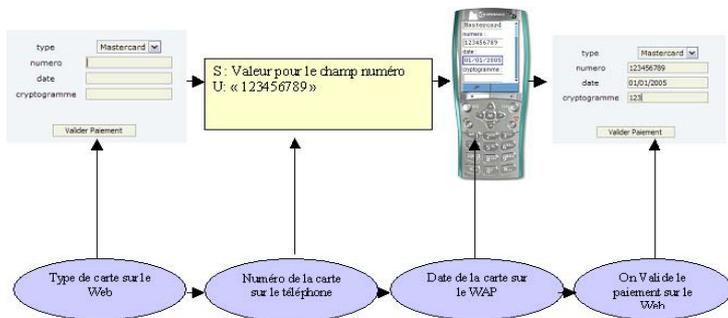


Figure 5 : Exemple d'utilisation du prototype en multicanal avec le téléphone, le Web et le WAP

Limites

La mise en place de ce prototype nous a permis de soulever plusieurs problèmes, par exemple, celui des authentifications multiples lors des ruptures. Les travaux de [10] pourraient être une solution à cela dans le futur. Ensuite, nous avons vu que la granularité temporelle est un aspect important lors du couplage et découplage des canaux/modalités, ce qui n'est pas forcément aisé à gérer de manière fluide et utilisable (utilisabilité [7]) par l'utilisateur.

Enfin, la liaison entre le déroulement des tâches (nous détaillons notre approche e-Service dans [3]) et la composition des canaux est un aspect difficile à gérer.

CONCLUSION

Dans cette contribution, nous essayons de mettre en évidence le rapprochement des domaines de multicanal et multimodal. Selon nos investigations nous pensons qu'il n'existe pas de frontière tangible entre les deux, mais plutôt une continuité basée sur une granularité temporelle croissante (multimodal vers multicanal).

La réalisation d'un prototype nous montre certaines difficultés auxquelles nous devons faire face lors de l'implémentation de notre démonstrateur.

Comme nous l'avons déjà exprimé, notre choix porte sur Jade [5], une plate-forme multi-agents pour l'implémentation.

Pour réaliser ce démonstrateur, nous nous baserons sur notre architecture UMR [3], reposant elle-même sur notre cadre théorique [2] et sur une ontologie des canaux qui fera l'objet d'une prochaine contribution.

REMERCIEMENTS

L'auteur tient à remercier Yves Bayart, Directeur de la Recherche et Développement du Groupe 3 Suisses International, et la Cité Numérique, ainsi que le groupe NIPO (Nouvelles Interactions Personnes/Organisations) et la Région Nord Pas de Calais pour leur soutien dans ces travaux de recherche. Un grand merci également à Alain Derycke et José Rouillard (laboratoire Trigone) pour leur soutien et leurs conseils.

BIBLIOGRAPHIE

1. Caelen Jean. 10 ans de recherches... en multimodalité(s). Colloque sur les Interfaces Multimodales - 09/10 Mai 2000. IRIT. Toulouse.
2. Chevrin Vincent, Derycke Alain, Rouillard José, Un Cadre Théorique pour la Caractérisation des Interactions Multicanal en E-Marketing, IHM 2003, Caen, 2003. pp. 97-104.
3. Chevrin Vincent, Derycke Alain, Rouillard José, L'Architecture Logicielle UMR pour les interactions Multicanaux et Multimodales avec les e-Services, IHM 2004, Namur, Belgique, 2004. pp. 199-202.
4. Derycke, A., Rouillard, J., Chevrin, V., Bayart, Y. When Marketing meets HCI: Multi-channel customer relationships and multimodality in the personalization perspective. HCI International 2003, Heraklion, Crete, Greece, 2003, pp. 626-603 Volume 2.
5. JADE : <http://jade.tilab.com>
6. Moran, D. et Al. Multimodal User Interfaces in The Open Agents Architecture. Proceedings IUI'97 ACM conference, Orlando, Florida, 8p.
7. Nielsen J. Chapter 5 : Usability Heuristics, in *Usability Engineering*, Academic Press, 1993.
8. Rouillard, José, HCI 2003, *Plastic ML and its toolkit*, Heraklion, Crete, Greece, 2003.
9. Rouillard José et Derycke Alain. La Personnalisation de l'Interaction dans des Contextes Multimodaux et Multicanaux : une Première Approche pour le Commerce Electronique, IHM 2002, Poitiers, 2002, pp. 97-104.
10. Van Thanh, Do. Vanem Erik, Tran, Dao van & Tore E. Jønvik: Extending the "Always-on" concept to heterogeneous devices, Proceedings of the 14th International Symposium on Services and Local access (ISSLS 2002), Seoul, Korea, April 14-17 2002.

Multicanal / Multimodal : Quelles différences ?



Vincent Chevrin - Laboratoire TRIGONE – Lille - USTL
Rencontres Jeunes Chercheurs en Interaction Homme-Machine
Lacanau - 2004



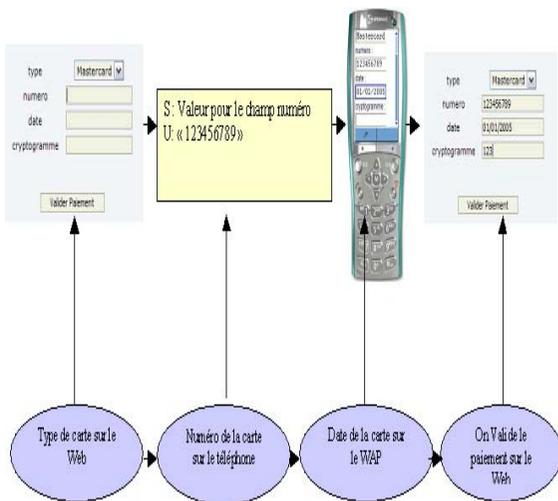
Problématique

Dans les domaines de l'IHM et du marketing direct, on parle respectivement de multimodal (MM) et de multicanal (MC). Existe-t-il un lien entre ces deux domaines ? Pouvons nous faire un rapprochement cohérent entre ces deux termes ?

Comparaison sur un cadre de référence

		C	A	S	E			C	A	R	E
MM \ MC	MC	Concurrent	Alterné	Synergique	Exclusif	MM \ MC	MC	Complémentarité	Assignation	Redondance	Équivalence
Concurrent		Équivalence				Complémentarité		Temporalités différentes. Distinction nette entre multicanal et multimodal			
Alterné			Temporalités différentes. Distinction nette entre multicanal et multimodal			Assignation		Équivalence			
Synergique				Équivalence (on tend vers la multimodalité, car grain temporel fin)		Redondance			Équivalence		
Exclusif					Équivalence	Équivalence					Équivalence

Prototypage



Un client effectue une tâche de paiement via notre application. Il commence par remplir le premier champs par le canal Web. Il remplit le second champs à l'aide du canal vocal. Le troisième champs est saisi sur le canal Wap. Et enfin le client revient sur le canal Web pour valider le paiement.

L'application rassemble toutes les données des différents canaux pour former un tout cohérent, elle fait alors une fusion du flux d'interaction.

Dans le cas où c'est le système qui décompose le flux d'interaction vers différents canaux (utilisés par le client), on se place dans une situation de fission du flux d'interaction.

Ce prototype gère uniquement la fusion du flux d'interaction avec une certaine granularité et pas la fission.

Conclusion / Perspectives

Nous essayons de mettre en évidence le rapprochement des domaines de multicanal et multimodal. Selon nos investigations nous pensons qu'il n'existe pas de frontière tangible entre les deux, mais plutôt une continuité basée sur une granularité temporelle croissante (multimodal vers multicanal).

La réalisation d'un prototype montre certaines difficultés auxquelles nous devons faire face lors de l'implémentation de notre démonstrateur. Notamment au niveau de la fusion et de la fission du flux d'interaction et de la granularité avec laquelle on les effectue. Nous nous sommes désormais lancé dans l'élaboration d'un démonstrateur basé sur une plate-forme multi-agents (Jade).

Le problème de la plasticité dans la conception d'IHM

Alexandre Demeure

Laboratoire CLIPS-IMAG, équipe IHM
385, rue de la bibliothèque – BP 53
38041 Grenoble Cedex 9, France
alexandre.demeure@imag.fr

RESUME

Cet article décrit mes travaux de thèses, ceux-ci portent sur la plasticité des IHM et plus particulièrement sur la conception et l'exploitation de tels IHM.

MOTS CLES : plasticité, modèles d'évolution, interfaces NG, capitalisation.

ABSTRACT

This paper describes my thesis works, it is about HCI plasticity and more specifically about conception and use of such softwares.

CATEGORIES AND SUBJECT DESCRIPTORS: H.5.2 [GUI].

GENERAL TERMS: Design

KEYWORDS : plasticity, evolution models, NG interfaces, capitalisation.

INTRODUCTION

La plasticité d'une interface dénote sa capacité à s'adapter aux variations du contexte d'usage en termes d'utilisateur, de plate-forme et/ou d'environnement dans le respect de son utilisabilité. L'*utilisabilité* est un facteur qualité logiciel [4]. Il relève de l'acceptation d'un système [5]. Il se réfère à l'adéquation d'un système interactif aux capacités de l'utilisateur. L'*utilisabilité* complète l'*utilité*, relative à l'adéquation du système interactif aux besoins des utilisateurs, pour caractériser, de façon plus globale, la serviabilité ou « usefulness » du système (Figure 3). La serviabilité se réfère à la faculté du système à permettre à l'utilisateur d'atteindre ses buts¹. L'*utilisabilité* en qualifie les facilités d'apprentissage et d'appropriation, l'efficacité à l'usage, la robustesse aux erreurs et enfin, de façon plus subjective, la convivialité ou le plaisir concrètement ressenti à l'usage. Aussi, l'*utilisabilité* ne se limite pas à des critères de performance dans l'accomplissement de tâches. Elle se réfère, de façon plus générale, à la satisfaction de buts personnels et collectifs² (Gilmore, 1995).

¹ « Usefulness is the issue of whether the system can be used to achieve some desired goals » (Nielsen, 1993) p 24.

² « Redefine usability as successful fulfillment of user and organisational goals, rather than rapid, error-free performance of unit-tasks » (Gilmore, 1995) p 177.

L'objet de ma thèse est d'étudier certains points relatifs à la conception de systèmes interactifs plastiques. En particulier il s'agira de formaliser un modèle explicite d'évolution du système et de proposer une méthode de classification des systèmes interactifs afin de favoriser la réutilisation et l'interopérabilité des systèmes interactifs à la conception et à l'exécution.

LE MODELE D'EVOLUTION

L'adaptation du système interactif peut être plus ou moins profonde selon qu'elle porte sur une réorganisation des concepts et des tâches, des interfaces abstraite, concrète ou finale. L'adaptation du système peut ne concerner qu'une sous partie de ce dernier, on parle dans ce cas de l'adaptation d'éléments d'interactions. Dans tous les cas nous identifions quatre classes de stratégies :

- l'adaptation par polymorphisme. C'est une stratégie qui conserve l'élément mais en change la forme. Ce changement de forme peut s'opérer à tout niveau de réification selon trois cardinalités, 1-1, 1-N, N-1, selon que la forme est remplacée par une autre (cardinalité 1-1), N autres (cardinalité 1-N) ou que N formes sont agrégées en une seule (cardinalité N-1) ;
- l'adaptation par substitution. Par opposition au polymorphisme, cette stratégie ne préserve pas l'élément. Elle la remplace par un ou N autres. On distingue trois types de substitution selon leur cardinalité : la substitution 1-1 consiste à remplacer l'élément par un autre ; la substitution 1-N remplace l'élément par N autres ; la substitution N-1 remplace l'élément et N-1 autres par un unique autre ;
- l'adaptation par ajout. Un élément est ajouté au système interactif. C'est le cas typiquement de la représentation multiple d'un même concept, sollicitée pour de l'insistance temporaire ;
- l'adaptation par suppression : c'est la duale de l'ajout. Un élément est supprimé parce que l'information n'est plus critique ou que la tâche ne fait plus sens dans le contexte d'usage courant.

Les stratégies sont déployées selon des politiques. Ces politiques dépendent de l'autonomie accordée à l'élément dans sa prise en charge de l'adaptation : reconnaissance de la situation, calcul et mise en oeuvre de la réaction. Pour chacun de ces aspects, un degré

d'autonomie est accordé à l'élément. Ce degré varie de 0 à 1, donnant lieu à quatre classes de politiques :

- la politique *non concertée externe* prive l'élément de système interactif de toute autonomie. L'évolution est calculée et mise en oeuvre par un tiers (un autre élément ou l'infrastructure d'exécution) ;
- à l'opposé, la politique *non concertée interne* confère à l'élément une autonomie maximale. Il décide seul de la réaction à mettre en oeuvre et l'applique en pleine autonomie ;
- les politiques *concertées* correspondent à une autonomie partielle : une négociation s'établit entre l'élément et un tiers (un autre élément ou l'infrastructure d'exécution) pour la prise en charge de l'adaptation. Tandis qu'une version optimiste autorisera l'élément à appliquer la décision sans accord préalable du tiers, une version pessimiste requerra cet accord avant toute application. La version optimiste s'expose à devoir annuler, en cas d'erreur, les mesures mises en oeuvre.

Le choix de la politique se fera au regard de critères tels que la performance par exemple. Seule la politique non concertée interne est aujourd'hui mise en oeuvre. Le modèle d'évolution est distribué au sein des différents agents composant la hiérarchie. C'est un ensemble de règles du style « *Si condition Alors Proposer(réaction, attributs)* » où :

- la *condition* porte sur le contexte d'usage ;
- la *réaction* spécifie de façon plus ou moins directive la réaction à mettre en oeuvre. Par exemple, « *migrer un composant sur PDA* », « *migrer la télécommande sur PDA* » ou « *remplacer le composant logiciel télécommande PC par le composant logiciel télécommande PDA-Version 3.4 à exécuter sur la PDA* ». La réaction est aujourd'hui décrite par la fonction logicielle permettant de l'exécuter. Elle est assortie d'un texte en langue naturelle qui en décrit l'effet ;
- les attributs décorent la proposition d'une force, d'un qualificatif et indiquent l'auteur de la proposition (utilisateur/interacteur). La force exprime le caractère conseillé ou déconseillé de la proposition ; le qualificatif en exprime la nature « De convenance » ou « De survie ». Une proposition de convenance relève du confort de l'utilisateur alors qu'une proposition de survie engage le caractère opérationnel du système interactif. C'est le cas typiquement d'un logiciel sur PDA dont la batterie faiblit et dont la migration s'impose.

Ces mécanismes d'évolution supposent :

- une base de données capitalisant les interacteurs disponibles. Ces interacteurs peuvent être de granularité variable. Ils peuvent se limiter aux interacteurs classiquement disponibles dans les boîtes à outils actuelles (par exemple, la notion de choix qui, cette fois, disposera de différentes présentations, en particulier, les boutons radio et menu déroulant) ou s'étendre à des interacteurs métier (par exemple, des télécommandes de diaporama ou même des logiciels de présentation). Ils sont capitalisés à différents niveaux de réification conformément au cadre de référence (liens concepts-tâches, interfaces abstraite, concrète et finale) ;
- des mécanismes de recherche d'information permettant l'exploitation de cette base de données. Ces mécanismes supposent une description de chaque composant capitalisé.

LE GRAPHE DES DESCRIPTIONS

Le GDD organise les descriptions des logiciels, quelque soit leurs niveaux d'abstractions, ces descriptions forment les nœuds du graphe. Il établit les liens qui relient ces logiciels, ces liens peuvent être relatifs à une notion d'héritage ou de composition. Nous illustrons cela dans la figure 3 en fin d'article. La description contenue dans chaque nœud comprend :

- une description des ports de communication en WSDL, un langage standard de fait dans le domaine des web-services.
- Un lien vers un exécutable ou un programme source si on a à faire à une version implémenté ou semi implémenté (squelette).
- Une description de l'interface dans un langage pivot à définir.

Le graphe des descriptions est utile en conception et à l'exécution.

En conception, il incite le concepteur à situer son logiciel par rapport à des logiciels existants, déjà référencés. Il favorise une vision large consistant à percevoir un logiciel comme une partie d'un ensemble interconnecté de composants, classés de façon rationnelle. Il soulève typiquement des questions: comment et où s'insère ce nouveau logiciel ? Quelles relations entretient-il avec les autres logiciels ? Quelles parties devrait-il rendre les plus indépendantes possibles pour en permettre la réutilisation ? Et enfin, quelles parties des autres applications pourraient être réutilisées dans ce logiciel ?

A l'exécution, le GDD permet de trouver des remplaçants à un composant donné. Ceci est utile notamment lorsqu'on cherche à migrer une application d'une plateforme vers une autre. Il peut par exemple s'agir de migrer une télécommande d'un PC vers un PALM, auquel cas il faut changer le logiciel lui-même. Dans le cas où la

télécommande de départ est spécifique à un logiciel bien particulier (figure 3), il se peut qu'aucune version n'est été prévue pour le PALM, auquel cas il faudra se rabattre sur une télécommande plus générale (figure 3).

UNDE BOITE A OUTILS ADAPTEES A LA DISTRIBUTION DES INTERFACES

La plasticité des interfaces pose aussi le problème de la distribution de celles-ci sur plusieurs ordinateurs simultanément. C'est notamment le cas dans des projets comme (Lachenal, 2003). Il s'agit idéalement de la mettre en œuvre sans surcoût pour le programmeur et l'utilisateur. D'un point de vue fonctionnel, la répartition pose cinq problèmes :

- α : Découverte des ressources d'interaction, dispositifs de saisie, de pointage, surfaces ou espaces 3D. d'affichage, éventuellement position des utilisateurs.
- β : Mise en relation des espaces logiques et physiques. L'espace logique est un modèle de l'espace dans lequel les interacteurs prennent place, il s'agit classiquement d'un plan mais ce pourrait être un espace 3D. L'espace physique est un modèle de l'espace qui nous entoure, généralement 3D. L'utilisation typique va consister à plaquer des portions d'espaces logiques sur des portions d'espaces physiques (écrans, murs, etc.).
- χ : Expression des relations entre utilisateurs, dispositifs et surfaces/espaces. Il s'agit d'exprimer qu'un dispositif correspond à un utilisateur donné ou qu'une surface est disposée à gauche d'une autre.
- δ : Répartition des éléments présents dans l'espace logique sur l'espace physique. Il s'agira typiquement de distribuer un interacteur sur plusieurs écrans.
- ε : Outils graphiques, offrant primitives et interacteurs, ceux qui peupleront l'espace logique.

Les fonctionnalités précédemment décrites ont des dépendances que nous explicitons dans la figure 1. Une dépendance est symbolisée par un côté commun, une flèche indique le sens de la dépendance.

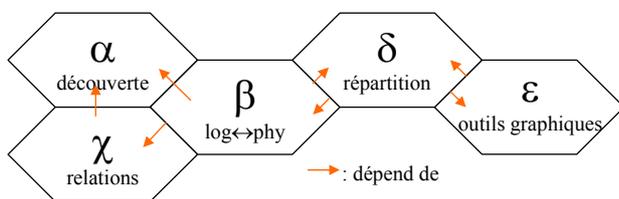


Figure 1: Décomposition fonctionnelle d'une boîte à outil assurant la distribution.

Observons les influences entre fonctionnalités.

- $\alpha \rightarrow (\beta \text{ et } \chi)$: β et χ dépendent de α qui les informe des ressources d'interaction détectées. β en a besoin pour connaître l'espace. χ en a besoin pour calculer les relations entre les entités physiques, par exemple le fait qu'une souris soit proche d'un clavier.
- $\chi \rightarrow \beta$: La position des surfaces entre elles dans l'espace physique peut permettre de calculer des mises en correspondance entre les espaces logique et physique. Par exemple, si un écran E2 se trouve à droite d'un écran E1 affichant l'espace logique entre 0 et 1024, on pourra en déduire que l'écran E2 affiche l'espace logique entre 1024 et 2048.
- $\beta \leftrightarrow \delta$: β émet les vues sur l'espace logique et les surfaces physiques auxquels elles correspondent, permettant à δ d'effectuer les rendus. Le mécanisme de répartition renseigne β sur les zones occupées de l'espace logique. Par exemple qu'une zone contenant des informations n'est pas affichée.
- $\delta \leftrightarrow \varepsilon$: ε émet, pour chaque plate-forme, les informations relatives aux interacteurs. Le mécanisme de répartition les répercute sur les autres plates-formes.

Les dépendances entre les modules fonctionnels n'ont pas toutes la même force. Nous voyons deux blocs se dessiner. D'un côté, nous trouvons le mécanisme de répartition qui est très dépendant des outils graphiques. Ils forment ensemble la couche basse de la distribution. De l'autre, nous trouvons la découverte des entités, le calcul des relations et la mise en correspondance entre les espaces physique et logique. Ensemble, ils forment la partie haute, « intelligente » de la distribution. Il nous paraît indispensable de bien séparer au moins ces deux groupes. En effet, il est souhaitable de pouvoir factoriser les algorithmes de mise en correspondance. Il nous semble illusoire de penser qu'une seule BAOIG sera adaptée pour tout. Soit par les technologies mises en œuvre, soit par leur logique intrinsèque (basée ou non sur des graphes de scène, ayant des interacteurs ou des primitives graphiques seulement, etc.), des BOAG très différentes verront le jour, ciblant chacun un problème précis.

En ce qui concerne la répartition (δ), en supposant qu'on adopte la structure de graphe de scène pour l'organisation des interacteurs (Demeure, 2004) (Ubit, 2004), nous proposons que le partage se fasse au niveau de sous graphes (figure 2). Pour l'ordinateur 1, les nœuds partagés par l'ordinateur 2 seront répliqués dans le sous graphe ayant pour racine le nœud pivot. L'ordinateur 2 fait de même de telle sorte que les sous graphes partagés par 1 et 2 soient les mêmes. Chacun accède à son sous graphe partagé par un nœud point de vue qui détermine la portion d'espace visualisée.

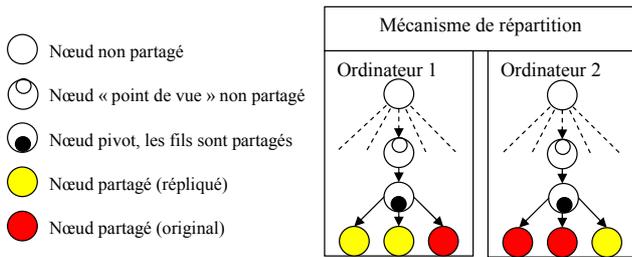


Figure 2: Partage de sous graphes entre ordinateurs.

CONCLUSION

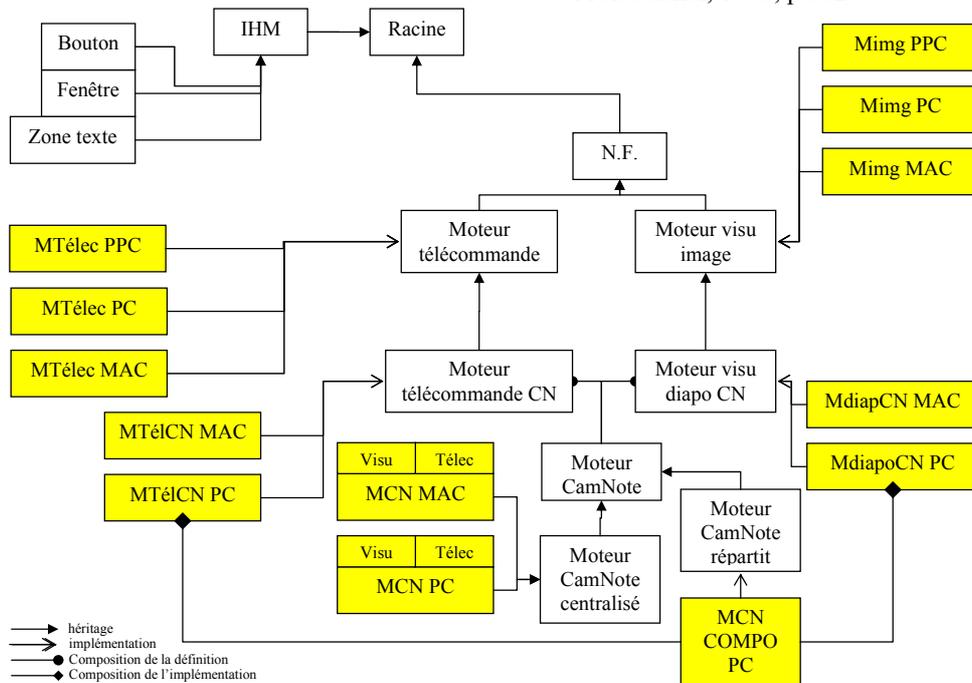
La plasticité des interfaces est un sujet qui pose de nombreux problèmes. Parmi ceux-ci je traite les problèmes suivants :

- modélisation de l'évolution de systèmes plastique en explicitant les règles qui déterminent celle-ci.
- capitalisation des systèmes plastique pour facilité la réutilisation en conception et à l'exécution.

- Création d'une boîte à outil supportant la distribution des interfaces au niveau du pixel et exploitant les capacités d'OpenGL.

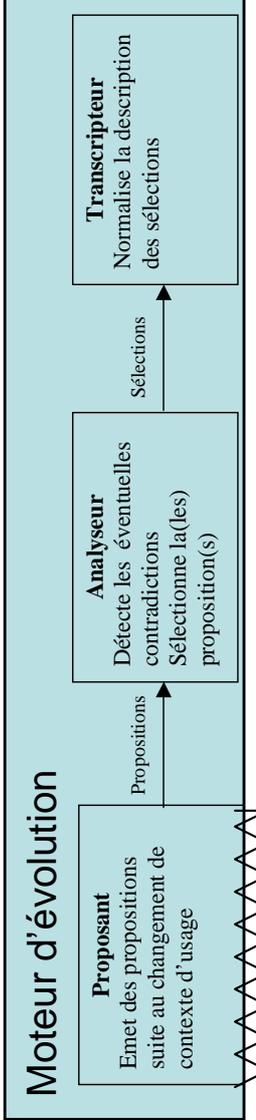
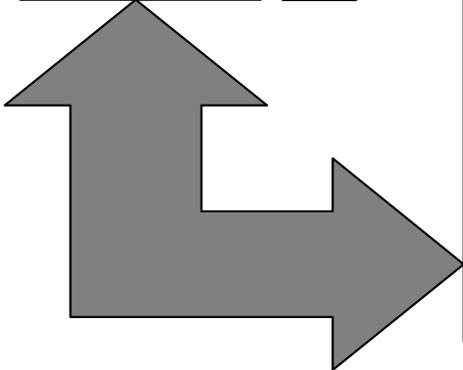
BIBLIOGRAPHIE

1. Demeure A., Rouillard S. Bérard F., Calvary G., *Requis et pistes pour les futures boîtes à outils d'interaction graphiques*. IHM'04, Namur.
2. Gilmore D.J., *Interface Design : Have we got it wrong ?*, Human Computer Interaction, *Interact'95*, Nordby, K., Helmersen, P.H., Gilmore, D.J., Arnesen, S.A. (Eds), Chapman and Hall Press, 1995, pp 173-178.
3. Lachenal, C., Coutaz, J. *A Reference Framework for Multi-surface Interaction*. HCII'2003 (Human computer Interaction International) Crete.,
4. McCall J., *Factors in Software Quality*, General Electric Ed., 1977.
5. Nielsen J., *Usability Engineering*, Academic Press Professional, 1993, p 362.

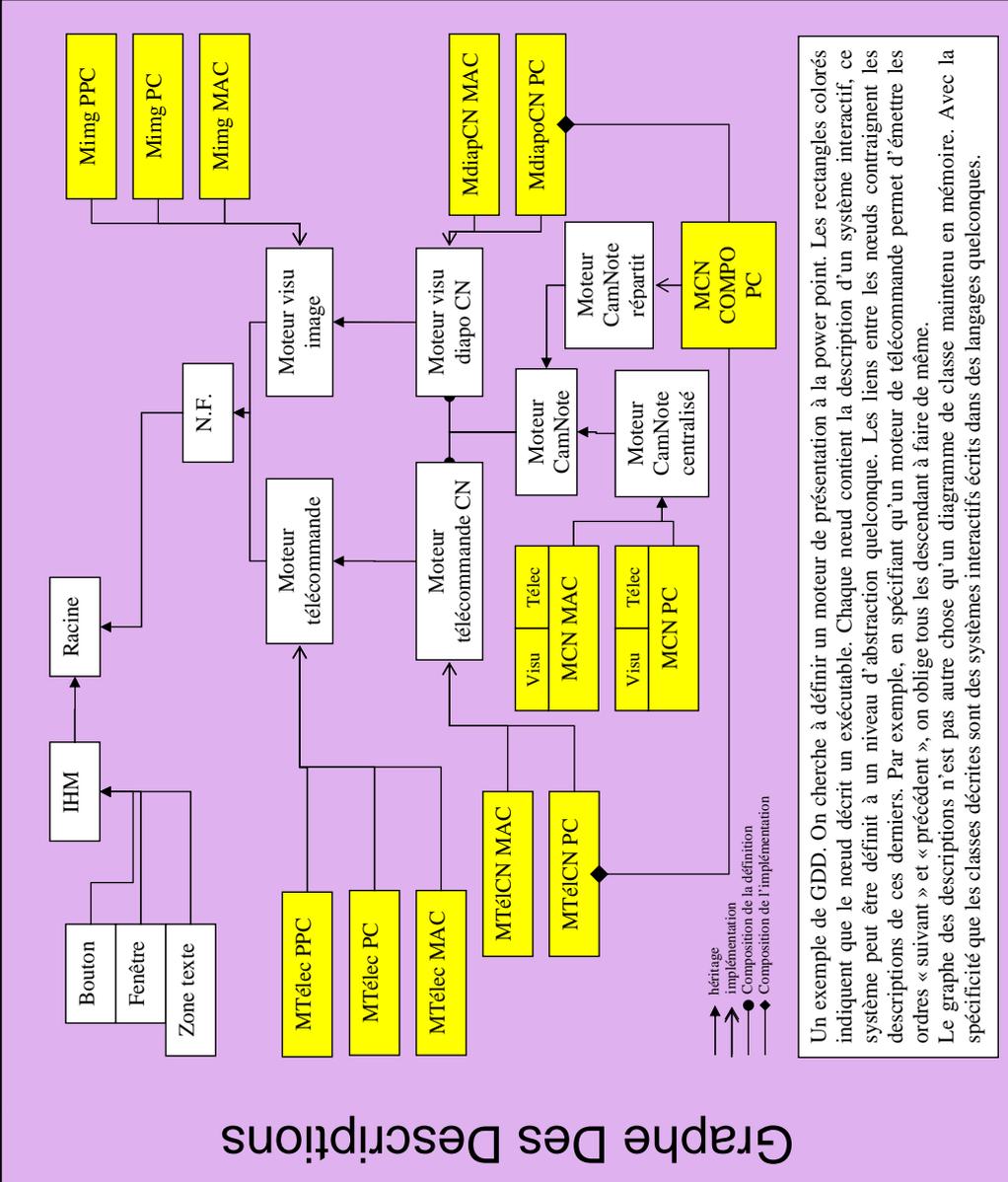
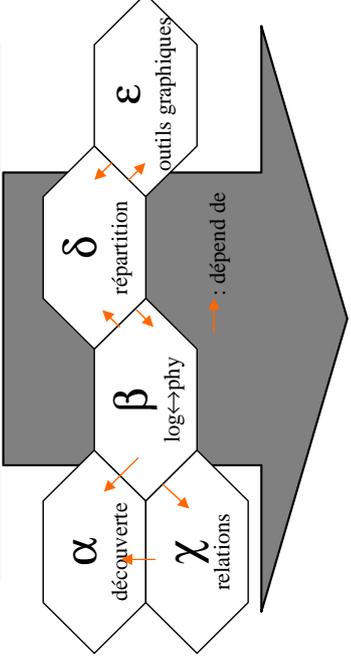


Un exemple de GDD. On cherche à définir un moteur de présentation à la power point. Les rectangles colorés indiquent que le nœud décrit un exécutable. Chaque nœud contient la description d'un système interactif, ce système peut être défini à un niveau d'abstraction quelconque. Les liens entre les nœuds contraignent les descriptions de ces derniers. Par exemple, en spécifiant qu'un moteur de télécommande permet d'émettre les ordres « suivant » et « précédent », on oblige tous les descendant à faire de même. Le graphe des descriptions n'est pas autre chose qu'un diagramme de classe maintenu en mémoire. Avec la spécificité que les classes décrites sont des systèmes interactifs écrits dans des langages quelconques.

figure 3 : Un exemple de graphe des descriptions.



Système interactif (CamNote PC)



AMUSINg : Outil d'Assemblage et d'Adaptation d'IHM

Anne-Marie Dery-Pinna

Rainbow project, I3S
930, route des Colles
BP 145
06903 Sophia-Antipolis
(+33/0)4 92 96 51 62
pinna@essi.fr

Jérémy Fierstone

Open-Plug & Rainbow, I3S
2600, route des crêtes
BP 176
06903 Sophia-Antipolis
(+33/0)4 97 24 50 68
fierston@essi.fr

RESUME

Dans ce papier nous présentons notre environnement de développement d'IHM basé sur des composants. L'IHM est considérée comme un service technique d'un composant métier à l'instar de la sécurité ou de la persistance. Le lien entre l'IHM et le composant métier est géré par un service d'interaction/coordination qui permet la reconfiguration des composants sans les modifier. Un service de fusion de composants d'IHM permet l'assemblage dynamique des composants d'IHM associés.

MOTS CLES : Assemblage, composition, fusion, composant, interaction, sémantique, vue abstraite, vue concrète.

ABSTRACT

In this paper we present our UI development environment based on components. The UI is considered as a technical service of a business component just like security or persistence. The dialog between UI and business components is managed by an interaction/coordination service that allows the reconfiguration of components without modifying them. A UI component merging service handles dynamic assembly of corresponding UI components.

CATEGORIES AND SUBJECT DESCRIPTORS: D.2.2 [SOFTWARE ENGINEERING]: Design Tools and Techniques – user interfaces.

GENERAL TERMS: Algorithms, Design, Languages.

KEYWORDS : Assembly, composition, merging, component, interaction, semantic, abstract view, concrete view.

INTRODUCTION

L'objectif de ce papier est de montrer comment les propriétés de la programmation par composants [6] (assemblage de composants, ajout et retrait dynamique de composants, interactions entre composants métiers et techniques) peuvent être exploitées pour faciliter le développement d'applications adaptables aux besoins des utilisateurs mobiles. L'idée est de faciliter, par exemple, la capacité de charger ou de décharger dynamiquement des

fonctionnalités en fonction du contexte d'utilisation tout en conservant une IHM adaptée à la situation. L'originalité de notre solution est de considérer l'IHM comme un assemblage de composants d'IHM et un composant d'IHM comme un composant technique au service d'un composant métier de la même façon que la prise en charge de la sécurité et de la persistance [2]. Cette solution permet de changer dynamiquement le composant d'IHM afin de l'adapter à un nouveau support ou à un nouveau contexte d'utilisation. De plus, l'IHM peut évoluer en fonction du contexte d'exécution d'une application par ajout et retrait de composants d'IHM correspondants aux évolutions fonctionnelles. Cependant, pour conserver les propriétés d'assemblage de composants dans le cadre d'assemblage d'IHM, il est indispensable de mettre en place des règles de composition d'IHM très spécifiques. Dans cet article, nous ne détaillerons pas la façon de composer les composants métiers et techniques [4], nous nous focaliserons sur l'assemblage dynamique des composants d'IHM qui leur sont associés. Aussi la partie suivante illustre-t-elle le type d'application visée, puis des règles de compositions statiques et dynamiques de composants d'interface sont développées par la suite.

ILLUSTRATION DES BESOINS PAR L'EXEMPLE

Prenons comme illustration le scénario d'un fournisseur d'applications de gestion construites pour répondre aux besoins spécifiques de chaque client. Notons en particulier, le cas du commercial qui intervient souvent sur le site de l'acheteur et dont les tâches peuvent dynamiquement évoluer en fonction des besoins de ce dernier. Cet exemple est issu d'une étude de cas réelle menée dans le cadre d'un contrat RNTL avec des partenaires industriels. L'assemblage de composants métiers (imposé par les applications ASP de ce type) permet de rapidement construire l'application demandée à partir d'une bibliothèque de composants prédéfinis (tels que Customer, Salesperson, Bill, et Stock). A la conception, plusieurs composants sont assemblés statiquement dans un objectif fonctionnel précis. Par exemple, le commercial qui part en déplacement peut charger sur son assistant personnel les composants Catalog et Order. A l'exécution, un nouvel assemblage peut s'avérer utile pour ajouter dynamiquement un composant nécessaire pour répondre à un

contexte d'exécution particulier. Par exemple, le commercial a besoin de connaître l'email d'un client particulier pour annuler un rendez-vous, il demande alors le chargement sur son terminal du composant Address (contenant les coordonnées d'un client) qui doit être dynamiquement assemblé avec le composant Customer afin d'enrichir l'application de base et ainsi associer à un client ses coordonnées. De la même façon, il doit pouvoir supprimer certaines fonctionnalités pour alléger la mémoire si nécessaire. L'IHM doit s'adapter à ce type d'évolutions fonctionnelles d'une application, aussi bien statiquement que dynamiquement. Si les modèles à composants classiques (EJB, CCM, .Net) permettent actuellement tous les assemblages statiques, voire le chargement dynamique de composants liés statiquement (c'est-à-dire au moment de l'utilisation des composants) tels que les travaux sur les DLL, seuls des travaux de recherche sur les composants tels que [1,2] permettent l'ajout et le retrait dynamique de composants non prévus à la conception. Les modèles à services tels que OSGi permettent la découverte de nouveaux services dynamiquement mais ne possèdent pas les propriétés d'assemblage des modèles à composants. Actuellement les travaux de recherche sur l'adaptabilité des IHMs concernent principalement la plasticité et la prise en compte du contexte [3]. Ils n'abordent pas le problème en terme de fusion de composants d'interfaces. En appliquant les technologies et propriétés inhérentes à l'assemblage de composants métiers à la construction d'IHM, nous devons obtenir une meilleure adaptation et réutilisation des IHM existantes.

Notre proposition se base sur un service d'interactions [2] qui prépare les composants pour interagir entre eux et un serveur d'interactions qui permet de définir des schémas d'interactions réutilisables pour permettre la liaison et la déliaison de composants métiers. L'originalité de cette proposition est de considérer l'IHM comme une propriété orthogonale (un composant technique associé au composant métier) et d'ainsi pouvoir éclater l'IHM globale en plusieurs composants d'IHM qui peuvent être fusionnés.

Dans la suite nous développons la spécificité de ces composants d'IHM et les algorithmes de fusion que nous avons mis en place afin de permettre l'ajout statique et dynamique de composants d'IHM à une IHM existante. Les aspects concernant la mise en œuvre d'interactions et du serveur d'interactions dans le cadre d'applications mobiles ne sont pas abordés dans cet article et font l'objet de travaux spécifiques du projet RAINBOW.

COMPOSANTS D'IHM FUSIONNABLES

Dans cette partie nous détaillons le cœur de l'environnement de développement d'IHM, Amusing [19]. La première section décrit les composants d'IHM et

les trois sections suivantes les différentes mises en œuvre de la fusion de tels composants.

Qu'Appelle-t-on un Composant d'IHM ?

Dans notre modèle de composants, un composant métier peut être lié à un ou plusieurs composant(s) d'IHM à l'aide d'interactions logicielles. Une interaction représente le contrôleur de communication (comme dans le modèle Arch). L'environnement de développement réalisé comprend un serveur d'interactions permettant d'enregistrer et d'utiliser des schémas d'interaction / coordination pour assembler des composants métiers et pour les faire interagir avec leurs composants techniques. Les schémas d'interaction sont spécifiés en ISL [2].

Afin de garantir une indépendance du support lors de l'assemblage, le composant d'IHM doit être décrit de manière abstraite. Nous proposons dans notre modèle un mini langage de spécification d'IHM indépendant des plates-formes cibles, SUNML (pour *Simple Unified Natural Markup Language*) détaillé en 3.3, dans la lignée XML. Un composant d'IHM est décrit dans ce langage puis instancié sous forme d'arbre abstrait représentant la structure abstraite de l'IHM [7]. Cette vue abstraite peut être ensuite projetée vers une ou plusieurs vues concrètes à l'aide d'un renderer (actuellement Swing ou Vocal). La vue abstraite et la ou les vues concrètes sont liées entre elles. La vue abstraite est utilisée, dans notre architecture, comme un méta objet représentant à la fois la structure abstraite et conservant les données des IHM concrètes (valeurs saisies par l'utilisateur) qui doivent être synchronisées en permanence avec cette vue abstraite. Cette indirection supplémentaire apporte la flexibilité utile à la composition des IHM ainsi qu'à leur mobilité. La vue abstraite est liée au composant métier qui ne suppose rien de la manière dont est projeté un widget abstrait (WA¹). Il est également possible de lier directement le composant métier à la vue concrète. Un tel lien direct peut être utilisé pour optimiser l'exécution lorsqu'un assemblage est figé lors de la conception. Mais dans ce cas, adaptation et assemblage dynamiques ne sont plus possibles. Le schéma suivant présente les éléments de base d'un composant d'IHM obtenu par réification de la description SUNML et projection de l'arbre abstrait :

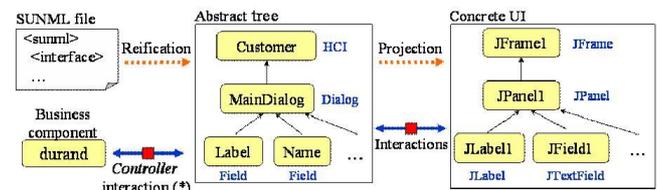


Figure 1. Architecture visée

¹ Dans notre terminologie, le terme *widget abstrait* correspond au terme *objet d'interaction abstrait* dans le domaine des IHM.

Les Algorithmes de Fusion

La fusion de composants d'interface est réalisée à partir des vues abstraites et permet d'obtenir une nouvelle vue abstraite. Comme toute vue abstraite, celle-ci peut être utilisée dans une autre fusion ou pour une projection. L'opérateur *add* inclut un composant d'IHM dans un autre composant d'IHM par ajout d'un arbre de WA dans un autre. L'opérateur *select* permet de n'extraire d'un composant d'IHM qu'une sous partie en récupérant un sous arbre à partir de son chemin (comme dans une arborescence de fichiers). L'opérateur *union* plus précis que l'ajout permet une fusion de deux composants d'IHM en supprimant les redondances d'information. L'union fusionne des arbres en unifiant les abstraits sémantiquement identiques. Dans la Figure 2 le nom du client a été identifié comme partie commune pouvant être représentée par un seul WA. A l'inverse, l'opérateur *intersect* permet de récupérer dans plusieurs composants d'IHM la partie commune. Il retourne l'ensemble des WA sémantiquement identiques. Dans la figure 3, les WA correspondant respectivement au widget concret (WC) *nom* et *name* sont identifiés comme devant être un WA commun.

Bien que ces algorithmes s'appliquent à des arbres, ceux-ci ne se simplifient pas en simples algorithmes de fusion d'arbres ordonnés. En effet, pour déterminer l'identité entre deux WA (éléments de l'arbre), nous avons introduit la notion de sémantique d'un WA, spécifiée sous forme d'un attribut *semantic* décorant l'arbre abstrait. Actuellement cet attribut est spécifié par le développeur et éventuellement modifiable dynamiquement avant fusion afin de spécifier une équivalence sémantique entre des WA hétérogènes (issus de développeurs différents, dans des langues différentes, etc.). L'automatisation de ces algorithmes pour des applications à grande échelle nécessite un travail en amont pour pouvoir en partie générer les attributs « semantic » correctement. Les pistes envisagées sont de partir d'ontologies associées à des domaines d'applications spécifiques et plus simplement, de la cohérence avec les composants métiers. Sur ce dernier point, cet attribut pourrait être généré par défaut à partir de la description fonctionnelle du composant métier (représentant les mêmes informations au niveau du composant métier)...

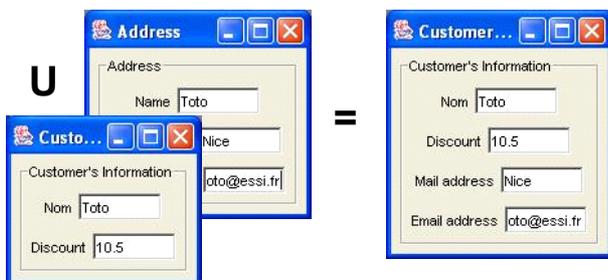


Figure 2. Fusion par l'opérateur d'union

Les opérations de fusion ne respectent pas les propriétés de commutativité et d'associativité. Deux WA déclarés

comme étant sémantiquement identiques ne contiennent pas forcément les mêmes données (modifiables par l'utilisateur) ni les mêmes caractéristiques (un élément d'un menu peut être sémantiquement identique à un bouton dans une autre IHM). Un choix entre ces deux instances de WA étant nécessaire, dans les algorithmes, l'instance correspondant au premier arbre en argument est retournée. Par exemple, les instances de WA correspondant aux WC *nom* et *name* (cf. Figures 2 et 3) ont la même sémantique, l'union et l'intersection privilégiant le premier argument donnent l'instance *nom* comme résultat. Les résultats des fusions dépendant fortement de l'ordre des composants à fusionner, la propriété ensembliste $A + B = (A \cup B) + (A \cap B)$ se réécrit par la propriété suivante que respectent les opérateurs de fusion :

$$A + B = (A \cup B) + (B \bullet A)$$

La fusion dynamique des IHM est possible en utilisant directement l'API d'Amusing ou en spécifiant la fusion à l'aide d'un langage de fusion, WML (pour Widget Merging Language) basé sur XML. Ce langage permet de spécifier la manière de fusionner des composants d'IHM, donnés en argument d'entrée, et de retourner le résultat (comme une méthode).

Les Interactions au Service des Fusions Dynamiques des Vues Abstraites

Lorsqu'un nouveau composant métier est ajouté dynamiquement dans l'application, les composants d'IHM correspondants doivent être également ajoutés dynamiquement à l'IHM en cours d'exécution. Il est alors capital de gérer l'assemblage structurel dynamiquement en utilisant la représentation abstraite de l'IHM. La fusion (cf. 3.2) peut gérer l'assemblage dynamique par manipulation des vues abstraites. Ce service de fusion peut être invoqué par l'utilisateur à l'exécution. De plus, afin d'étendre l'utilisabilité de ce service, on peut utiliser les interactions logicielles afin d'automatiser l'ajout d'une nouvelle vue abstraite lors de l'ajout d'un composant. Ainsi, l'interaction invoque dynamiquement le service de fusion avec les règles de fusion prédéfinies. La règle ISL suivante illustre cette fusion automatique avec le cas typique d'un composant, représentant statiquement une liste d'éléments quelconques, adaptée dynamiquement par cette règle pouvoir recevoir dynamiquement un composant Customer :

```
interaction CustomerListMerging(AUI customerList) implements MergingService {
    customerList.add(AUI customer) ->
    customerList.add(customer.select("*/name",
    "*/discount"))
}
```

Dans cet exemple, tout composant *Customer* ajouté dans la liste est fusionné avec la structure abstraite de la liste en utilisant l'opérateur *select*. Le comportement du com-

posant doit être également fusionné avec les comportements des autres composants grâce aux interactions logicielles. En effet, alors que les algorithmes de fusion ne proposent qu'une fusion structurelle des IHM, leurs comportements dépendent de cette fusion. Dans cet exemple, on voudrait également fusionner la sauvegarde de tous les clients lors de la sauvegarde de la liste (bouton *save*). Cette fusion comportementale n'est possible qu'après la fusion structurelle et peut être réalisée à l'aide du schéma d'interaction suivant :

```
interaction CustomerListMerging(AUI
list,
    sequence<AbstractUI> customers) {
list.manageEvent(AbstractEvent e) ->
    if(e.getDesc() == "save") {
        foreach(customers, customer) {
            customer.manageEvent(e)
        }
    }
}
```

Chaque composant *Customer* fusionné avec le composant *List* doit être lié par cette interaction qui stipule que chaque réception du message *manageEvent* inhérent à la réception d'un événement concret résultant du click sur le bouton *save*, doit correspondre au nouveau comportement spécifié en partie droite de la règle, en l'occurrence l'envoi de ce même message sur chaque composant client (dans notre exemple, l'IHM du composant *Customer* appelle la méthode de sauvegarde lors de sa fermeture).

CONCLUSION

Le travail présenté dans cet article est intégré dans un environnement de développement d'IHM ouvert Amusing [28]. Avec cet environnement, nous avons déjà développé plusieurs applications à base de composants : un jeu de bataille navale adaptable aux utilisateurs (voyants et non voyants) et une application de gestion pour commerciaux mobiles. Ces expérimentations nous ont permis de valider les algorithmes de fusion. De premières expérimentations prometteuses ont été faites pour automatiser en partie des adaptations à partir de graphes d'états de contextes [9], cependant cette approche expérimentale doit être poursuivie plus avant. Nous travaillons également à l'intégration d'un WYSIWYG pour assembler des composants de manière visuelle afin d'aider la personne en charge de l'assemblage des composants ou services de fusionner les composants en lui permettant de spécifier les WA qu'il considère communs par des relations d'équivalence sémantique. Une des principales difficultés de ce travail est d'avoir une représentation pertinente de l'arbre abstrait.

Si l'adaptation aux composants métiers paraît actuellement facilement atteignable, un autre de nos objectifs étant l'adaptation aux supports est en cours d'étude. Même si l'environnement intègre des renderers, il n'intègre actuellement aucun moteur d'adaptation et ne prend pas en compte la plasticité des interfaces. Une

première approche envisagée est de se connecter à un moteur d'adaptation existant en traduisant l'arbre abstrait résultant de la fusion sous forme d'arbres abstraits compris par le moteur cible (RIML, par exemple). SUNML étant un sous langage à balise, cette traduction doit être simple. Des travaux de recherche plus complets doivent être envisagés pour déterminer comment les règles de fusion proposées peuvent conserver ou non la plasticité des interfaces et sous quelles limites (gamme de supports...).

REMERCIEMENTS

Ce travail nécessite une expertise importante en terme de besoins applicatifs, nous tenons à remercier les partenaires du RNTL Aspect pour leur apport sur ce point.

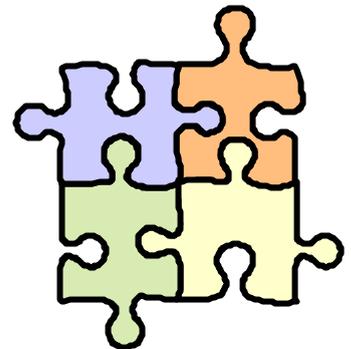
REFERENCES

1. L. Berger. Mise en oeuvre des interactions en environnements distribués, compilés et fortement typés: le modèle MICADO. PhD thesis, Université de Nice-Sophia Antipolis, octobre 2001.
2. M. Blay-Fornarino, D. Emsellem, A-M. Pinna-Dery, and M. Riveill. Un service d'interactions : principes et implémentation. Revue TSI, 2004.
3. Calvary, J. Coutaz, D. Thevenin. A Unifying Reference Framework for the Development of Plastic User Interfaces. IFIP WG2.7 (13.2) Working Conference, EHCI01, Toronto, May 2001, Springer Verlag Publ., LNCS 2254, M. Reed Little, L. Nigay Eds, pp.173-192.
4. Nano, M. Blay-Fornarino, A-M. Dery, and M. Riveill. An abstract model for integrating and composing services in component platforms. In ECOOP'2002, Malaga, June 10, 2002.
5. A-M. Pinna-Dery, J. Fierstone, M. Riveill, E. Picard. User Interface: a Technical Component in Component-Based Models in Response to Human Computer Interaction Adaptation. Rapport de recherche, février 2004.
6. M. Riveill and P. Merle. La programmation par composants. In Techniques de l'Ingénieur - Informatique, H2759, décembre 2000.
7. D. Thevenin. La plasticité des Interfaces Homme Machine : une approche, un outil. Computer Science Ph.D Thesis, Joseph Fourier University - Grenoble I, 2001, 212 p.
8. D. Thevenin. « Adaptation en Interaction Homme-Machine: Le cas de la Plasticité ». Ph.D. thesis, Université Joseph Fourier, Grenoble, 21 December 2001.
9. Dey, A.K., Salber, D. Abowd, G.D. A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications in Human-Computer Interaction (HCI) Journal, Vol. 16(2-4), 2001, pp. 97-166.

AMUSING : Adaptable & Mergeable USeR INterface

Objectives

- Provide an open-architecture for
 - Representing abstract HMI with **SUNML**
 - Merging UI components with **Fusion Algorithms**
 - Dynamic Merging with **WML**
- Provide a platform and API for
 - Projecting abstract UI into concrete UI
 - Merging abstract UI components
 - Managing adaptation of UI



Concepts

- Merging example:
 - Union algorithm : $(A \cup B) = (A + B) - (B \cap A)$
 - Based on the **semantic** value
 - **semantic** (Customer.name) == **semantic** (Address.name)



SunML example

```
<sunml>
<interface id="Customer" desc="Customer's Information" semantic="interface">
  <dialog id="MainDialog" desc="Customer's Information" semantic="dialog">
    <element id="name" mode="read-write" value="Toto" />
    <element id="discount" type="double" mode="read-write" value="10.5" />
  </dialog>
</interface>
</sunml>
```

Contact

- Rainbow team (I3S), <http://rainbow.essi.fr>
 - Anne-Marie Dery-Pinna : professor at ESSI, pinna@essi.fr
 - Jeremy Fierstone : PhD. Open-Plug & Rainbow project, fierston@essi.fr
 - Michel Riveill : professor at ESSI and team manager, riveill@essi.fr

Modélisation et vérification formelle des IHM multimodales

Nadjet KAMEL^{†*}

[†] LRIA/USTHB

BP 32,

El Alia, Bab Ezzouar, 16111, Alger, Algerie

^{*} LISI/ENSMA

BP 40109

86961, Futuroscope Cedex, France

nkamel@wissal.dz, kamel@ensma.fr

RESUME

Les techniques formelles basées sur les modèles mathématiques permettent la spécification sans ambiguïté et la vérification des systèmes à un stade avancé dans le cycle de développement. Peu de travaux appliquant ces techniques dans le développement des IHM multimodales (IHM3) existent. Dans cet article nous présentons une partie des travaux de notre thèse qui consiste en la définition de modèles et démarches formels pour le développement des IHM3. Nous présentons les premiers résultats concernant la modélisation formelle de l'IHM3.

MOTS CLES : Interactions Multimodales, Méthodes Formelles

ABSTRACT

Formal methods based on mathematical models allow specification and verification of systems at earlier step in the development cycle. Few works are interested to apply these techniques in the development of multimodal HCI. In this paper we present a part of our thesis work which consist of definition of models and formal approaches for developing IHM3. We present our first results concerning the formal modelisation of multimodal HCI.

CATEGORIES AND SUBJECT DESCRIPTORS: D.2 [Software Engineering] : Model Checking.

GENERAL TERMS: Verification, Experimentation

KEYWORDS : Multimodal Interaction, Formal Methods

INTRODUCTION

Avec la puissance des machines et le progrès des systèmes de reconnaissance de la parole et de synthèse vocale qui ont permis d'augmenter les capacités sensorimotrices et représentationnelles des systèmes, nous assistons au développement d'applications où les moyens de communication et d'interaction sont de plus en plus variés. Contrairement aux interfaces classiques, dans les interfaces multimodales l'utilisateur peut interagir en utilisant un ou plusieurs moyens de communication, avec

l'application, tels que la parole, le geste et la manipulation directe etc. Dans ce type d'interfaces, les interactions peuvent être utilisées de manières séquentielles, parallèles, indépendantes ou bien combinées. Les IHM3 sont caractérisées également par la concurrence entre les événements et tâches. Les travaux de recherche dans ce domaine s'intéressent à plusieurs aspects. Certains s'intéressent à l'implémentation d'applications multimodales [5][7][2], d'autres à la définition de concepts relatifs à la multimodalité [8] et d'autres aux propriétés d'interfaces multimodales [4].

Les méthodes formelles, par l'utilisation de modèles et de techniques reposant sur des définitions mathématiques rigoureuses, offrent de nouvelles possibilités pour la validation des logiciels. Leur mise en application permet de garantir des propriétés fondamentales du système, avec pour résultat concret d'éliminer plus d'erreurs dès la conception. Ceci évitera les testes effectuées en fin du cycle du développement et qui provoquent en général un coût élevé de maintenance. L'utilisation des techniques formelles dans le domaine des systèmes interactifs, que représente les IHM, a intéressé plusieurs travaux de recherche, mais leur utilisation pour les IHM3 reste rare et c'est dans ce contexte que se situe notre travail. Notre objectif est la modélisation formelle de tout type d'interface multimodale, la modélisation des propriétés et l'utilisation de techniques de vérification formelle. Les modèles que nous définissons sont indépendants de toute technique formelle et de tout outil. Les premiers résultats que nous présentons dans cet article consiste en la définition d'un modèle formel générique permettant la conception d'interfaces multimodales en se basant sur les espaces de conception déjà définis par les travaux de recherches dans ce domaine. La sémantique formelle de ce modèle est donnée en termes de systèmes de transitions étiquetées. Pour la vérification formelle des propriétés, il existe principalement deux techniques formelles : la vérification sur modèle que nous avons mise en œuvre dans [11] avec SMV[8] pour vérifier des propriétés CARE [4] et la technique de preuve qui en cours d'étude en utilisant l'atelier B.

Cet article est organisé comme suit : la prochaine section présente le contexte de notre étude, où nous présentons l'espace de conception que nous avons utilisé pour définir notre modèle formel qui sera présenté dans la section 3. Dans la section 4, nous présentons les techniques formelles de vérifications que nous avons mises en œuvre pour la vérification des propriétés d'IHM3. La dernière section présente la conclusion et nos futurs travaux.

CONTEXTE D'ETUDE

L'objectif de notre travail est de définir des modèles formels pour assister le développement d'IHM3 au niveau conception et vérification de propriétés d'utilisateur. Pour cela nous nous sommes intéressés d'abord aux travaux déjà réalisés dans le domaine en terme d'espace de conception et de propriétés d'utilisateur d'IHM3. L'interaction dans les IHM3 est complexe dans le sens où elle est définie par plusieurs événements provenant de plusieurs canaux d'entrée. Plusieurs modalités peuvent participer à la réalisation d'une interaction élémentaire : un *énoncé*. Ces énoncés participent à la réalisation d'interactions utilisateurs plus complexes : *tâches*. Les énoncés et les tâches peuvent être produits d'une manière parallèle ou séquentielle. Un énoncé peut être produit par une ou plusieurs modalités de manière séquentielle ou parallèle. La combinaison de tous ces critères et autres définit les espaces de conception. Notre modélisation repose sur l'espace défini dans [1], et qui repose sur les trois critères suivants : production des énoncés (séquentiel ou parallèle), usage des modalités (exclusif ou simultané) et nombre de modalités par énoncé. La combinaison de ces critères a produit plusieurs types de multimodalité. Nous présentons ici à titre d'exemple trois types :

1. *Alternée* : plusieurs modalités peuvent être utilisées alternativement pour produire un énoncé. Les énoncés sont produits de manière séquentielle.
2. *Synergique* : les énoncés sont produits de manière séquentielle mais plusieurs médias peuvent être utilisés dans un même énoncé et de manière parallèle.
3. *Parallèle exclusive* : plusieurs énoncés indépendants peuvent être produits en parallèle. Un seul média est utilisé pour chaque énoncé et à un instant donné un seul média est actif.

Pour vérifier l'utilisabilité d'une IHM3, un ensemble de propriétés a été défini. Ce sont les propriétés CARE : Complémentarité, Assignment, Redondance, Equivalence. Elles définissent les propriétés de flexibilité et de robustesse des IHM3. Deux modalités sont équivalentes si chacune d'elle permet de réaliser la même tâche. Le choix offert par cette propriété définit la flexibilité de l'IHM3. Deux modalités sont complémentaires si elles participent toutes les deux dans la réalisation d'une tâche. L'assignment désigne la spécialisation d'une modalité pour la réalisation d'une tâche tandis que la redondance exprime l'utilisation de modalités équivalentes

pour la même tâche de manière redondante. La redondance augmente la robustesse d'un système. Une syntaxe formelle de ces propriétés a été donnée dans [4]. Elle se base sur les notions d'*états*, *but*, *modalité* et *relations temporelles*

MODELE FORMEL POUR L'IHM3

Nous présentons, dans cette section, notre modèle formel pour la conception l'interaction multimodale en entrée. Nous définissons la syntaxe du modèle par des règles de grammaire et la sémantique dynamique par les systèmes de transitions étiquetés. Nous avons défini un ensemble d'opérateurs de composition empruntés des algèbres de processus Lotos et CCS. Le modèle est paramétré de sorte qu'il permet de définir tout type d'IHM3 selon l'espace défini dans la section 2. Dans la sous section suivante, nous définissons les notions de modalité et événement utilisés dans notre modèle.

Modalités et événements

Nous définissons une modalité par un type d'événements. Un événement est une interaction élémentaire produite par une modalité et qui est non décomposable. Nous notons par A_{mi} , l'ensemble des événements ei produits par la modalité mi . A est l'ensemble des événements e du système. Ainsi si on dispose de n modalités,

on pose $A = \bigcup_{i=1}^n A_{mi}$. Enfin, AM désigne l'ensemble

des événements générés par un sous-ensemble de modalités possibles. $AM = \bigcup_{k \in 1..n} A_{mk}$ avec $A_{mk} \subseteq A$.

Syntaxe

Nous donnons ici la syntaxe des modèles de quelques types d'IHM3 sous forme de grammaire. Pour chacune des grammaires la règle T est utilisée pour générer le langage en fonction des tâches utilisateurs à un niveau élevé d'abstraction. La règle E définit les énoncés par la composition des événements élémentaires. Les énoncés sont paramétrés par les modalités utilisées pour leur réalisation. La signification informelle des opérateurs de composition utilisés dans la syntaxe des modèles est donnée comme suit :

δ est un terme qui ne fait rien (mot vide) ;

$[]$ est le choix non déterministe entre deux tâches ;

\gg est l'opérateur de séquence entre deux tâches ;

$;$ est le préfixage ;

\parallel désigne l'opérateur d'entrelacé ;

$\|$ est l'opérateur du parallélisme.

Type Alterné. Les énoncés sont composés séquentiellement et produits par des événements issus de différentes modalités entrelacées.

$$T ::= T [] T \mid T \gg T \mid E_{AM}$$

$$E_{AM} ::= e ; E_{AM} \mid e \parallel E_{AM} \mid \delta \text{ avec } e \in AM$$

Type Synergique. Les énoncés sont composés de manière séquentielle et peuvent être produits par plusieurs modalités en parallèle ou entrelacées.

$$T ::= T [] T \mid T \gg T \mid E_{AM}$$

$$E_{AM} ::= e ; E_{AM} \mid e \parallel E_{AM} \mid e \parallel\parallel E_{AM} \mid \delta$$

avec $e \in AM$

Type Parallèle Exclusif. Chaque énoncé est composé d'évènements issus d'une seule modalité.

$$T ::= T [] T \mid T \gg T \mid T \parallel\parallel T \mid E_{Amk}$$

avec $A_{mk} \subseteq AM$

$$E_{Ami} ::= ei ; E_{Ami} \mid \delta \text{ avec } ei \in A_{mi}$$

Sémantique

Sémantique statique. La sémantique statique définit des attributs aux éléments syntaxiques afin de pouvoir exprimer certaines propriétés statiques de l'IHM3. Ces attributs peuvent être codés dans la technique formelle utilisée pour implémenter le modèle. Ces attributs peuvent caractériser le type d'évènement, sa durée etc.

Sémantique dynamique. La sémantique dynamique ou opérationnelle définit le comportement des différents opérateurs de composition pendant l'exécution du système. Elle est donnée par des règles de transition définissant le système de transitions sous-jacent. Soient P, Q des termes de la grammaire, e, e_1 et e_2 des évènements de A . La transition $P \xrightarrow{e} Q$ décrit le passage d'un état caractérisé par P à un état caractérisé par Q , lorsque l'évènement e est déclenché. La sémantique des opérateurs est donnée par les relations de transitions décrites selon le style de G.Plotkin [10] au moyen de règles de la forme $\frac{\text{prémisses}}{\text{conclusion}}$. Elles sont définies par deux axiomes et neuf règles.

$$\delta : \quad \delta \not\rightarrow \quad \text{axiome}$$

$$; : \quad e ; P \xrightarrow{e} P \quad \text{axiome}$$

$$[] : \quad \frac{P \xrightarrow{e} P'}{P [] Q \xrightarrow{e} P'} \quad \frac{Q \xrightarrow{e} Q'}{P [] Q \xrightarrow{e} Q'}$$

$$\gg : \quad \frac{P \xrightarrow{e} P' \text{ et } P' \neq \delta}{P \gg Q \xrightarrow{e} P' \gg Q}$$

$$\quad \frac{P \xrightarrow{e} P' \text{ et } P' = \delta}{P \gg Q \xrightarrow{e} Q}$$

$$\parallel\parallel : \quad \frac{P \xrightarrow{e} P'}{P \parallel\parallel Q \xrightarrow{e} P' \parallel\parallel Q} \quad \frac{Q \xrightarrow{e} Q'}{P \parallel\parallel Q \xrightarrow{e} P \parallel\parallel Q'}$$

$$\parallel : \quad \frac{P \xrightarrow{e} P'}{P \parallel Q \xrightarrow{e} P' \parallel Q} \quad \frac{Q \xrightarrow{e} Q'}{P \parallel Q \xrightarrow{e} P \parallel Q'}$$

$$\frac{P \xrightarrow{e_1} P' \text{ et } Q \xrightarrow{e_2} Q' \text{ avec } e_1 \in A_{mi}, e_2 \in A_{mj} \text{ et } A_{mi} \neq A_{mj}}{P \parallel Q \xrightarrow{(e_1, e_2)} P' \parallel Q'}$$

Le système de transitions d'un terme (mot) du langage généré par la grammaire est obtenu par induction structurelle à partir des règles précédentes. C'est sur ce système de transitions que seront vérifiées les propriétés dynamiques du système.

Vérification formelle des propriétés

Les propriétés d'utilisabilité CARE peuvent être spécifiées formellement dans une logique pour être vérifiées sur le modèle du système. Il existe deux techniques formelles pour la vérification et la validation des systèmes : La vérification sur modèle et la preuve. La vérification sur modèle est une technique basée sur la construction d'un modèle fini du système et la vérification des propriétés sur ce modèle. La vérification est effectuée par énumération exhaustive de l'espace d'états. Une approche de cette technique a été développée par Clarke et Emerson [3] et Quille et Sifakis [9]. Elle consiste à exprimer les propriétés dans une logique temporelle [5] et modéliser le système par un système de transitions fini. Un algorithme est utilisé pour vérifier si le système de transitions est un modèle pour la spécification. Dans [11] nous avons utilisée cette technique pour vérifier des propriétés CARE à l'aide du vérificateur sur modèle SMV[6]. Les propriétés ont été spécifiées dans la logique temporelle CTL et le système de l'IHM3 a été modélisé par notre modèle présenté dans la section précédente. Le système de transitions est décrit dans le langage d'entrée de SMV. La technique de preuve permet de vérifier des propriétés exprimées dans la logique de premier ordre sous forme d'invariants à l'aide d'un système d'inférence. Une implémentation avec cette technique est en cours d'étude en utilisant le B évènementiel pour décrire le système de transitions du modèle du système, les invariants pour décrire les propriétés et l'atelier B pour la preuve.

CONCLUSION

Dans cet article, nous avons présenté notre contribution pour la conception et la vérification des IHM3. Notre démarche est générique est indépendante de tout outil. Nous avons défini un modèle formel pour la conception de l'IHM3 selon un espace de conception défini par les chercheurs de ce domaine. Nous avons présenté également les deux démarches de vérification des propriétés d'utilisabilité d'une IHM3 que nous avons mise en oeuvre.

Les travaux en cours concernent la modélisation formelle des propriétés CARE ainsi qu'à l'utilisation de l'approche descendante avec un développement par décomposition en utilisant le raffinement et la technique de preuve pour la vérification des propriétés en utilisant l'atelier B et le B évènementiel.

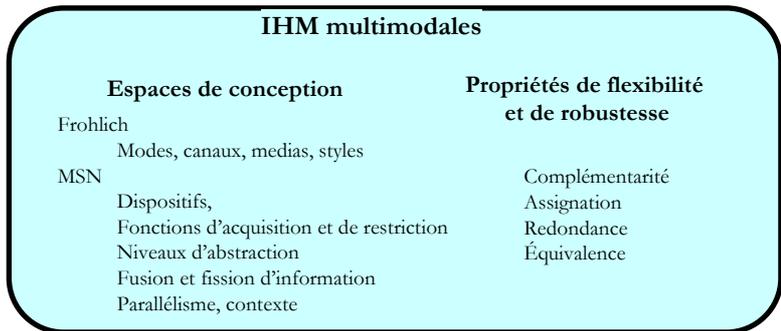
Les travaux futurs s'intéressent à la proposition d'une approche globale combinant les deux techniques de conception et de vérification afin d'exploiter les avantages de chacune dans un même processus de développement.

BIBLIOGRAPHIE

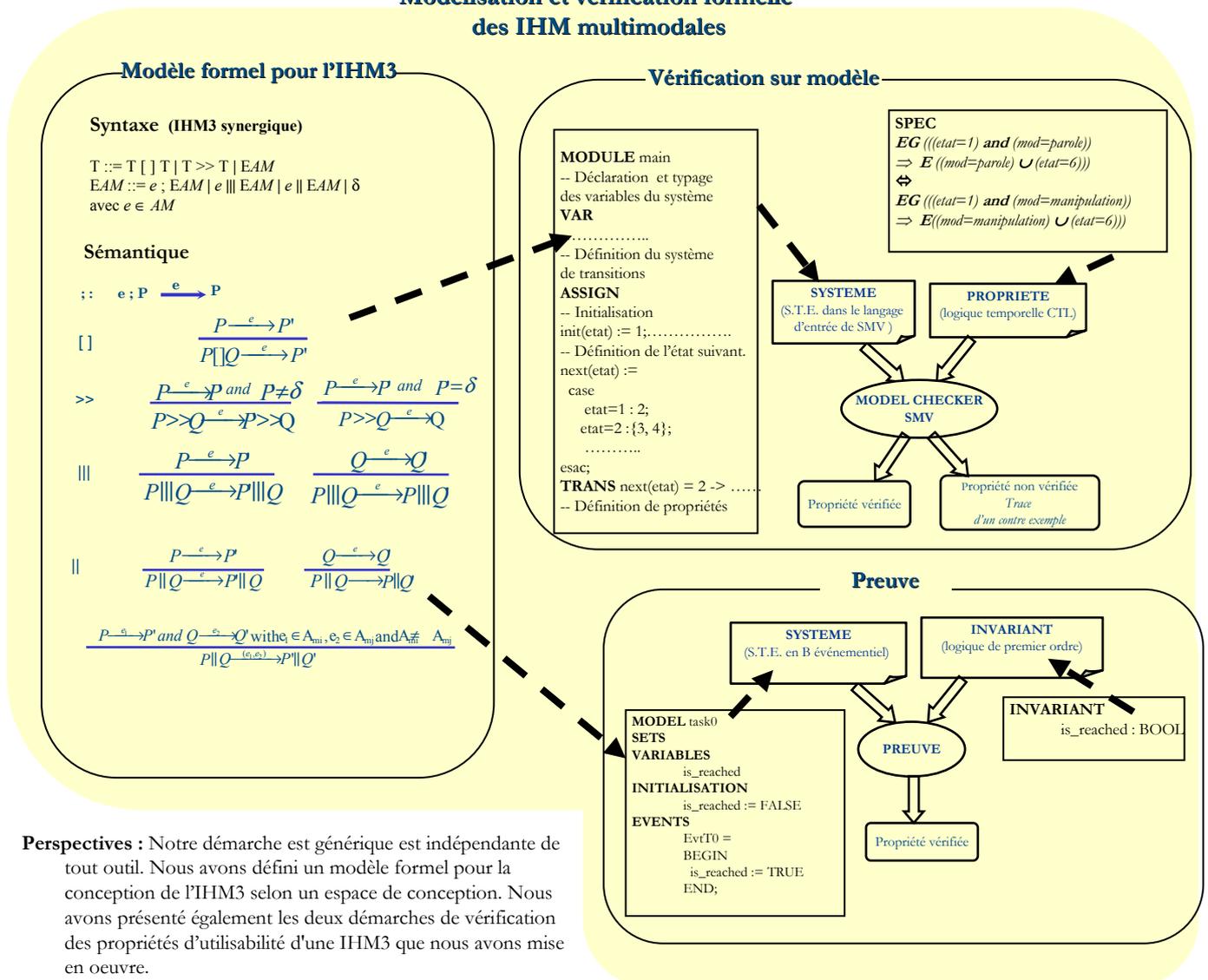
1. Bellik Yacine « Interfaces multimodales : concepts, modèles et architectures » thèse de doctorat d'université 1995.
2. Bier, E., Stone, M. Pier, K. Buxton, W. & DeRose, T. "Toolglass and Magic Lenses: the See-Through Interface", Proceedings of ACM SIGGRAPH'93, ACM Press, pp.73-80, 1993.
3. Clarke, E.M. & Emerson, E. A. "Synthesis of synchronization skeletons for branching time temporal logic". In logics of programs : workshop, volume 131, York-town Heights, New York, 1981. Springer-Verlag.
4. Coutaz, J., Nigay, L., Salber, D., Blandford, A., May, J. and Young, R.M. « Four easy pieces for assessing the usability of multimodal interaction: the CARE properties ». In K. Nordby, P.H. Helmersen, D.J. Gilmore and S.A. Arnesen (eds) *Human Computer Interaction: Interact '95*. Chapman and Hall: London. pp. 115-120, 1995.
5. Z.Manna & A.Pnueli. « *the temporal logic of reactive and concurrency systems : specification* ». Springer-verlag, 1992.
6. McMillan, K. L., "The SMV system", Technical report, Carnegie Mellon University, 1992.
7. L. Nigay & J. Coutaz. « A Generic Platform for Addressing the Multimodal Challenge », Proceedings of ACM CHI 1995, pp. 98-105.
8. L. Nigay, J. Coutaz, « *Espaces conceptuels pour l'interaction multimédia et multimodale* », TSI, spéciale Multimédia et collectif, AFCET & HERMES Publ, Vol 15(9), 1996, p. 1195-1225
9. J. Queille & J. Sifakis. « *Specification and verification of concurrent systems in Caesar* ». In 5th international symp. On programming, Lecture Notes in Computer Science, pages 337-351. Springer Verlag, 1981.
10. D. Plotkin. "A structural approach to operational semantics". Rapport de recherche DAIMI FN-19 Université d'Arhus Computer Science departement Arhus Danemark 1981.
11. N. Kamel. « utilisation de SMV pour la vérification de propriétés d'IHM multimodales ». 16eme conférence francophone sur l'Interaction Homme Machine. Namur 2004, pp 219-222. ACM Press.

Modélisation et vérification formelle des IHM multimodales

Objectif : les techniques formelles sont basées sur des modèles mathématiques qui permettent la description des systèmes d'une manière non ambiguës. Elles permettent également la spécification des propriétés, leur vérification et la validation à des étapes très avancées dans le processus du développement. Ceci évitera les testes effectués en fin du cycle du développement qui provoquent en général un coût élevé de maintenance. L'utilisation des méthodes formelles pour la modélisation des IHM multimodale (IHM3) et la vérification de ses propriétés constitue l'objectif de nos travaux.



Modélisation et vérification formelle des IHM multimodales



Perspectives : Notre démarche est générique est indépendante de tout outil. Nous avons défini un modèle formel pour la conception de l'IHM3 selon un espace de conception. Nous avons présenté également les deux démarches de vérification des propriétés d'utilisabilité d'une IHM3 que nous avons mise en oeuvre.

- Les travaux en cours concernent la modélisation formelle des propriétés CARE ainsi qu'à l'utilisation de l'approche descendante avec un développement par décomposition en utilisant le raffinement et la technique de preuve pour la vérification des propriétés en utilisant l'atelier B et le B événementiel.
- Les travaux futurs s'intéressent à la proposition d'une approche globale combinant les deux techniques de conception et de vérification afin d'exploiter les avantages de chacune dans un même processus de développement.



Nadjet KAMEL

kamel@ensma.fr

LISI / ENSMA

Téléport 2 - 1 avenue Clément Ader

BP 40109

86961 Futuroscope Chasseneuil cedex

La conception rationalisée pour les systèmes interactifs

Xavier Lacaze

LIHS-IRIT

Université Paul Sabatier, 118, route de Narbonne, 31062 Toulouse Cedex 4

lacaze@irit.fr

RESUME

Cet article présente le design rationale (DR) : ses principes, ses buts. Après avoir mis en évidence les manques des approches de DR actuelle, nous proposons une notation étendue de QOC. Notre approche étend la notation de façon à ce que les spécificités liées à la conception de systèmes interactifs soient prises en considération. Enfin pour réduire le coût de saisie du design rationale, nous proposons un outil supportant la notation décrite.

MOTS CLES : QOC, conception rationnelle, outil.

ABSTRACT

This paper presents design rationale: its principles, its goals. We first highlight what is lacking in DR. To solve these problems, we propose an extended notation. These extensions are based on the QOC notation. The extended notation is more dedicated to the design of interactive systems. Finally, to reduce the cost production of DR we present a tool supporting the notation described.

KEYWORDS: QOC, design rationale, tool support.

INTRODUCTION

Tout au long du cycle de vie de la conception d'un système interactif des choix, et des justifications sur ces choix, sont pris par les concepteurs graphistes, le client, les développeurs, etc. Toutes ces informations ne sont ni capitalisées, ni réutilisées pour la conception d'autres systèmes interactifs. Le design rationale apporte, en partie, une solution à ce problème en permettant de tracer les choix de conception. Cependant le DR n'est que peu usité pour les raisons suivantes.

Rationaliser la conception rencontre les mêmes difficultés quant à l'acceptation par les concepteurs/développeurs de systèmes informatique que la conception centrée utilisateur. Peu de personnes la pratiquent car le ratio coût/bénéfice est difficile à percevoir. Pourtant, la conception rationalisée permet d'explorer de façon systématique les options de conception tout au long du processus de conception, de justifier les choix de conception et de garder la trace des décisions prises. Ce genre d'approche permet d'obtenir des conceptions plus rationnelles c'est à dire moins fondées sur la confiance et la croyance dans les capacités du concepteur. Une des façons classique de justifier la mise en œuvre d'une telle conception consiste à mettre l'accent sur la qualité accrue des systèmes ainsi réalisés. Toutefois, il est extrêmement difficile de prouver ce genre de résultats et encore plus

difficile de convaincre une personne sceptique. De la même manière que la conception centrée utilisateur prouve sa rentabilité dans la phase d'utilisation des systèmes, la conception rationalisée ne peut prouver sa rentabilité que sur plusieurs projets. En effet, un des intérêts majeurs de la rationalisation de la conception est de pouvoir réutiliser à la fois les options étudiées (acceptées ou rejetées) dans les projets précédents ainsi que les justifications qui ont été mises en avant pour choisir telle ou telle option. Cette rentabilité sur le long terme est difficile à justifier et est sans doute une des raisons de la mise en œuvre très rare de conception rationalisée en entreprise et ce, même dans le cadre de projets touchant à des systèmes critiques. Une des raisons de la mise en œuvre très occasionnelle de la conception rationalisée réside sans doute aussi dans le manque d'outils aidant à sa mise en place qui est par nature une activité très lourde à la fois pour le recueil et l'enregistrement des informations mais aussi pour leur exploitation (rechercher des informations est les extraire pour ensuite les réutiliser).

Notre approche possède un but triple : (1) réduire les coûts dus à la phase de construction du DR en proposant un environnement d'édition, (2) augmenter les bénéfices en spécialisant le DR pour la conception des systèmes interactifs, (3) proposer de raccrocher des théories, des approches spécifiques à l'IHM au DR. Dans cet article nous décrierons le DR, ses objectifs et ses apports. Nous présenterons, ensuite, les extensions que nous proposons ainsi que l'outil associé.

DESIGN RATIONALE

Nous prenons comme définition celle proposée par [12] : *"Le DR est une explication de comment et pourquoi un artefact, ou des portions de ce dernier, a été conçu comme il est. Le DR est une description du raisonnement justifiant les résultats de la conception – comment les structures accomplissent leurs fonctions, comment des structures particulières sont choisis au détriment d'autres alternatives, quel comportement est attendu et sous quelles conditions. Pour faire court, le DR explicite les pourquoi d'un processus de conception en décrivant ce qu'est l'artefact, ce qu'il est supposé faire, et comment il a été conçu. Le DR représente les connaissances de la conception."*

Objectifs

Tout d'abord, avant de parler d'utilité, il faut toujours penser que le DR n'est utilisé que si la balance

coût/bénéfice est équilibrée. Cela signifie que l'investissement mis en œuvre pour la construction du DR doit être relativement peu coûteux par rapport aux bénéfices engendrés. La première tâche à réaliser est de capturer le DR. Un enregistrement vidéo d'une réunion peut être une solution économique, mais donne des résultats peu ou pas exploitables. L'outil gIBIS [7] et [9], basé sur la notation IBIS (*Issued-Based Information System*) [14], a été proposée dans l'optique premier de capturer les informations nécessaires au moindre coût, tout en permettant de retrouver les informations rapidement. IBIS connaît des problèmes de passage à l'échelle, ceci est dû à la non hiérarchisation des diagrammes. La notation PHI (*Procedural Hierarchy of Issues*) [18] a été proposée, en partie, pour combler cette non hiérarchisation, mais aussi pour rester au plus près de la discussion. PHI a fortement structuré les diagrammes, mais la notation ne permet pas d'extraire les informations importantes inhérentes de la discussion.

Le DR force les concepteurs à clarifier leurs décisions et à les argumenter suffisamment. Ceci a un impact direct sur la qualité future de l'artefact final [21]. La notation DRL (*Decision Representation Language*) [16] est la notation de DR qui recherche, avant tout, à capturer le processus qui a motivé le ou les concepteurs à choisir une alternative parmi d'autres. DRL saisit plus d'information que nécessaire pour le DR ceci rend les diagrammes rapidement illisibles. Des inconsistances et/ou des incohérences peuvent être détectées plus tôt dans la phase de conception, cet aspect est plus ou moins spécifique suivant la notation utilisée et à l'environnement associé. La notation formelle DQN (*Design Question Notation*), basée sur QOC (*Questions, Options, Criteria*) [17], proposée par Bramwell [3], [4] permet effectivement de détecter les incompatibilités très tôt dans le processus de conception. Enfin le DR agit comme une mémoire partagée du projet pour les différentes parties prenantes du projet. Ce dernier point est important pour les projets importants en personnes et/ou en temps.

Apports

Tout d'abord, parlons de la maintenance, en effet d'après [8], lors de la conception d'un système, 70% des coûts utilisés dans le développement du système le sont uniquement pour la phase de maintenance. Et plus de la moitié des efforts investis dans la phase de maintenance sont passés à comprendre le système. Le DR aide justement à la compréhension du système ainsi qu'aux démarches intellectuelles associées à la mise en œuvre. La réutilisation des solutions, par blocs, est envisageable, afin de ne pas "réinventer la roue" à chaque fois, la notation QOC [17] a été pensée dans ce but [20]. Ensuite le DR, est utilisé comme une base commune de communication. La compréhension des notations est relativement triviale et ne nécessite pas ou peu de pré requis, ceci en fait une plateforme de communication commune pour les différents intervenants. Ceci est d'autant plus intéressant

quand on connaît la diversité des intervenants : clients, développeurs, ergonomes, concepteurs graphique, utilisateurs. Le DR permet au client de contrôler et de vérifier l'état d'avancement du projet. En effet, le client peut voir la tendance générale du projet, et vérifier si les concepteurs tiennent compte de ses exigences (i.e. les facteurs). Si le DR issue de la conception d'un système interactif est stocké numériquement, il est alors possible de réutiliser directement ce document comme une partie de l'aide utilisateur, ou comme un document accompagnant la spécification. Enfin, le DR ne prends pas de décision à la place des concepteurs, la décision finale leur appartient toujours.

NOTATION QOC ETENDUE

Dans ce paragraphe, nous allons d'abord présenter la notation QOC originelle décrite par MacLean, ensuite nous décrirons les extensions que nous proposons. L'objectif de ces extensions est de fournir une notation de DR spécifique à la conception des systèmes interactifs. Notre choix s'est porté sur la notation QOC pour les raisons suivantes. Une étude complète de son utilisabilité a été réalisée [22]. L'écriture et la lecture des diagrammes sont triviales. [2] montre que d'après des observations empiriques sur les activités de conception montre que ces dernières se décomposent naturellement en terme de questions, options et critères. QOC est la seule notation qui offre une notion de critère alors que les autres proposent des arguments. Ces critères permettent d'évaluer les options et de faciliter la réutilisation.

Notation QOC

QOC (*Questions, Options, Critères*), est une notation semi formelle proposée par MacLean [17]. Cette notation donne lieu à une représentation sous forme de diagramme, comme montré sur la *Figure 1*.

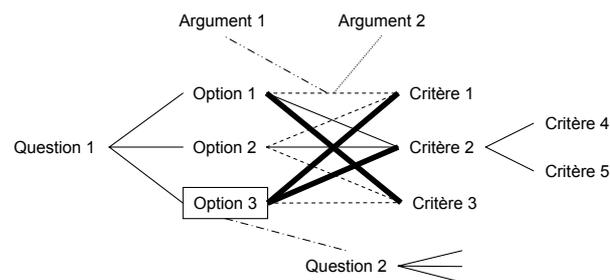


Figure 1 - Squelette de diagramme QOC.

Ce diagramme peut se décomposer en trois colonnes, une par élément de la notation (questions, options, critères), et des liens entre les éléments de ces colonnes. A chaque question on associe plusieurs options (i.e. choix de conception), et à ces différentes options on associe des critères qui favorisent (barre épaisse) ou défavorisent (barre en pointillé) les options. L'option retenue (i.e. l'option 3) est encadrée. Dans QOC, une option peut déboucher sur une nouvelle question (i.e. question 2), si bien qu'il apparaît des liens explicites entre différents diagrammes QOC. En outre des arguments peuvent être rajoutés pour supporter ou non l'évaluation portée par les

liens entre les options et les critères. Enfin les critères peuvent être définis en critères plus généraux.

Les défauts de la notation QOC ont été révélés par [22], une partie des extensions sont liées à ces remarques. L'autre partie des extensions intègre des théories issues du domaine de l'IHM. La première d'entre elles concerne l'insertion de modèles de tâches.

Modèles de Tâches

Cette extension proposée par Lacaze [15] vise à intégrer les modèles de tâches dans les diagrammes QOC. Cette extension est motivée par deux points : (1) les modèles de tâches influent fortement sur les choix de conceptions, ils doivent donc apparaître dans les diagrammes, et (2) les modèles de tâche sont des informations déjà existantes donc leur coût de production est nul. L'extension est destinée spécifiquement à la conception de la partie dialogue d'un système interactif (si l'on se réfère au modèle de Seeheim [11]). L'apport des modèles de tâche est double. Ils permettent (1) de relier rationnellement les options aux critères et d'éliminer les liens options/critères inutiles; (2) de pouvoir faire de l'évaluation rationnelle (par exemple de performance) d'un critère par rapport à une option. Les modèles de tâches et les scénarios ne sont que facultatifs, leurs insertions dépendent fortement du type de la question. Cette extension sera utilisée lors des problématiques liées au dialogue.

Facteurs

La notation de MacLean permet de raffiner des critères en critères plus génériques. Ces critères plus génériques n'apportent pas d'informations supplémentaires au diagramme. Nous proposons de nous rattacher à la proposition de [10] qui prévoit de remplacer ces critères génériques par des facteurs. Les critères et les facteurs sont redéfinis en adoptant les définitions de [19] : "Les facteurs de qualité représentent l'expression des exigences (point de vue externe, client). Les critères qualité sont les caractéristiques du produit (point de vue interne, technique). Les métriques permettent de mesurer un critère." Les facteurs permettent de faire apparaître clairement les besoins du client dans le processus de conception et dans la phase de choix. Les besoins du client ont un impact fort sur la prise de décision, c'est pourquoi ils doivent apparaître dans le diagramme. Néanmoins, les exigences du client n'ont pas toutes la même importance.

Pondérations

Nous proposons d'affecter des poids aux facteurs en fonctions de leur importance. Cinq valeurs sont disponibles allant de 1 (hautement prioritaire/obligatoire) à 5 (pas prioritaire/souhaitable). Les poids des facteurs se répercutent sur le poids des critères avec lesquels ils sont reliés. Les critères sont aussi pondérés de 1 à 5. Les poids associés peuvent évoluer au cours du temps (en fonction des changements de stratégie, délai de livraisons, etc.). Ces changements de poids ont un impact sur les options retenues. Les pondérations affinent les besoins, mais ne structure pas pour autant les diagrammes.

Seeheim

Le modèle de Seeheim [11] propose une architecture logicielle générique pour les systèmes interactifs (voir Figure 2). Bien que ce modèle ait été critiqué [23] à juste raison et ait évolué [1], ce modèle est générique et suffisamment précis pour décrire les différents ancres potentiels du DR sur l'architecture logicielle d'un système interactif.

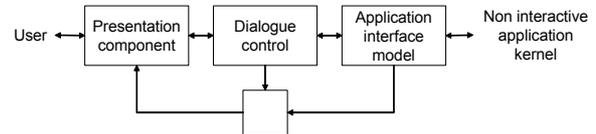


Figure 2 - Le modèle de Seeheim, [11].

Nous proposons, pour suivre le modèle de Seeheim, d'associer les questions à une des trois couches du modèle. Les problèmes liés à la présentation (respectivement au contrôleur de dialogue et noyau fonctionnel) seront ainsi regroupés. Ces regroupements permettent de structurer rationnellement les diagrammes QOC (lors de la conception) et de faciliter la relecture. Cette extension, ainsi que les précédentes sont supportées par l'outil.

DESCRIPTION DE L'OUTIL

La Figure 3 est une capture d'écran de l'outil développé.

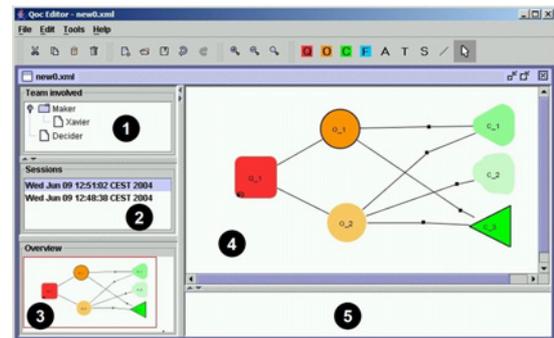


Figure 3 - Capture d'écran de l'éditeur QOC.

Les carrés représentent les questions, les ronds les options et les triangles les critères. Les critères ont une représentation associée à leur poids. La zone 1 affiche les personnes impliquées dans le projet. La zone 2 présente les différentes versions du diagramme, seule la dernière session est éditable. La zone 3 présente une mini vue du diagramme visualisé sur la zone 4. La zone 4 permet de visualiser et d'éditer les diagrammes. La zone 5 est destinée à la visualisation bifocale du diagramme [6]. Toutes ces zones sont interactives. L'outil est développé en Java. La manipulation des objets graphiques a été réalisée à l'aide de la librairie graphique JGraph [13].

Fonctionnalités

L'édition des diagrammes QOC, se fait par une interaction entièrement graphique. Les fonctionnalités classiques d'un éditeur sont présentes: couper, copier, coller, défaires, refaires. Ces fonctionnalités basiques permettent cependant de réutiliser facilement des portions de diagrammes d'un projet à l'autre. La fonction copier coller est toutefois spéciale car elle clone une entité. Par exemple, si le critère 1 est dupliqué, il apparaîtra deux fois sur

le diagramme, et les deux représentations partageront les mêmes informations (étiquette, poids, identificateur, fichiers liés). Le clonage des nœuds permet de disperser en plusieurs points du diagramme une même information. Ceci est surtout utilisé pour les critères et les facteurs, car un même critère (resp. facteur) apparaît généralement plusieurs fois dans le diagramme. Ainsi l'impact d'une modification est répercuté sur tout le diagramme. Il est alors possible de déceler d'éventuelles incohérences. L'outil permet de renseigner plus précisément un nœud, en y ajoutant des informations textuelles et en associant des fichiers (une vidéo, une image, une page web, etc.).

Visualisation

Le nombre d'information contenu dans un diagramme QOC est important. Ce problème nous a amené à proposer plusieurs solutions à l'utilisateur pour naviguer dans le diagramme, et le visualiser. L'utilisateur peut naviguer entre les différentes versions du diagramme. Ensuite, il peut naviguer à l'intérieur du diagramme, soit en utilisant la zone de travail (la zone 4), ou alors en utilisant la mini vue (la zone 3). Ces deux visualisations représentent l'information sous une même forme, mais à des niveaux de zoom différents. Enfin une visualisation bifocale [6] est proposée. La visualisation bifocale est une vue arborescente comportant deux focus : un focus contextuel et un focus détaillé. Cette visualisation nécessite une reconstruction du diagramme en arbre). Un filtre thématique (cf. modèle de Seeheim est aussi proposé, pour voir, par exemple, apparaître uniquement les questions traitant du noyau fonctionnel. Cela permet d'isoler les problèmes. Enfin, il est prévu de voir les impacts d'un facteur et/ou d'un critère sur le diagramme. L'outil ne prend jamais de décision à la place de l'utilisateur.

CONCLUSION ET PERSPECTIVES

Les travaux présentés vont dans le sens d'augmenter l'intérêt d'utiliser le DR. Pour augmenter les bénéfices, les extensions proposées permettent d'intégrer les modèles de tâches, ce classifier les questions. Les facteurs permettent d'intégrer clairement les exigences du client. Enfin l'outil doit réduire les coûts liés à la saisie du diagramme, mais aussi assister l'utilisateur dans la recherche d'information. Une étude de cas est actuellement en cours afin de valider ces points. L'étude de cas permettra de valider la notation étendue, en vérifiant si la notation permet aux concepteurs de saisir toutes les informations nécessaires à la conception d'un système interactif. Les perspectives concernent l'élaboration d'une bibliothèque de couples critères/facteurs pour chacun des niveaux de Seeheim. Pour la partie présentation nous nous baserons en partie sur les critères ergonomiques proposés par [5]. Le but de ces couples critères/facteurs est de faciliter la réutilisation.

BIBLIOGRAPHIE

1. Bass, L., Little, R., Pellegrino, R., Reed, S., Seacord, R., Sheppard, S., and Szejt, M. *The Arch Model: Seeheim Revisited*. User Interface Developers' Workshop. 1991.

2. Ball, L.J. and Ormerod, T.C. *Applying ethnography in the analysis and support of expertise in engineering design*. Design Studies, 21, pp. 403-421. 2000.
3. Bramwell, C. *Formal Development Methods for Interactive System: Combining Interactors and Design Rationale*. Ph.D. Thesis. University of York. 1995.
4. Bramwell, C., Fields, B., and Harrison, M. *Exploring Design Options Rationally*. DSV-IS'95: 2nd Eurographics Workshop on the Design Specification and Verification of Interactive Systems. Toulouse, June 1995.
5. Bastien, J. M. C., & Scapin, D. L. *Ergonomic criteria for the evaluation of human-computer interfaces* (Tech. Rep. 156). Rocquencourt, France: INRIA. 1993.
6. Cava, R.A., Luzzardi, P.R.G., Freitas, C.M.D.S., *The Bifocal Tree: a Technique for the Visualization of Hierarchical Information Structures*. In: IHC 2002 - 5th Workshop On Human Factors In Computer Systems, Forta. Fortaleza, 2002.
7. Conklin, J. and Begeman, M.L. *GIBIS: A Hypertext Tool for Exploratory Policy Discussion*. ACM Transactions on Office Information Systems, 6(4). pp. 303-31. 1988.
8. Conklin, J. *Design Rationale and Maintainability*. Proceedings 22nd Hawaii International Conference on System Science (IEEE Computer Society Press) pp. 533-539. 1989.
9. Conklin, J. and Burgess-Yakemovic, K.C. *A Process-Oriented Approach to Design Rationale*. In Moran, T.P., Carroll, J.M., eds *Design Rationale: Concepts, Techniques, and Use*. pp.393-428. Lawrence Erlbaum Associates. 1996.
10. Farenc, C. and Palanque, P. *Exploitation Des Notations De Design Rationale Pour Une Conception Justifiée Des Applications Interactives*. IHM'99. 11^{èmes} Journées Sur L'Ingénierie Des Interfaces Homme-Machine. 1999.
11. Green, M. *Report on Dialogue Specification Tools*. User Interface Management Systems. Springer-Verlag, pp. 9-20, 1985.
12. Gruber, T. R., and Russell, D. M. *Design Knowledge and Design Rationale: A Framework for Representation, Capture, and Use*. Technical Report, KSL 90-45. Knowledge Systems Laboratory, Stanford University. 1990.
13. JGraph est disponible à l'adresse suivante <http://www.jgraph.com>
14. Kunz, W. and Rittel, H. *Issues As Elements of Information Systems*. Berkeley : University of California. 1970.
15. Lacaze, X., Palanque, P. and Navarre, D. *Evaluation de Performance et Modèles de Tâches Comme Support à la Conception Rationnelle des Systèmes Interactifs*. 14^{ème} Conférence Franco-phoné sur L'Interaction Homme Machine, IHM 2002. pp.17-24. 2002.
16. Lee, J. *Extending the Potts and Bruns Model for Recording Design Rationale*. Proceedings of the 13th International Conference on Software Engineering. pp.114-125. 1991.
17. MacLean, A., Young, R.M., Bellotti, V.M.E. and Moran, T.P. *Questions, Options, and Criteria: Elements of Design Space Analysis*. In Moran, T.P., Carroll, J.M., eds *Design Rationale: Concepts, Techniques, and Use*. LEA. 1996.
18. McCall. *PHI: A Conceptual Foundation for Design Hypermedia*. Design Studies, 12. pp. 30-41. 1991.
19. McCall, J., Richards, P. and Walters, G. *Factors in Software Quality*. Rome Air Development Center (RADC), RADC-TR-77-369, Vol III, November 1977.
20. McKerlie, D., MacLean, A. *Reasoning with Design Rationale: Practical Experience with Design Space Analysis*. Design Studies (15). pp. 214-226. 1994.
21. Newman S.E. and Marshall, C.C. *Pushing Toulmin Too Far: Learning From an Argument Representation Scheme*. Xerox PARC Technical Report No. SSL-92-45. 1991.
22. Shum, S.J. *Cognitive analysis of design rationale representation*. PhD Thesis. University of York, UK, 1991.
23. Ten Hagen, P.J.W. *Critique of the Seeheim Model*. User Interface Management and Design. Springer-Verlag, pp. 3-6. 1990.

CONCEPTION RATIONALISÉE POUR LES SYSTÈMES INTERACTIFS

Xavier LACAZE



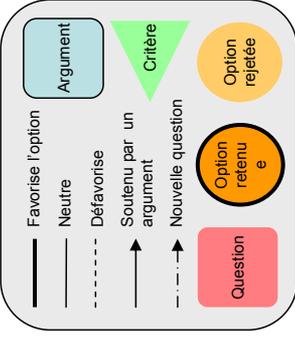
LIHS – IRIT
 Université Paul Sabatier - 118, route de Narbonne, 31062 Toulouse Cedex 4
 lacaze@irit.fr

Définition : Design Rationale

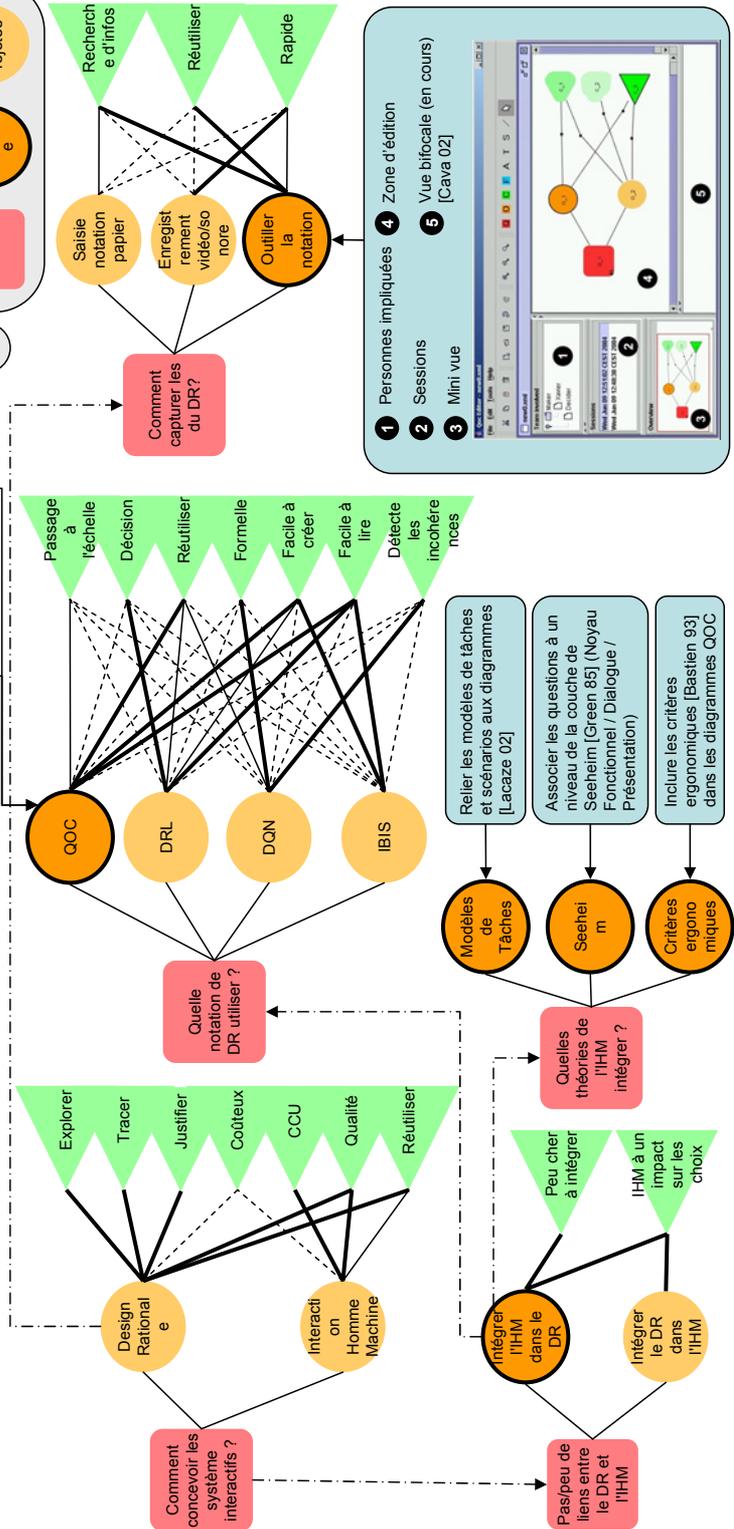
Le DR est une explication de comment et pourquoi un artefact, ou des portions de ce dernier, a été conçu comme il est. Le DR est une description du raisonnement, justifiant les résultats de la conception – comment les structures accompagnent leurs fonctions, comment des structures particulières sont choisies au détriment d'autres alternatives, quel comportement est attendu et sous quelles conditions. Pour faire court, le DR explicite les pourquoi d'un processus de conception en décrivant ce qu'est l'artefact, ce qu'il est supposé faire, et comment il a été conçu. Le DR représente les connaissances de la conception. (Gruher 1992)

Étude sur l'utilisabilité de la notation QOC [Shum 96]

Les activités de conception se décomposent naturellement en Q, O, C [Ball 00]



Légende



Apports de l'ergonomie à la conception de dispositifs techniques de lecture numérique

Angélica LEAL
lealangelica@yahoo.fr

Laboratoire PARAGRAPHE
Équipe Conception, Création, Compétences, Usages (C3U)
2, rue de la Liberté – 93526 – Saint-Denis
Directeur de thèse: Pierre RABARDEL – Co-direction: Viviane FOLCHER

RÉSUMÉ

Le travail de thèse en cours vise à contribuer directement dans le processus de conception des nouvelles technologies et des systèmes interactifs à travers l'analyse ergonomique des dispositifs techniques utilisés aujourd'hui pour la lecture numérique. Nous observerons que différentes populations sont concernées par cette activité et que pour enrichir les études d'*usability* des interfaces nous pouvons prendre en compte des concepts tel que le *design-for-all* et la plasticité des systèmes, notamment lors d'une analyse ergonomique. Nous présenterons ici les premiers éléments de la recherche en cours, avançant quelques hypothèses de travail et des perspectives développées à partir des résultats obtenus.

MOTS-CLÉS: Ergonomie cognitive, conception, *design-for-all*, plasticité de systèmes, *usability*.

ABSTRACT

It is the intent of this paper to describe the several phases developed in an exploratory study of ergonomics. This study aims at observing how users will interact with some devices and systems during the digital reading activity, providing information about the interactions between the users and the interfaces analyzed. In addition, the preliminary results obtained show that considering other concepts such as the *design-for-all* approach and the systems plasticity will increase the next investigations, guiding the considerations about the ergonomics contributions in a design process.

CATEGORIES AND SUBJECT DESCRIPTORS: H.5 [Information Interfaces and Presentation (e.g., HCI)]: User Interfaces.

GENERAL TERMS: Design, Human Factors.

KEYWORDS: Cognitive ergonomics, design, *design-for-all*, systems plasticity, *usability*.

INTRODUCTION

Comment l'ergonomie cognitive inscrite dans une approche systémique des théories de l'activité peut contribuer dans le processus de conception sera la question abordée dans ce travail de thèse. Il va à la rencontre d'une des préoccupations majeures de nos jours, la pluridisciplinarité et le rôle des disciplines scientifiques et techniques dans l'atteinte d'un but commun: la conception de produits et systèmes mieux adaptés à tout type d'utilisateur.

Ce travail de recherche s'insère dans deux problématiques distinctes, propres à chaque discipline concernée, mais pour autant partagées par différentes communautés:

- (1) l'ergonomie cognitive et les méthodes d'analyse de l'activité inscrites en milieu professionnel, dans la vie quotidienne ou dans la formation, qui permettront de rendre compte des nouveaux usages développés dans le temps, à partir des nouveaux artefacts issus des nouvelles technologies [06]. L'observation des usages et l'étude du développement d'instruments au service des activités humaines contribueront dans la production de recommandations pour nourrir le processus de conception et enrichir ce développement [12].
- (2) le design industriel et le concept de *design-for-all* ou *Universal Design*, qui vise à simplifier la vie de chacun en fabricant des produits et des systèmes de communication en prenant en compte la diversité des personnes, dans une perspective d'inclusion de toutes les caractéristiques et besoins de la population dans la construction d'un environnement plus confortable à l'usage [11]. Des réflexions sur la réelle possibilité d'universalisation des produits et les limites du concept ont conduit à son évolution vers le concept d'*usability* [04], avec des nouvelles réflexions sur l'accessibilité des produits et services, et plus actuellement, avec la croissance des outils informatiques, sur la plasticité des interfaces numériques [05]. Ces réflexions visent à contribuer, avec des éléments d'analyse, à la conception de produits et systèmes mieux adaptés aux personnes de tous âges, toutes tailles et toutes capacités.

L'idée centrale de cette étude est de porter des réflexions sur le rôle des sciences cognitives dans le processus de conception, permettant l'intégration des diverses méthodes utilisées dans les différents domaines, contribuant à l'interaction et au partage de connaissances et

d'informations dans les différentes étapes de la conception, en ayant pour but d'avoir des produits plus en harmonie avec le caractère développemental des utilisateurs. Nous considérerons que:

- (1) l'ergonomie vise à contribuer dans l'élaboration de produits qui favorisent (ou tout au moins n'empêchent pas) leur appropriation par les sujets, pour la constitution d'instruments qui puissent enrichir et collaborer au développement de nouveaux usages et des nouvelles activités [12].
- (2) le *design-for-all* réfléchit à une démarche complémentaire aux généralement envisagées par les concepteurs (i.e. la discrimination positive: concevoir un produit adapté à tel handicap, excluant tout autre type de handicap et limitant l'usage à un public ciblé et figé), puisque le *design-for-all* cherche davantage des solutions permettant la conception d'un seul et unique produit ou système adaptable et utilisable par toutes les personnes, malgré leurs différences physiques et motrices [11].

Ces considérations nous permettront d'étudier quelques-unes des méthodes d'aide à la conception existantes, dans la perspective de construction d'une grille comparative qui vise à montrer leurs points communs, les points de divergence et notamment comment l'une pourra compléter l'autre.

Le cadre théorique de cette thèse sera abordé à partir d'une étude de cas qui porte sur l'usage des dispositifs techniques dédiés à la lecture numérique sur écran par une population diversifiée, qui développe différentes pratiques de lecture selon le contexte d'usage et les buts à atteindre, ainsi que selon leurs besoins, connaissances et expériences personnels. Cette étude s'insère dans deux problématiques traitées aussi et typiquement par des travaux de recherche du domaine de l'informatique:

- (1) la plasticité des interfaces numériques et des systèmes interactifs, c'est-à-dire la capacité de s'adapter à tout type de dispositif technique (ordinateur de bureau, assistant personnel, téléphone portable et autres), d'environnement (à domicile, au travail, à l'école, dans un train) et d'utilisateur (débutants, expérimentés) [05].
- (2) l'accès à la technologie d'intelligence ambiante, c'est-à-dire avoir différents dispositifs tels que les écrans d'ordinateur et de télévision, les téléphones portables, les interfaces vocales et autres permettant d'interagir directement avec les objets de l'environnement tout en étant capables de communiquer entre eux, avec l'utilisateur et avec d'autres personnes, en particulier par des réseaux sans fil du type WiFi [10].

HISTORIQUE ET ÉTAT D'AVANCEMENT

Une première étape de ce doctorat a été faite dans le cadre d'un DEA d'ergonomie [08]. Une étude exploratoire a été faite au sein d'une bibliothèque municipale. Nous cherchions à observer l'activité de lecture développée à partir d'un dispositif technique issu des nouvelles technologies de lecture électronique sur écran, le livre

électronique ou *ebook*. La bibliothèque mettait en place un système de prêt qui permettait à tout abonné d'emprunter des *ebooks*, d'accéder à distance et de lire directement sur écran plus de 300 titres différents qui figuraient sur le fonds littéraire disponible. Notre objectif était d'observer cette activité de lecture de plus en plus nomade, pour pouvoir répondre à des questions concernant notamment le développement de nouvelles pratiques à partir de ces outils électroniques. Nous nous questionnions aussi sur la possibilité de remplacement des supports papier traditionnels par les supports électroniques et quelle serait la place occupée par ces dispositifs dans l'activité de lecture, dans la perspective de contribuer à la conception de ces outils et leur interface.

Des questionnaires répondus par un nombre important d'utilisateurs [03] nous ont permis de tracer les premiers éléments de réponse à quelques-unes de nos questions. Nous avons observé qu'à l'issue de cette expérimentation en bibliothèque, la population d'abonnés (qui était au départ constituée par des jeunes étudiants, des salariés et cadres d'âge moyen ainsi que par des personnes à la retraite) devenait de plus en plus restreinte et se constituait en grande partie par des personnes vieillissantes, qui présentaient à la fois un handicap moteur ou visuel. Cela pourrait s'expliquer par des raisons distinctes: (i) le contenu limité, composé en majorité par des ouvrages du domaine public, (ii) la facilité de transport de plusieurs ouvrages dans un seul support, (iii) l'autonomie de lecture qui procurent ces dispositifs, avec notamment le réglage de la taille des caractères et le retro-éclairage.

Les premiers résultats obtenus nous ont permis d'avoir un petit aperçu de la diversité de la population et des pratiques qui commençaient à se développer, mais nos questions concernant la place de l'*ebook* au sein de l'activité de lecture, son appropriation par les utilisateurs et un éventuel développement ou évolution de la propre activité n'avaient toujours pas de réponse. Nous n'avions pas non plus assez d'informations pour contribuer à un possible projet d'amélioration de l'interface du système, et ainsi élargir (ou du moins ne pas limiter) les contextes d'usage. Nous avons donc procédé à des entretiens individuels avec les utilisateurs pour recueillir des données sur leurs expériences et vécus avant et après l'utilisation des *ebooks*. Ces informations nous ont permis de mieux comprendre l'activité de lecture et à partir de leurs propos nous avons essayé de reconstruire le développement des pratiques dans le temps, identifiant des éléments du dispositif qui ont contribué (ou empêché) ce développement [07].

Des contraintes liées à la disponibilité des utilisateurs et du matériel ont difficilement la réalisation de tests-utilisateur, qui pourraient fournir des données plus fiables sur la performance des utilisateurs lors de l'usage des *ebooks*, telles que les erreurs commises, les retours en arrière, la vitesse de lecture et d'autres indicatifs observables durant la réalisation d'un scénario donné. Les données possibles

à recueillir étaient uniquement des verbalisations enregistrées durant les 5 entretiens réalisés, et pour les traiter nous avons choisi l'une des méthodes d'ergonomie cognitive qui privilégie le regard de l'utilisateur sur le dispositif technique, sans avoir à prendre en compte des données quantitatives de l'activité réalisée [13]. Ce choix et le nombre réduit des sujets interviewés limitent la généralisation des résultats obtenus. Cependant, le caractère plus illustratif que démonstratif de cette première étape de l'étude nous renseigne sur les nouvelles pratiques et nous donne des bases d'analyse pour les étapes suivantes.

Nous avons initialement procédé à un découpage de l'activité de lecture en une unité significative pour les sujets: les classes de situation. Identifiées à partir des récits des utilisateurs et à partir des situations singulières et diversifiées de leurs activités, nous avons constaté l'existence de différentes classes de situation qui ont été ensuite croisées à toute ressource nécessaire au déroulement de l'activité, à fin d'identifier quelles ressources disponibles dans le dispositif technique étaient mobilisées ou substituées par les sujets interviewés [07], [08]. La méthode utilisée, qui met en avant l'existence d'un système de ressources [13] construit par les propres utilisateurs pour pouvoir réaliser l'activité de lecture, nous a donné des résultats sur deux plans spécifiques: sur le plan synchronique de l'activité (i) nous observons que les ressources disponibles dans le dispositif technique sont à moitié mobilisées, que les substitutions dues à l'absence ou aux défaillances des ressources sont importantes et que différentes ressources sont construites par les sujets à partir de leurs expériences et connaissances préalables. Sur le plan diachronique de l'activité (ii) cette méthode nous donne des pistes de compréhension de l'appropriation des dispositifs techniques par les sujets pour la constitution d'instruments [12], car nous avons avéré des traces de transformations de leur activité à partir de l'analyse des substitutions effectuées par les sujets.

Avec ces analyses nous avons reconstruit d'une manière historique l'activité de lecture développée sur l'*ebook* et nous avons observé un développement différencié selon les sujets. Ce développement est dans certains cas "constructif", c'est-à-dire enrichissant pour les utilisateurs et leurs pratiques de lecture, mais parfois il est "destructif", dans le sens où une rupture ou un empêchement des pratiques se produit, causé notamment par les défaillances du dispositif technique. Ceci peut nous renseigner sur la logique et l'évolution de l'organisation de l'activité selon les différents objectifs définis par chaque individu, notamment dans le cadre des personnes âgées ou handicapées visuelles qui développent une activité outillée par un support d'aide à la lecture. On observe que ces dispositifs peuvent changer le rapport des utilisateurs avec eux-mêmes, avec les autres et aussi avec l'environnement.

La procédure de recueil des verbalisations des sujets et de leur codage nous a donné des pistes pour la conception

d'une nouvelle interface d'*ebook*. Pendant les entretiens les utilisateurs exprimaient, par l'intermédiaire de leurs pratiques et usages, les difficultés rencontrées, les échecs, les éventuels doutes et ambiguïtés ressentis durant l'interaction avec les machines, causés notamment par l'interface proposée par les concepteurs. Des éléments d'amélioration relatifs aux icônes, barres de défilement/repérage ou propres au système d'affichage des informations peuvent être identifiés à partir de leurs expériences et pourront faire l'objet de recommandations à intégrer lors de la conception d'un nouveau modèle d'*ebook*. L'un des points à mettre en relief dans cette démarche est la similarité avec des méthodes utilisées par les professionnels du design pour identifier les points faibles et les atouts des prototypes lors de la conception d'un produit/système (comme des questionnaires du type *user needs*), mais avec l'importante différence de pouvoir intégrer une telle démarche en amont dans le processus de conception (en phase de prototypage), au lieu de faire une analyse du produit final [02].

PERSPECTIVES

Dans la perspective d'enrichir l'étude en cours, nous envisageons de poursuivre les recherches futures par des chemins divers. Les perspectives de cette étude en cours sont diversifiées et peuvent diriger le travail vers différents résultats et conclusions. Nous identifions deux pistes distinctes à suivre:

- (1) contribuer à définir des éléments de comparaison entre différentes méthodes d'aide à la conception, visant à enrichir ce processus.
- (2) contribuer à la conception de dispositifs techniques permettant la lecture sur écran, et qui puissent répondre aux préoccupations évoquées avec les concepts de *design-for-all* et plasticité des systèmes.

Les conclusions initiales permettent de réfléchir au rôle de l'ergonomie dans la conception des produits/systèmes issus des nouvelles technologies. Un partenariat entre designers, informaticiens et ergonomes serait envisagé pour améliorer ces dispositifs, puisque l'analyse ergonomique initiée donne des pistes sur la conception d'un nouvel outil de lecture, avec des changements aussi bien pour l'interface que pour les fonctions agrégées. Cependant, nous n'avons pas assez d'éléments qui renseignent sur les limites de la démarche choisie, ainsi que sur les possibilités de généralisation des résultats obtenus en ce qui concernent d'autres dispositifs techniques. Et c'est précisément sur ce point qu'un partenariat avec des concepteurs permettrait une étude comparative entre les différentes méthodes adoptées selon le domaine d'application et qui visent la conception, pour obtenir des informations favorisant la construction de la grille comparative qui montrerait les points communs, les points de divergence et les points de complémentarité des méthodes étudiées. Une telle initiative pourrait contribuer directement dans le processus de conception des produits inscrits dans les problématiques du *design-for-all* et de la plasticité des systèmes interactifs.

Les premiers résultats obtenus indiquent que les utilisateurs qui présentent un handicap visuel constituent une partie importante de la population susceptible d'utiliser ces dispositifs techniques et nous nous demandons si leur rôle est autre que celui d'aide à la lecture. Nous observons que l'ergonomie et d'autres disciplines comme le design et l'informatique ont pu contribuer à l'amélioration des dispositifs destinés aux malvoyants, notamment en ce qui concernent les outils d'amélioration de la visibilité et lisibilité des textes, comme les agrandisseurs d'écran [14]. Cependant, une parcelle importante des analyses effectuées sur ces outils se limite à exposer des résultats sur la performance, la vitesse de lecture, la direction du regard, la fatigue visuelle, le temps de concentration et autres [01], sans pour autant fournir des éléments concluants sur l'activité en soi et le développement des pratiques de lecture dans le temps. L'une des questions à poser est si n'y a-t-il pas d'autres éléments à considérer, d'autres concepts à mettre en relief comme le repérage dans le temps et dans l'espace lors de la lecture, la possibilité de prise de notes, la conservation des traces de l'activité réalisée, les possibles interactions avec le contenu numérique, voire même la dimension collective de la lecture et le contexte auquel cette lecture est-elle inscrite? L'ergonomie peut apporter des éléments de compréhension de l'activité de lecture, dans des termes beaucoup plus profonds et complexes que simplement la 'tâche de lire'. Ceci inclut la compréhension du texte numérique et l'intégration de cette activité dans la vie au travail, dans la formation et dans les autres activités quotidiennes [08].

Pour contribuer à la conception des dispositifs techniques permettant la lecture sur écran, nous comptons élargir les investigations en intégrant d'autres outils tels que d'autres modèles plus récents d'*ebooks*, mais également des assistants personnels digitaux (PDA), des smartphones et des ordinateurs portables, c'est-à-dire des supports qui utilisent la lecture numérique comme base d'intégration de différentes activités nomades. Notre objectif est d'observer la nature et l'orientation des pratiques de lecture développées par la diversité de la population, pour identifier des éléments qui s'inscrivent aux problématiques du *design-for-all* et de la plasticité des systèmes, et qui contribueront à l'amélioration des dispositifs techniques en terme d'accessibilité. Par ailleurs, nous comptons analyser plus finement l'activité de lecture effectuée à partir des différents dispositifs techniques proposés, à travers des scénarii enregistrés et des carnets de bord détaillés, contenant des informations sur les ressources mobilisées ainsi que sur les difficultés et problèmes rencontrés lors des interactions, et éventuellement les solutions envisagées par les sujets. L'analyse de ces données nous permettra dans un premier temps de faire une comparaison entre les différents dispositifs pour les classer en terme d'*usability* et d'adéquation avec les besoins spécifiques des utilisateurs [02], et dans un deuxième temps d'évaluer les ressources

nécessaires au déroulement de l'activité de lecture instrumentée, ainsi que l'influence, voire l'impact et les ruptures causés par ces ressources dans les nouvelles pratiques identifiées.

BIBLIOGRAPHIE

01. Baccino T., Colombi T., (2001). L'analyse des mouvements des yeux sur le Web. In *Les interactions homme-système: perspectives et recherches psychologiques*. A. VomHofe (Éd.), Paris: Hermès.
02. Bastien C., Scapin D., (2001). Évaluation des systèmes d'information et critères ergonomiques. In *Systèmes d'information et interaction homme-machine*. C. Colski (Éd.), Paris: Hermès.
03. Bélisle C. et coll., (2002). Contrats de lecture: rapport sur une expérimentation de prêt de livres électroniques en bibliothèque, dimensions technico-économiques et socio-cognitives. En ligne sur http://www.isdn.enssib.fr/otr_pg/archiv.htm#etudes.
04. Bevan N., Kirakowski J., Maissel J., (1991). What is usability? Proceeding of the 4th International Conference on HCI. September 1991, Stuttgart.
05. Calvary G., Coutaz J., (2002). Plasticité des interfaces: une nécessité. Actes des 2^e assises nationales du GdR I3 (pp. 247-261).
06. Kuutti K., (1996). Activity Theory as a Potential Framework for Human-Computer Interaction Research. In *Context and Consciousness: Activity Theory and Human-Computer Interaction* (pp. 17-44). Cambridge, Massachusetts: The MIT Press.
07. Folcher V., Leal A., (2004). *Mobilisation et construction de ressources dans l'utilisation d'artefacts électroniques de lecture numérique*. Communication acceptée au Colloque Alternatives en Sciences Cognitives (ARCo'04). Décembre 2004, Compiègne.
08. Leal A., (2003). *Les supports de lecture électronique et la lecture numérique: des artefacts pour une nouvelle activité?* Mémoire de DEA Ergonomie. CNAM, Paris.
09. Norman D. A., (1988). *The Design of Everyday Things*. London: The MIT Press.
10. OZONE Consortium, (2003). En ligne sur <http://www.extra.research.philips.com/euprojects/ozone>.
11. Pasquier T. L., (2004). *Le rôle des sciences cognitives dans le design for all*. Actes du 12^{ème} Séminaire CONFERE. Juillet 2004, Nantes.
12. Rabardel P., (1995). *Les hommes et les technologies: Approche cognitive des instruments contemporains*. Paris: Armand Colin.
13. Rabardel P., Bourmaud G., (2003). From computer to instruments systems: a developmental perspective. In *Interacting with Computers: the Interdisciplinary Journal of Human-Computer Interaction*. Vol. 15, 665-691.
14. Spérandio J. C., Uzan G., (2002). Ergonomie des aides techniques informatiques pour personnes handicapées. In *Handicap*, revue de Sciences Humaines et Sociales, n° 93.

Apports de l'ergonomie à la conception de dispositifs techniques de lecture numérique

Les questionnements de départ

- La place des dispositifs techniques dans la lecture électronique
- Remplacement du livre papier ou complémentarité
- Différents usages selon la diversité de la population
- Impact sur l'ancienne activité de lecture et sur les sujets
 - les activités évoluent dans les temps avec l'introduction de nouveaux dispositifs techniques
 - les nouveaux dispositifs techniques font l'objet d'une appropriation par les sujets, avec le temps
 - les sujets utilisent les ressources disponibles des dispositifs techniques pour réaliser l'activité de lecture

Questionnaires

- Diversité de la population
- Construction d'une "cartographie" des usages

MÉTHODE D'ANALYSE

Entretiens

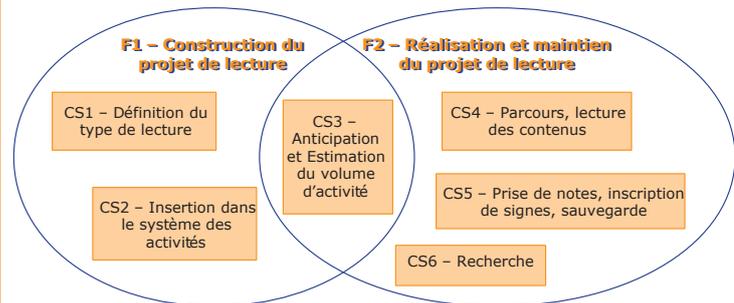
- Reconstruction de l'activité de lecture et de son organisation
- Articulation de l'activité de lecture avec d'autres activités
- Compréhension du processus d'appropriation des artefacts

MDSR

- Découpage en domaines, familles d'activité et classes de situation
- Nature des ressources observées (artefact, sujet et situation)
- Analyse des ressources disponibles (usage, non-usage, substitution)

MÉTHODE DES DÉFAILLANCES ET SUBSTITUTION DES

Domaine d'activité de la LECTURE



DÉCOUPAGE DE L'ACTIVITÉ ANALYSÉE

Barre 01

Navigation et repérage dans l'artefact

- TM Table de matières
- BR Barre de repérage
- OU Ouvrir ouvrages
- IN Information ouvrages
- AC Retour à l'accueil
- MO Monter
- DE Descendre

Barre 02

Navigation dans les ouvrages

- BD Barre de défilement
- NP Nombre de pages
- AP Avancer pages
- RP Reculer pages
- AF Avancer à la fin
- RD Retourner au départ
- 34 Afficher barres 3/4

Barre 03

Configuration de l'artefact et de l'affichage

- TC Taille des caractères
- IN Information ouvrages
- AN^p Afficher nombre de pages
- A4 Afficher barre 4
- AB^o Afficher barre de défilement
- AC Retour à l'accueil
- M3 Masquer barre 3

Barre 04

Manipulation du contenu numérisé

- CF^c Couleur fond et caractère
- ST Souligner texte
- PN^c Prise de notes clavier
- PN^e Prise de notes écriture
- MP Marque pages
- AP Afficher portrait
- AP^p Afficher paysage
- RM Recherche mots*

Dictionnaire
Mot-clef

SCHÉMATISATION DES RESSOURCES ANALYSÉES

TABLEAU D'ANALYSE DES RESSOURCES (EXTRAIT)

Usage (●), non-usage (○) et substitution (★) de ressources dans la Réalisation du Projet de Lecture					
Classes de situations	C3. Estimation volume de l'activité	C4. Parcours, lecture des contenus	C5. Inscription de signes, sauvegarde	C6. Recherche	
Ressources disponibles					
02. Navigation dans les ouvrages					
Barre de défilement des pages	● ● ● ● ●				
Nombre de pages	● ● ● ● ●				
03. Configuration de l'artefact et de l'affichage					
Taille des caractères		● ● ● ● ● ● ●			
04. Manipulation des contenus					
Couleur (fond et caractères)			● ●		
Soulignement			○ ○		
Prise de notes clavier virtuel			○ ● ○ ○		
Prise de notes reconnaissance écriture			○ ★ ○ ★		
Marque-pages	● ●		●		
Affichage paysage / portrait		○			
Recherche de mots – dictionnaire					○ ● ●
Recherche de mots – mot-clef					○ ● ● ○ ★

Analyse préliminaire de l'usage des ressources

- Les ressources disponibles sont utilisées à l'ordre de 40% et on compte 30% de non-usage. On distingue 40% de ressources substituées.
- Les ressources plus utilisées sont celles concernant la circulation dans le texte (icône et bouton pour avancer ou reculer les pages et marque-pages, ainsi que sommaire et table de matières) et la configuration de l'affichage (agrandissement des caractères et rétro-éclairage).
- Les ressources concernant la manipulation et l'interaction avec le texte (prise de notes, recherche de mots et enregistrement des traces) ne sont quasiment pas utilisées par les sujets et font en grande partie l'objet d'une substitution.
- La prise de notes et la sauvegarde semblent faire l'objet d'une discordance entre l'unité de l'activité et celle de l'artefact, ce qui peut expliquer le non-usage.
- L'estimation du volume de l'activité mobilise la ressource d'affichage du nombre de pages et fait l'objet d'une mobilisation différenciée selon le sujet.

Angélica LEAL • lealangelica@yahoo.fr

Directeur de thèse: Pierre RABARDEL
Co-direction: Viviane FOLCHER

Laboratoire PARAGRAPHE
Équipe Conception, Création, Compétences, Usages (C3U)
2, rue de la Liberté • 93526 • Saint-Denis

Méthodes de conception pour les systèmes de Réalité Augmentée

Benoît Mansoux

CLIPS-IMAG équipe IIHM & TIMC-IMAG équipe GMCAO
BP53
38041, Grenoble cedex 9, France
benoit.mansoux@imag.fr

RESUME

Mes travaux de recherche doctorale ont trait à la conception et à la réalisation de systèmes interactifs dans le cadre d'applications de chirurgie assistée par ordinateur. Dans ce contexte pluridisciplinaire (Interaction Homme-Machine et Gestes Médico-Chirurgicaux Assistés par Ordinateur), l'originalité de nos travaux réside dans l'étude de l'interaction entre le chirurgien et le système informatique. En effet aujourd'hui encore la conception de systèmes de chirurgie assistée par ordinateur est principalement guidée par les technologies disponibles et non par les besoins des utilisateurs. La conception reste ad-hoc. Afin de capitaliser les expériences de conception, l'objectif de nos travaux est donc la mise au point d'éléments et méthode de conception généraux, indépendants d'une spécialité clinique, pour des systèmes de chirurgie assistée par ordinateur.

MOTS CLES : Chirurgie Augmentée, Réalité Augmentée, Conception, Techniques d'Interaction, Mini-écran.

ABSTRACT

My doctoral research focuses on the design and development of Computer Assisted Surgery (CAS) system. The aim of our work is twofold. Firstly, we want to reinforce the consideration given to the surgeon and his/her interaction with a CAS system when designing such systems. This is crucial to facilitate the design of more usable CAS systems and to increase their ability to be smoothly integrated in a medical / surgical environment. This first goal relies on tools and mechanisms that will help the designer to choose well-suited devices and the best ways of merging digital and physical medical worlds. Secondly, we want to adopt a systematic design approach of CAS systems, in order to capitalise design experiences and so to increase the software and ergonomic quality of the developed CAS systems.

CATEGORIES AND SUBJECT DESCRIPTORS: H.5.2 [Information Interfaces And Presentation]: *User-Centered Design, Theory and Methods.*

GENERAL TERMS: Design, Human Factors.

KEYWORDS : Computer Assisted Surgery, Augmented Reality, Design Method, Interaction Techniques, Mini-screen.

MOTIVATION ET POSITIONNEMENT DES TRAVAUX

La Réalité Augmentée (RA) vise à enrichir la réalisation d'une tâche qui se situe dans le monde réel au moyen d'éléments issus du monde numérique. Les domaines d'application des systèmes de RA se diversifient (maintenance, transport, culture, jeux, etc.). Dans le domaine médical, l'évolution des techniques de chirurgie assistée par ordinateur et d'imagerie a permis la diversification des applications et par conséquent la diversification des techniques d'interaction utilisant la RA. Cependant, la majorité des systèmes, bien qu'innovante du point de vue clinique, est conçue de manière ad-hoc : nous constatons très peu de capitalisation et de réutilisation du savoir-faire. De plus, les méthodes de conception traditionnelles ne sont pas adaptées aux systèmes de RA qui nécessitent notamment la prise en compte du monde physique autour de l'utilisateur du système. Actuellement, de nombreux travaux de recherche portent sur la conception de systèmes de RA (ou de systèmes mixtes en général). Nous classons ces travaux en quatre catégories en fonction de leur apport au long du processus de développement. Ces catégories sont : Notations de conception, Approches par modèles, Approches par composants, Patrons de conception.

Notations de conception

Nous retenons deux notations ASUR [2] et IRVO [2]. Ces notations s'intéressent principalement à la conception de l'interaction en soulignant notamment les différentes caractéristiques des entités prenant part à l'interaction. Ces notations mettent aussi l'accent sur les relations entre les entités du monde physique qui participent à l'interaction. Bien qu'utilisant un vocabulaire différent, les mêmes concepts se retrouvent dans ces deux notations, même si IRVO est a priori adaptée pour la représentation de tous les systèmes mixtes alors qu'ASUR

se limite pour l'instant aux systèmes de RA. Ces notations constituent de bons outils pour décrire, comparer et donc classer les systèmes existants. Leurs apports lors de la conception de nouveaux systèmes semblent difficiles à évaluer. Des méthodes d'utilisation de ces notations ainsi que des patrons de conception pourraient favoriser l'assimilation des concepts manipulés et être très utiles en phase de conception.

Il convient également de noter dans cette catégorie la notation définie dans [10], utile dans le cadre des systèmes mixtes collaboratifs et mobiles.

Approches par modèles

Les travaux présentés dans [11] s'appuient sur le concept d'espace d'interaction mixte. Un espace d'interaction est modélisé par quatre aspects concernant les entités prenant part à l'interaction. Ces aspects sont : les relations spatiales, les relations temporelles, le contexte d'insertion et l'attention de l'utilisateur. Un système peut être constitué de plusieurs espaces d'interaction combinés, un espace d'interaction étant relatif à la réalisation d'une tâche donnée. Comme les notations ci-dessus, cette approche concerne particulièrement la conception de l'interaction par l'identification de questions que le concepteur doit traiter en phase de conception.

Approches par composants

Deux approches se placent dans cette catégorie : DWARF [1] et AMIRE [6]. DWARF est une plate-forme à composants. Chaque composant est un module qui fournit et requiert un ou plusieurs services. Une application est donc formée de composants reliés entre eux grâce à un mécanisme de gestion des services. AMIRE adopte la même démarche en proposant un mécanisme de réutilisation de solutions existantes (composants, bibliothèques, fonctions, boîtes à outils, etc.) pour des problèmes spécifiques d'implémentation qui reviennent régulièrement dans les systèmes mixtes.

Ces deux solutions, même si elles prennent en compte plusieurs rôles dans le processus de développement (concepteur, développeur de composants, utilisateur final) restent très axées sur l'architecture logicielle et la conception logicielle au détriment de la conception de l'interaction et des problèmes ergonomiques inhérents.

Patrons de conception

Enfin, [8] propose un ensemble de cinquante patrons de conception pour la RA regroupés au sein d'un modèle de référence des systèmes de RA. Ce modèle comprend six groupes de patrons de conception qui sont : localisation (*tracking*), contexte, modèle du monde, interaction, présentation et application.

Toutes ces approches partagent des objectifs et des concepts généraux à la Réalité Augmentée (*tracking*, re-

lations spatiales, contexte, etc.). Cependant, la plupart d'entre elles, hormis les notations de conception, sont dédiées à l'architecture et la conception logicielle au sein du processus de développement. Par ailleurs, aucune des approches présentées ne couvre tout le processus de développement du système. Enfin peu d'entre elles considèrent vraiment l'utilisabilité du système final.

Notre approche est résolument centrée sur la conception de l'interaction et vise à identifier des solutions de conception génériques qui serviront ensuite d'éléments logiciels de base à la conception logicielle. Notre objectif est également d'intégrer une prise en compte de critères ergonomiques dès la conception de l'interaction, pour faciliter ou guider le choix du concepteur parmi plusieurs solutions de conception.

METHODE

Dans nos travaux, nous adoptons une approche de conception descendante, de l'interaction abstraite à l'interaction concrète, que nous construisons en intégrant la notation ASUR. ASUR est un acronyme pour : Adapter, System, User, Real Objects. Cette notation permet de représenter graphiquement l'interaction concrète pour une tâche donnée. Un diagramme ASUR regroupe donc différentes entités qui prennent part à l'interaction principalement en s'échangeant des données. Entités et relations sont définies par un certain nombre de caractéristiques. Entre l'utilisateur (entité U) et le système (entité S), les adaptateurs servent de point de jonction entre les deux mondes : réel et numérique. Les adaptateurs A_{in} sont des dispositifs d'entrée (une souris, un micro, une pédale de commande), les entités A_{out} désignant des dispositifs de sortie (un écran, un casque de RA, des haut-parleurs, un vidéo projecteur). Le dernier type d'entité constituant un diagramme ASUR caractérise les objets réels (entité R). Les objets réels appartiennent à l'environnement physique de l'utilisateur et prennent également part à l'interaction. Un objet réel est soit objet de la tâche (R_{object}), soit outil de la tâche (R_{tool}). A titre illustratif, la Figure 1 est la représentation ASUR de l'interaction lors d'une tâche de ponction péricardique [3] avec transmission de données de guidage du système (S) vers le chirurgien (U).

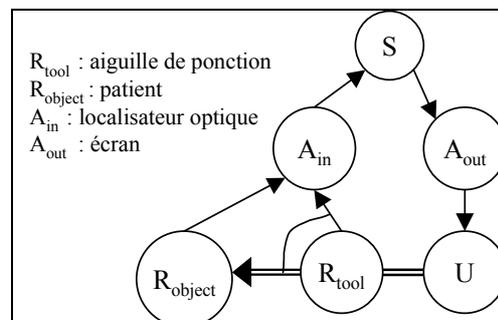


Figure 1 : Exemple de diagramme ASUR

Dans notre démarche de travail, tandis qu'ASUR est dédiée à l'interaction concrète, l'un de nos premiers objectifs est de fournir un espace de conception pour l'interaction abstraite, première phase d'une méthode de conception descendante. Pour cela, nous sommes partis des résultats du projet MMM (interface Médecin-Malade-Machine). En effet l'approche décrite dans [4] a permis d'identifier des fonctions génériques d'interaction, communes à plusieurs applications de chirurgie assistée par ordinateur. S'appuyant sur ces fonctions, notre approche a consisté à abstraire ces fonctions d'interaction des technologies qu'elles emploient. Nous obtenons donc un ensemble de situations d'interaction génériques (indépendantes de l'application) et abstraites (indépendantes de la mise en œuvre technique). Une situation d'interaction est un diagramme ASUR où l'on ne prend pas en compte les caractéristiques des entités et des relations. Seuls les échanges de données entre les différents types d'entité sont représentés. L'ensemble de ces situations d'interaction forme notre premier espace de conception décrit dans la section suivante.

RESULTATS

Résultats conceptuels

Notre espace de conception est composé de huit situations d'interaction. Ces situations d'interaction sont indépendantes de toute spécialité clinique et de toute modalité d'interaction spécifique. Cela permet au concepteur d'explorer l'ensemble des solutions de conception sans être limité par la technologie à employer. Pour structurer notre espace de conception, nous distinguons les situations en entrée (de l'utilisateur vers le système) des situations en sortie (du système vers l'utilisateur). Nous distinguons également si les situations d'interaction sont directes ou indirectes. Une situation est qualifiée de directe si elle implique l'objet de la tâche dans la relation entre l'utilisateur (U) et le système (S). Elle est dite indirecte si l'objet de la tâche n'est pas présent dans cette relation. Les huit situations d'interaction sont présentées dans la liste ci-dessous.

Situations d'interaction en entrée indirectes

- $U \rightarrow A_{in} \rightarrow S$
- $U \rightarrow R_{tool} \rightarrow A_{in} \rightarrow S$

Situations d'interaction en entrée directes

- $U \rightarrow [R_{tool}, R_{object}] \rightarrow A_{in} \rightarrow S$
- $U \rightarrow R_{object} \rightarrow A_{in} \rightarrow S$

Situations d'interaction en sortie indirectes

- $S \rightarrow A_{out} \rightarrow U$
- $S \rightarrow A_{out} \rightarrow R_{tool} \rightarrow U$

Situations d'interaction en sortie directes

- $S \rightarrow A_{out} \rightarrow [R_{tool}, R_{object}] \rightarrow U$
- $S \rightarrow A_{out} \rightarrow R_{object} \rightarrow U$

Cet espace, couvrant toutes les relations possibles entre U et S, est donc complet. Nos situations d'interaction constituent par conséquent des schémas de base qui peu-

vent être affinés ou même combinés [9] comme dans les applications de télé chirurgie où il est possible d'insérer une situation d'interaction dans une autre. Notre espace constitue donc une base de travail dans une démarche de conception descendante de l'interaction. Pour concevoir l'interaction concrète, le concepteur part de la tâche à réaliser. Pour cette tâche, elle/il explore l'espace de conception pour identifier les situations d'interaction les plus adaptées. Puis elle/il concrétise les différentes situations d'interaction en précisant les caractéristiques des entités et des relations ASUR, c'est-à-dire en choisissant les modalités d'interaction : les dispositifs et les langages d'interaction. L'espace de situations d'interaction permet ainsi de faire un premier choix en fonction de la nature de l'interaction. Les différentes situations d'interaction servent également de moyen de comparaison lorsque le concepteur hésite entre deux interactions plus spécifiques. L'interaction concrète décrite, il convient ensuite de développer les solutions logicielles et matérielles correspondantes.

Pour illustrer notre démarche, nous considérons un cas précis : l'application de ponction péricardique CASPER [3]. Lors de la tâche de ponction, le système localise la position de l'aiguille de ponction par rapport au cœur du patient. Le système met alors en correspondance cette position courante avec une trajectoire de ponction pré-établie sur un modèle numérique du cœur du patient. Il présente ensuite, de manière graphique, les données de guidage qui permettent au chirurgien de respecter au mieux la trajectoire prévue. Dans une première version, les données de guidage sont présentées à l'utilisateur sur un écran standard. Cela correspond à la situation d'interaction en sortie $S \rightarrow A_{out} \rightarrow U$. L'évaluation ergonomique de cette version a soulevé le problème de discontinuité perceptuelle identifié dans [2]. En effet, le chirurgien est en permanence obligé de détourner la tête du champ opératoire pour observer les données de guidage. Nos autres situations d'interaction en sortie amènent différentes réponses à ce problème, en proposant que les données soient mises en correspondance :

- avec l'outil ($S \rightarrow A_{out} \rightarrow R_{tool} \rightarrow U$),
- avec l'outil et l'objet ($S \rightarrow A_{out} \rightarrow [R_{tool}, R_{object}] \rightarrow U$),
- avec l'objet de la tâche seul ($S \rightarrow A_{out} \rightarrow R_{object} \rightarrow U$).

Par exemple, la mise en correspondance des données de guidage avec l'objet de la tâche correspond à la projection vidéo de ces données sur le corps du patient. Un retour d'effort dans la main de l'utilisateur qui tient l'outil, comme dans le système PADyC [12] est une illustration d'une mise en correspondance des données de guidage avec l'outil de la tâche.

L'étude de ces situations d'interaction nous a incités à étudier la mise en correspondance des données de guidage avec l'outil (ici une aiguille de ponction). Pour rendre concrète cette interaction, nous avons choisi d'utiliser un mini-écran LCD fixé à l'aiguille et qui agi-

rait comme une sorte de viseur. Nous adoptons alors la stratégie d'augmentation [7] d'un objet physique particulier, l'outil de la tâche.

Cet exemple concret a montré l'intérêt en conception de notre espace de situations. Il nous a aussi permis de cerner la difficulté de passer d'une situation abstraite à une interaction concrète et d'identifier le besoin d'espaces de caractérisation des modalités.

Ainsi, comme aide à la conception concrète, nous avons établi un deuxième espace de conception spécifique au mini-écran [9]. Cet espace caractérise les modalités d'interaction qui exploitent ce dispositif particulier. L'espace est constitué de 2 axes : (i) caractérisation des informations à afficher, (ii) caractérisation de l'utilisation du dispositif. Les différentes caractéristiques de ces deux axes ont été choisies en fonction des résultats des travaux effectués sur les interfaces « incarnées » [5].

Résultats logiciels

Une première version du module d'affichage dans le mini-écran a été développée afin de mener à court terme des évaluations ergonomiques en intégrant cette interface à l'application CASPER et à une application de ponction rénale.

TRAVAUX FUTURS

A court terme, nous réaliserons une évaluation ergonomique de la solution du mini-écran intégrée dans une application de ponction. Cette solution innovante sera comparée avec la version classique de l'application.

A plus long terme, nous souhaitons enrichir notre espace de situations d'interaction en y intégrant des critères ergonomiques pour guider le concepteur. Ceci permettra par exemple de souligner l'apparition de problèmes de discontinuité. Concernant la conception logicielle, nous identifierons les composants logiciels nécessaires aux solutions de l'espace du mini-écran. Ces composants logiciels constitueront ainsi les briques de bases d'une boîte à outils.

REMERCIEMENTS

Mes remerciements vont à Laurence Nigay et Jocelyne Troccaz (responsables de cette thèse) et aux membres des équipes IIHM et GMCAO pour leurs conseils. Ces travaux ont été en partie financés dans le cadre du projet RNTL MMM.

BIBLIOGRAPHIE

1. Bauer et al. Design of a Component-Based Augmented Reality Framework, *In Proceedings of The Second IEEE and ACM International Symposium on Augmented Reality (ISAR 2001)*.

2. Chalon, R., David, B. IRVO: an Architectural Model for Collaborative Interaction in Mixed Reality Environments, *The International Workshop on Exploring the Design and Engineering of Mixed Reality Systems MIXER'04*, January 2004, Funchal, Portugal, pp. 35-42.
3. Dubois, E. "Chirurgie Augmentée, un Cas de Réalité Augmentée ; Conception et Réalisation Centrées sur l'Utilisateur", Thèse de l'Université Joseph Fourier, Grenoble, (2001), 275 pages.
4. Dubois, E. et al. Analytical Approach for the Design of the Surgeon's Interaction with a Computer Assisted Surgery System. *Actes de la conférence Surgetica 2002*, (2002), Grenoble, France, Sauramps Medical, pp. 211-216.
5. Fishkin, K.P., Moran, T.P. et Harrison, B.L. Embodied User Interfaces: Towards Invisible User Interfaces. *In Proceedings of EHCI'98*, (1998), Kluwer Academic Publ., pp. 1-18.
6. Grimm P. et al. AMIRE - Authoring Mixed Reality, *The First IEEE International Augmented Reality Toolkit Workshop*, Darmstadt, Germany, 29 September, 2002.
7. Mackay, W.E. Augmented Reality: linking real and virtual worlds, *In Proceedings of ACM AVI '98, Conference on Advanced Visual Interfaces*, L'Aquila, Italy: ACM Press.
8. MacWilliams, A. et al. Design Patterns for Augmented reality Systems, *In Proceedings of MIXER'04*, (2004), Funchal, Portugal, pp. 64-71.
9. Mansoux, B, Nigay, L. et Troccaz, J. Abstract and Concrete Interaction with Mixed Reality Systems / The case of the mini-screen, a new interaction device in Computer-Assisted Surgery, *In Proceedings of MIXER'04*, (2004), Funchal, Portugal, pp. 12-18.
10. Renevier, P., Nigay, L. Notation de conception pour les Systèmes Mixtes Collaboratifs et Mobiles, *Actes de Mobilité et Ubiquité 2004*, (2004), pp. 66-73.
11. Trevisan, D., Vanderdonckt J. et Macq, B. Designing Interaction Space for Mixed Reality Systems *In Proceedings of MIXER'04*, (2004), Funchal, Portugal, pp. 27-34.
12. Troccaz, J., Delnondedieu, Y. *Semi-Active Guiding Systems in Surgery: A Two-DOF Prototype of the Passive Arm with Dynamic Constraints (PADyC)*. *Mechatronics* (1996), 6(4), Elsevier Science, pp. 399-421.



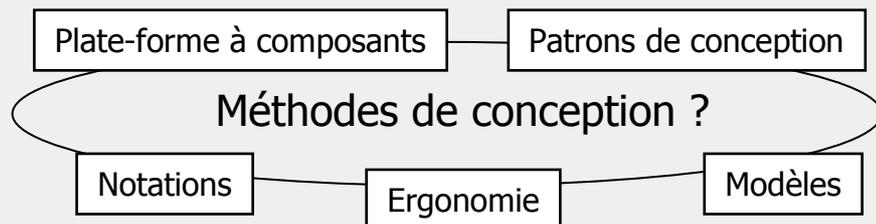
Méthodes de conception pour les systèmes de Réalité Augmentée

Benoît Mansoux

Motivations

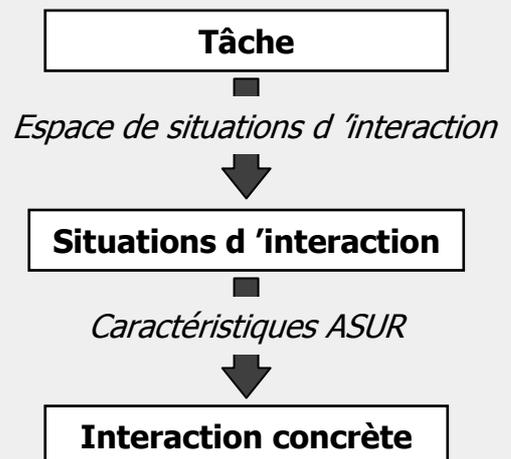
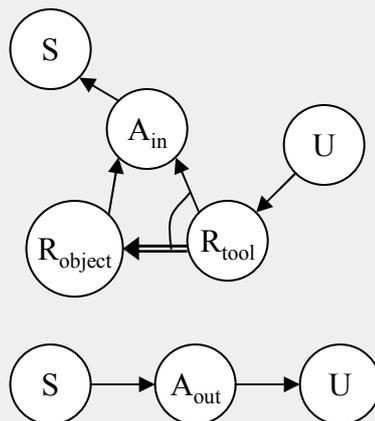
• Un domaine d'application : la Chirurgie Augmentée

- Diversité des applications cliniques
- Diversité des techniques d'interaction
- Solutions d'interaction ad-hoc



Réalisations

• Situations d'interaction



• Cas d'étude : Interaction avec un mini-écran

- Discontinuité perceptuelle
- Interaction innovante
- Utiliser notre approche

Espace de conception spécifique

- Données en sortie
- Usage du mini-écran

Développement d'un composant Mini-écran



application CASPER

Perspectives

Evaluation ergonomique de la solution mini-écran

Vérification ergonomique à partir des situations d'interaction

Conception de l'interaction → Conception logicielle

Validité d'une évaluation in situ via la méthode du Magicien d'Oz

Merlin Bruno

IRIT – DIAMANT
118 Route de Narbonne
31062 Toulouse
05.61.55.63.05
merlin@irit.fr

RÉSUMÉ

Dans le cadre d'évaluations de dispositifs à commandes multimodales sur Pocket PC et d'études d'usage in situ de la multimodalité, nous avons jugé opportun d'étudier la possibilité d'utiliser la méthode du « Magicien d'Oz » en milieu non contrôlé. Des pré-expérimentations, destinées à évaluer l'usage de claviers logiciels multimodaux sur Pocket PC, nous ont amenés à tester la validité de cette approche Magicien d'Oz en contexte d'observation *in situ*.

Mots clefs : Magicien d'Oz, mobilité, multimodalité, méthodologie d'évaluation.

ABSTRACT

In the context of studies dealing with the use of multimodality in a non-controlled environment, we decided to try to apply the Wizard of Oz methodology in a context *in situ*. Pre-experimentation, supposed to evaluate multimodal software keyboard for Pocket PC, brought us to test the validity of this methodology in this observation context.

CATEGORIES AND SUBJECT DESCRIPTORS : H.5 [CHI], J.4 [SOCIAL AND BEHAVIORAL SCIENCES].

GENERAL TERMS : Experimentation, Human Factors, Verification

Keywords : Wizard of Oz, mobility, multimodality, evaluation methodology.

INTRODUCTION

Des techniques d'interaction comme la reconnaissance vocale ou la reconnaissance gestuelle et la multimodalité

présentaient, initialement, le charme de leur nouveauté. Elles ont été confrontées aux contraintes et restrictions liées à leur usage (cf. les 10 Mythes d'Oviatt [12]) et ne se sont finalement que peu développées dans les applications grand public. Le développement des technologies d'informatique mobile, à travers la démocratisation des PDAs, l'omniprésence et la poly-potence des téléphones, ou encore l'apparition des *wearable computers*, repose la question de l'usage des modalités d'interaction. Des modalités qui ne se présentaient que dans de rares contextes comme une alternative efficace aux périphériques d'interaction classiques (claviers/souris en entrée, écran en sortie), pourraient obtenir une importance nouvelle en contexte de mobilité. La multimodalité pourrait ainsi permettre de palier le handicap, en terme d'interaction, procuré par le contexte de mobilité et la taille réduite des IHMs.

Consécutivement à cet essor des technologies d'informatique mobile, un volet d'études, concernant l'usage de la multimodalité en situation de mobilité, a vu le jour. Cependant, l'implémentation des technologies d'interaction, nécessitant un investissement technique et financier non négligeable, reste contraignante, incitant à mener des expérimentations sous la forme de séance de Magicien d'Oz [1].

L'évaluation d'un dispositif destiné à un usage *in situ* nous a amené à sortir la méthodologie du Magicien d'Oz de son contexte normal d'utilisation pour expérimenter la réalisation d'une séance de Magicien d'Oz en milieu non contrôlé. Nous avons donc mis en place une pré-expérimentation destinée à étudier les facteurs permettant de favoriser le bon déroulement et le respect des contraintes méthodologiques, ainsi que l'obtention des données relatives à l'expérimentation.

MULTIMODALITE & EVALUATION

En matière d'étude des usages de la multimodalité, la technique du Magicien d'Oz présente l'avantage principale de permettre une simulation des modalités d'interactions

qui, pour des raisons techniques ou pratiques, peuvent ainsi rester non implémentées [13]. Un « Compère » situé dans une pièce voisine communique à distance, soit directement avec l'utilisateur (dans le cadre de la simulation d'un système vocal interactif [9], [7], par exemple) ou avec l'application de l'utilisateur, en répercutant à distance les commandes effectuées par l'utilisateur (applications multimodales par exemple [11]). Dans les deux cas, cette méthode implique que le compère puisse soit voir, par l'intermédiaire de caméras, soit seulement entendre l'utilisateur de manière à percevoir l'information nécessaire à la simulation.

Plusieurs contraintes sont cependant induites par la séance de Magicien d'Oz, l'optique étant de permettre au « Compère » d'opérer la simulation de l'usage des modalités sans que l'utilisateur sujet d'expérimentation ne soit conscient de cet aspect simulé. Le but est que l'utilisateur interagisse le plus naturellement possible avec le dispositif évalué [7]. En effet, des études ont montré que la connaissance de l'aspect simulé de l'interaction, notamment pour l'interaction en langage naturel, influe de façon importante le vocabulaire utilisé par l'utilisateur [3].

Les contraintes méthodologiques se posent ainsi :

- En terme d'observation : sans être intrusif, comment accéder aux données pour opérer la simulation, comment récolter les données de l'observation ;
- En terme de communication : comment simuler l'interaction de l'utilisateur.

Dans un grand nombre de projets existants traitant de l'étude de la multimodalité en situation de mobilité, comme l'évaluation d'un système vocal d'aide au guidage [10], ou l'étude de l'interaction naturelle dans une voiture [2] (VICO), ou encore dans le cadre de NIM [5], les contraintes liées à la séance de magicien d'Oz ont été résolues par la limitation de l'observation à l'étude en milieu contrôlé. Les utilisateurs étaient placés dans un milieu techniquement adapté au Magicien d'Oz.

Nous avons évalué la validité de l'application d'une séance *in situ* de Magicien d'Oz à travers la réalisation d'une pré-expérimentation. Seuls les résultats concernant la méthodologie seront abordés dans le cadre de cet article.

METHODE

Tâche & Support d'expérimentation

Le support de l'expérimentation est un éditeur de texte muni d'un clavier logiciel développé sur Pocket PC. Le clavier logiciel est de type *fullscreen keyboard* [12], dont les touches sont accessibles par l'intermédiaire des deux pouces, auquel a été ajouté une liste de complétion.

Deux modes de saisie sont envisagés par l'éditeur de texte : un mode de saisie susceptible de favoriser à proprement parler l'activité de saisie ; et un mode classique (cf. Figure 1) où l'utilisateur est plutôt supposé réaliser des tâches d'édition.

Langage d'interaction multimodale

Plusieurs modalités d'interaction, destinées à améliorer l'accessibilité de certaines fonctions du clavier, ont été ajoutées au dispositif. Celles-ci sont soit implémentées, soit simulées par le compère.

Une modalité *embodied* [13] (tangible dont l'artefact est l'ordinateur lui-même) permet, tout d'abord, de passer d'un mode de saisie à l'autre. Cette modalité se justifie par la nécessité pour l'utilisateur d'effectuer une rotation lorsqu'il souhaite passer du mode de « saisie » au mode d'« édition » (cf. Figure 1). Elle permet à l'utilisateur : de joindre une commande à la manipulation de l'artefact et de ne pas nécessiter d'interaction supplémentaire pour activer le clavier. Comme nous ne disposons pas d'un module de reconnaissance de mouvement, celle-ci sera en conséquence simulée. A noter qu'il existe sur le marché un périphérique de reconnaissance tangible susceptible d'être adapté à cet usage [15] (cf. Figure 2).

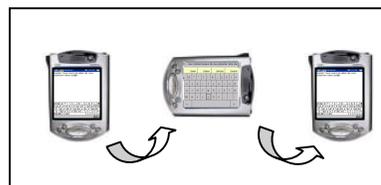


Figure 1 : Utilisation de la reconnaissance tangible



Figure 2 : ECER TiltControl

La reconnaissance vocale, également simulée, permet de réaliser des opérations d'édition et de sélection dans la liste de complétion. Ces commandes permettent de supplanter une et surtout plusieurs opérations tactiles (comme les opérations d'édition. Ex : ctrl + c, ou ouverture d'un menu suivi de la sélection d'un item du menu). Une modalité de reconnaissance de geste 2D (cf Figure 3), implémentée, offre des possibilités équivalentes aux commandes offertes par la reconnaissance vocale.

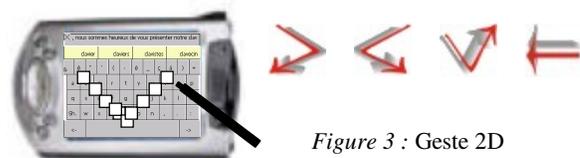


Figure 3 : Geste 2D

Lors de l'expérimentation, la tâche de l'utilisateur est de relever, dans le campus universitaire, des informations affichées sur les panneaux d'information.

Architecture : observation et compère

Compère. Plusieurs impératifs viennent contraindre le développement du poste du compère. Le compère doit pouvoir observer visuellement l'utilisateur, a priori de façon médiée ou non, dans le but de simuler l'interaction *embodied*. La prohibition de moyens invasifs, comme les caméras, sous-entend l'impossibilité d'une interaction visuelle médiée et la nécessité d'une interaction visuelle directe entre le compère et l'utilisateur. L'application du compère se doit donc d'être embarquée. Développée sur Pocket PC, l'application du compère communique avec l'application de l'utilisateur par WiFi. Le compère peut ainsi rester près de l'utilisateur le long de son parcours.

Le nombre réduit de commandes à simuler (commandes sans sémantique complexe) permet d'exploiter simplement la totalité de l'interface du Pocket PC. Celle-ci est divisée, en fonction du mode actif (saisie ou édition), en un nombre limité de boutons représentant les commandes vocales. Un bouton supplémentaire change le mode actif, simulant parallèlement l'interaction *embodied* chez l'utilisateur.

Le contact presque direct de l'utilisateur et du compère soulève la question de l'anonymat du compère et pourrait remettre en cause la validité de la méthodologie. Nous expliquerons plus loin l'approche que nous avons mise en œuvre pour contourner ce problème.

Observation des données. De façon à pouvoir être collectées de manière non intrusive, les données de l'interaction entre l'utilisateur et l'éditeur de texte (éditeur + clavier) ainsi que les interactions médiées entre le compère et l'utilisateur sont acquises automatiquement par les supports logiciels. Les traces de l'interaction sont ensuite transmises sous forme de messages textuels, par réseau, à une plate-forme d'observation distante. Cette plate-forme d'observation capture et présente les données de l'interaction en vue d'analyses qualitatives et statistiques.

Évaluation de la méthodologie

La pré-expérimentation a engagé 10 personnes. Ces personnes étaient toutes issues du domaine de l'IHM et familières des techniques de Magicien d'Oz. L'origine des personnes constituait un risque supplémentaire d'échec et d'invalidation de la méthode. En effet, une personne au courant de l'état de l'art ainsi que des techniques courantes d'évaluation est plus spontanément portée à soupçonner l'activité du compère et le contexte Magicien d'Oz. Nous devons par conséquent trouver un rôle et une justification à la présence du compère. Deux possibilités se sont offertes : présenter le compère comme un autre utilisateur passant simultanément l'évaluation sur un autre prototype ; ou bien désigner le compère comme une personne accompagnatrice

destinée à prendre de simples notes et veiller au bon déroulement de l'expérimentation.

La première solution présentait plusieurs intérêts, et particulièrement celui de détacher complètement l'utilisateur du sentiment d'être observé. Cependant cette solution risquait aussi de motiver une communication de l'utilisateur vers le compère, cette communication pouvant être dommageable tant pour le rôle du compère que pour l'activité de l'utilisateur. Cette solution interdit par ailleurs l'omniscience du compère et son pouvoir d'apporter une solution en cas de problème technique ou circonstanciel.

Le compère a été finalement introduit comme un observateur (cf. Figure 7) destiné à guider l'utilisateur dans le campus et recueillir d'éventuelles remarques spontanées de la part de celui-ci. Un bouton physique du Pocket PC permet en outre au compère de pouvoir masquer son interface par une fausse fenêtre de prise de notes pré-remplie.

Utilisateur

Compère



Figure 7 : Déroulement de la séance de magicien d'Oz

Afin d'écartier le plus possible les soupçons qui auraient pu peser sur le rôle exact du compère, les consignes données aux utilisateurs insistaient sur la nécessité de s'appliquer à la réalisation des interactions sur des périphériques peu fiables. Après le déroulement de la pré-expérimentation, un débriefing a été effectué avec les utilisateurs. Nous occulterons ici les résultats concernant le reste de la pré-expérimentation pour centrer nos remarques sur l'évaluation de la méthodologie Magicien d'Oz.

Ce débriefing était tout d'abord destiné à obtenir des informations sur leurs connaissances des techniques de Magicien d'Oz. Le débriefing devait ensuite permettre de déterminer la conscience que les utilisateurs avaient eue de l'aspect simulé de l'expérimentation. Il a été réalisé sous la forme d'un questionnaire destiné à amener progressivement l'utilisateur sur la piste de la séance de Magicien d'Oz. L'aspect progressif des questions permettait de mesurer le degré de « conscience » qu'avait l'utilisateur de l'aspect simulé des interactions.

RESULTATS

Malgré le rapport des utilisateurs à l'IHM et leurs connaissances approfondies des techniques de Magicien

d'Oz. La majorité des utilisateurs (7 sur 10) n'a pas soupçonné l'usage de celle-ci.

Le facteur ayant permis à deux des trois personnes d'identifier l'usage de la méthode de Magicien d'Oz était une très bonne connaissance des PDAs et des dispositifs permettant l'interaction tangible. Cette connaissance leur a permis d'identifier que le Pocket PC n'était pas équipé pour permettre une reconnaissance *embodied* et de déduire le rôle du compère. Dans une expérimentation future utilisant cette méthodologie d'évaluation, il sera donc impératif de tenir compte de cet aspect et d'éventuellement installer un leurre sur le PDA.

Le facteur ayant permis à la troisième personne de soupçonner simplement l'usage du Magicien d'Oz était lié aux trop bonnes performances de la reconnaissance vocal relativement au cadre bruyant de l'environnement.

La proximité du compère avec les utilisateurs, dans un contexte naturel et public (induisant la présence d'autres personnes extérieures à l'expérimentation), ne s'est pas montrée intrusive. Elle a ainsi permis à celui-ci de remplir convenablement son rôle. Par ailleurs, les facteurs qui semblent avoir favorisé le fait que 7 personnes sur les 10 n'aient pas soupçonné la séance de Magicien d'Oz ont été les consignes données lors de la prise en main du dispositif et l'aspect naturel de l'interaction, renforcé par le cadre naturel du déroulement de l'expérimentation. Le contexte non contrôlé et hors laboratoire, ainsi que l'absence de dispositifs ostensibles d'observation ont permis à l'utilisateur de pouvoir faire abstraction du cadre expérimental de son activité.

CONCLUSION & PERSPECTIVES

On doit préalablement noter que nos contraintes expérimentales étaient assez restreintes. Restreintes en matière d'observation, dans la mesure où les données statistiques acquises au moyen de la plate-forme d'observation étaient suffisantes et où nous n'avions donc pas besoin d'enregistrements vidéos, et restreintes en matière de langage d'interaction à simuler. Celui-ci étant fini et restreint, l'application du compère et son usage en situation de mobilité étaient simplifiés.

Cependant, bien que nous ne puissions généraliser l'usage de cette méthode, nous avons mis en évidence non seulement la possibilité de réaliser des séances de Magicien d'Oz en contexte *in situ*, mais également le bénéfice apporté par le contexte *in situ* sur la méthodologie elle-même. Elle nous engage également à privilégier un dispositif simple d'observation, intégré au milieu naturel, par rapport à un dispositif intrusif contraignant physiquement l'utilisateur et susceptible de le transformer en objet de curiosité pour son environnement.

REMERCIEMENTS

Je remercie J. Phalip, S. Chamberlhac et L. Benech, (ex-) étudiants en DESS IHM pour l'aide qu'ils m'ont apportée dans l'élaboration du support de cette expérimentation.

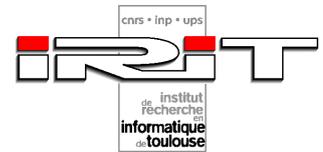
REFERENCES

1. Amalberti, R., Valot, C. Le Magicien d'Oz, CERMA, 1992.
2. Bensen, N. O., Dybkjær, L. Exploring natural interaction in the car. Workshop on Information Presentation and Natural Multimodal Dialogue, Verone, décembre 2001, pp. 75-79.
3. Brennan, S.E. "Conversation as Direct Manipulation: An Iconoclastic View". In "The Art of Human-Computer Interface Design", B. Laurel ed., 1990.
4. Buisson, M., Bustico, A., Chatty, S., Colin, F.R., Jestin, Y., Maury S., Mertz, C., Truillet, P. Ivy : un bus logiciel au service du développement de prototypes de systèmes interactifs. IHM'02, Poitiers.
5. Calvet, G., Kahn, J., Zouinar M., Salembier P., Briois J-C., Nigay, L., Rey., L. Pasqualetti Étude empirique de l'usage de la multimodalité avec un ordinateur de poche. IHM-HCI2001, CEPADUES Publ., pp. 5-8.
6. Dahlbäck, N., Jönsson, A. "Empirical studies of discourse representations for natural language interfaces". Chapter of the ACL, 291-8, 1989.
7. Dahlbäck, N., Jönsson, A., Ahrenberg, L. Wizard of Oz Studies-Why and How. IUI, 1993.
8. Fraser, N. M., Gilbert G. N. Simulating speech systems. Computer Speech and Language, 1991.
9. Fraser, N., Gilbert N., McDermid, C. "The Value of Simulation Data". ANLP, Trento, Italie, avril 1992.
10. Manstetten, D., Krautter, W., Grothkopp, B., Steffens, F., Geutner, P. Using A Driving Simulator To Perform A WOZ Experiment On Speech-Controlled Driver Information Systems. HCTSC 2001.
11. Nigay, L., Coutaz, J. A Design Space For Multimodal Systems: Concurrent Processing and Data Fusion Laurence. INTERCHI '93, 1993, pp. 172-178.
12. Oviatt, S.L. Ten myths of multimodal interactions. ACM Vol. 42, N°11, Novembre, 1999, pp. 74-81.
13. Salber, D., Coutaz, J., Applying the Wizard of Oz Technique to the study of Multimodal Systems. HCI'93, août 93.
14. Spb Software House, SPB Full Screen Keyboard, <http://www.spbsoftwarehouse.com/products/fsk/?en>
15. TiltControl, ECERTech. <http://www.ecertech.com/tiltcontroldevice.aspx>

Validité d'une évaluation in situ via la méthode du Magicien d'Oz

IRIT – DIAMANT
118 Route de Narbonne
31062 Toulouse

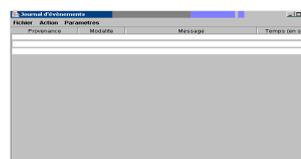
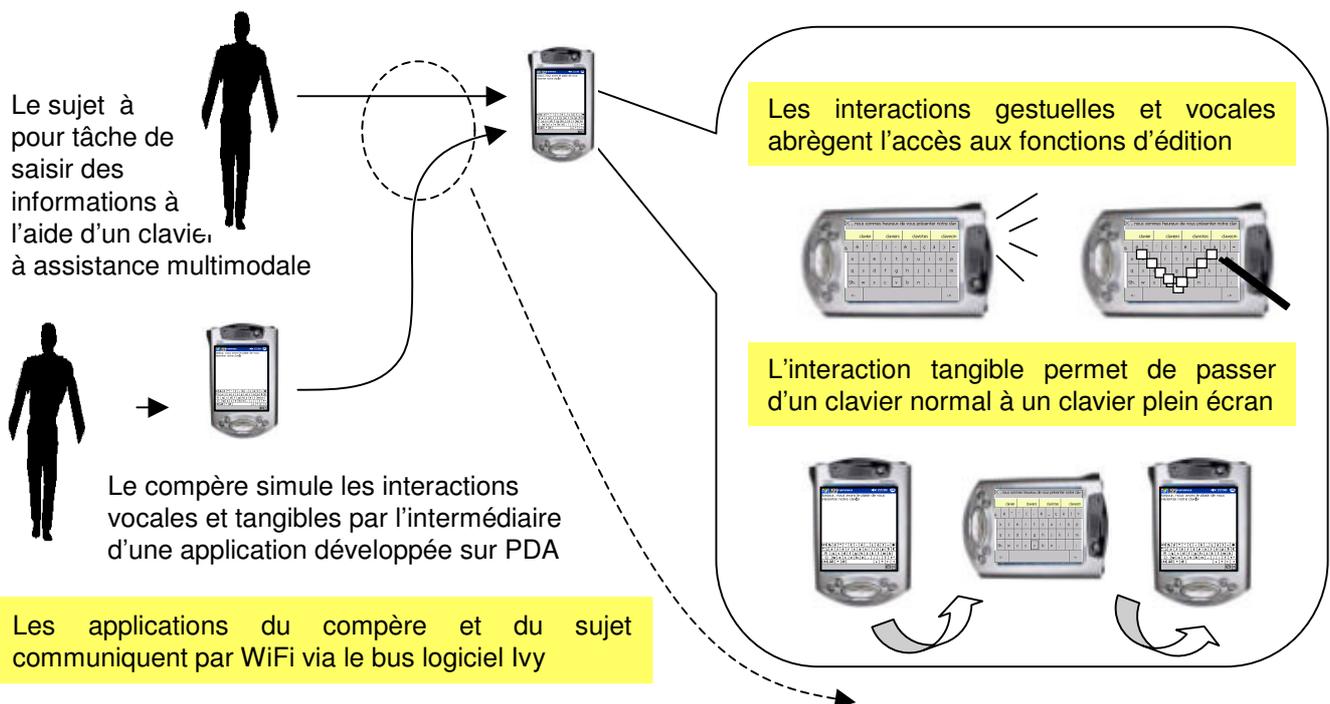
Merlin Bruno
merlin@irit.fr



La présence du compère peut-elle gêner l'activité du sujet ?

Le compère peut-il jouer son rôle sans que celui-ci puisse être identifiée ?

Peut-on récolter les données nécessaires à l'observation ?



Les traces de l'interactions sont acquises logiciellement de manière non intrusive

Le compère (C) peut, tout en restant à distance raisonnable du sujet (S), à la fois observer l'interaction opérée par celui-ci et jouer son rôle, sans pour autant interférer avec la tâche du sujet

Les expérimentations, menées sur 10 sujets travaillant dans le domaine de l'IHM, montrent que la méthodologie du magicien d'Oz peut être, au moins dans certains contextes, utilisée *in situ* et en mobilité.

7 des 10 sujets pourtant familiarisés avec les techniques de magicien d'Oz, n'ont pas perçu que l'interaction était simulée par le compère.

Les données de l'interaction ont été capturées logiciellement sans caméras.

KeyGlasses : Des touches semi-transparentes pour optimiser la saisie de texte

Mathieu Raynal

Equipe DIAMANT
IRIT UMR CNRS 5505
118, Route de Narbonne
31062, Toulouse Cedex 4, France
raynal@irit.fr

RESUME

Cet article présente un système d'optimisation pour claviers logiciels. Celui-ci repose sur l'ajout dynamique de touches semi-transparentes. Ces dernières sont ajoutées autour de la touche qui vient d'être pressée. Les lettres sont déterminées en fonction de ce qui a été saisi précédemment grâce à un système de prédiction. Les résultats a priori de ce système laisse espérer un gain d'au moins 30% de la vitesse de saisi.

MOTS CLES : clavier logiciel, système de prédiction, transparence

ABSTRACT

This article presents a system of optimization for soft keyboard. It rests on the dynamic addition of semi-transparent keys. These one are added around the key witch has just been typed. The letters are given according to keyboarding letters previously and thanks to a predictive system. Results a priori of this system let hope a profit from at least 30% of the text entry speed.

KEYWORDS: soft keyboard, transparency, predictive system

INTRODUCTION

A l'origine, les claviers logiciels ont été créés pour permettre aux personnes handicapées moteur d'utiliser un ordinateur. En effet grâce à ces claviers intégrés au système, ces personnes peuvent via un dispositif de pointage saisir du texte. Avec l'émergence des dispositifs mobiles dépourvus de clavier physique (par exemple, les assistants personnels), les claviers logiciels sont maintenant aussi utilisé sur ce type de système à l'aide d'un stylet.

Cependant les performance d'un clavier physique de type AZERTY sont très nettement supérieures à celles de son homologue logiciel. Ceci peut être expliqué par le fait que l'on peut saisir du texte avec plusieurs doigts sur un clavier physique ce qui limite les déplacements entre chaque touche. Limitée au seul dispositif de pointage, la saisie avec un clavier logiciel voit ses performances diminuer considérablement à cause du temps de déplacement d'une touche à l'autre.

Ainsi pour palier ces faibles performances, des recherches sont menées pour essayer d'améliorer la vitesse de saisie de texte sur clavier logiciel. La principale problématique consiste à ne pas ajouter de charge cognitive trop forte à l'utilisation du clavier logiciel, afin de ne perturber l'utilisateur.

Dans cette optique, nous proposons un système basé sur des touches ajoutées en fonction du contexte afin d'améliorer la vitesse de saisie de texte. Ces touches sont semi-transparentes et de couleur vive pour attirer l'œil de l'utilisateur sans lui faire perdre ses repères par rapport au clavier initial.

Au travers de cet article, nous montrerons en quoi consiste les recherches actuelles en optimisation de claviers logiciels. Puis nous expliquerons comment évaluer ces systèmes. Enfin nous decrirons notre système KeyGlasses, les premiers résultats théoriques de celui-ci et les travaux à venir.

TECHNIQUES D'OPTIMISATION

Optimiser la disposition des touches

Un des axes de recherche consiste à essayer d'optimiser la disposition des touches. Ainsi en placant côte à côte les touches qui ont le plus de chance de se succéder, on diminue les distances parcourues lors de la saisie et par conséquent le temps de cette dernière.

Plusieurs claviers ont été réalisés dans cette optique. Certains ont été conçus en utilisant les tables bi-grammes donnant les probabilités pour qu'une lettre succède à une

autre. On peut notamment citer le clavier commercialisé Fitaly¹ ou encore le clavier OPTI [4] (cf. fig. 2). D'autres claviers mettent en jeu en plus des modèles linguistiques des techniques normalement utilisées en physique telles que l'algorithme Metropolis qui a permis de concevoir le clavier de même nom [9].



Fig. 1 : Clavier OPTI [2]

Une autre façon de limiter les déplacements consiste à mettre plusieurs lettres sur la même touche comme par exemple [2]. Ces claviers sont dits ambigus. Bien que pratiques pour les appareils mobiles du fait de leur faible encombrement, ces claviers se trouvent moins efficaces pour la saisie de texte pas des personnes handicapées [8].

Mais cette optimisation de la disposition des touches, bien que meilleure en moyenne, peut être inappropriée pour un ensemble de mots, du fait de son statisme durant la saisie.

Des modèles dynamiques

Des systèmes plus dynamiques permettent d'améliorer la vitesse de saisie de texte. Le système Sybilette [6] permet de réorganiser l'ensemble des touches en fonction de ce qui vient d'être saisi. Mais ce clavier logiciel ne fonctionne pas avec un dispositif de pointage. Il est basé sur un défilement automatique linéaire des touches et l'utilisateur appuie sur un bouton poussoir lorsque le système est sur la bonne touche. Grâce à ce défilement automatique, l'utilisateur a le temps de percevoir les changements du clavier. Mais ce genre de système ne donnerait pas de bon résultats avec un système de pointage car la charge cognitive supplémentaire pour « apprendre » le nouveau clavier serait trop importante.

Dans notre cas d'utilisation (i.e. avec un système de pointage), le dynamisme des claviers est plus souvent basé sur l'ajout dynamique de menus au cours de la saisie. Ces menus peuvent être constitués soit par la liste des mots les plus probables comme dans [5], soit par une proposition de lettres par exemple sous forme de pie-menu [3].

Cependant la présentation de ces menus peut poser des problèmes à l'utilisateur. En effet, l'opacité du menu

cache une partie du clavier et son apparition peut gêner l'utilisateur dans sa saisie.

MODELES D'EVALUTION

Dès les années 50, Fitts [1] détermina une équation permettant d'estimer a priori le temps pour se déplacer d'une cible C_i à une autre C_j (cf. Fig. 1). Le temps de déplacement entre ces deux touches est de :

$$MT = a + b \log_2(D_{ij} / W_j + 1) \quad (1)$$

Où D_{ij} représente la distance entre les deux cibles, W_j la taille de la cible C_j et a et b sont des coefficients fixés de manière empirique.

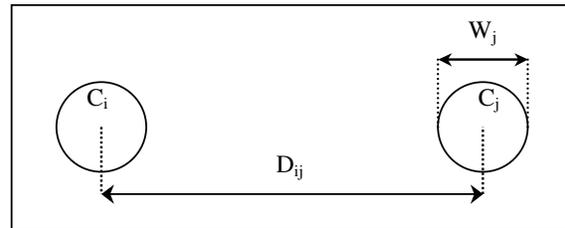


Fig. 2 : Distance D_{ij} entre une cible C_i et une cible C_j

Cette loi de Fitt a bien sûr été utilisée pour évaluer la vitesse de saisie de texte des claviers logiciels. Mackenzie, un des précurseurs dans le domaine a estimé les coefficients à $a=0.127$ et $b = 1 / 4.9$ [2].

Plus récemment, Zhai a proposer une formule permettant d'évaluer l'ensemble du clavier [9] en prenant en compte la distance entre chaque touche, pondérée par la probabilité qu'elles se succèdent. Cette formule permet ainsi de comparer aisément les différents claviers.

Cependant cette loi ne prend en compte que le temps mis pour aller d'une touche à l'autre. Ceci implique que l'utilisateur maîtrise le clavier qu'il utilise et qu'il ne passe pas du temps à chercher les caractères sur le clavier.

NOTRE SYSTEME : LES KEYGLASSES

Idée de départ

De la même façon que les pie-menus, nous souhaitons apporter près de la touche saisie, des touches supplémentaires qui pourraient accélérer la saisie. Mais à la différence de [3] qui propose toujours le même pie-menu, nos touches supplémentaires sont dépendantes de ce qui vient d'être saisi.

En outre, afin de ne pas trop encombrer l'écran et la charge de l'utilisateur, nous souhaitons limiter le nombre de touches supplémentaires. Dans un premier temps, seules les 4 caractères les plus probables sont affichés.

¹ <http://www.fitaly.com>

Le principe

Au fur et à mesure que l'utilisateur va saisir du texte, notre système lui propose autour de la touche qu'il vient de presser les quatre lettres qu'il a le plus de chance de saisir juste après celle-ci (cf fig. 3). Ces lettres sont déterminées grâce à un système de prédiction basé sur ce qui vient d'être saisi. Ce sont ces lettres rajoutées en fonction du contexte que nous appelons KeyGlasses du fait de leur semi-transparence.

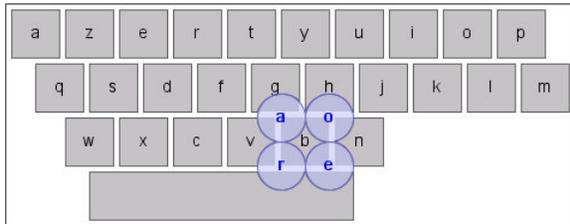


Fig. 3 : Disposition des touches après avoir saisi la lettre 'b'

La semi-transparence des KeyGlasses, permet de toujours voir l'intégralité du clavier. Leur couleur plus vive que le clavier permet d'attirer l'attention de l'utilisateur vers ces touches. Si une des KeyGlass lui convient il peut l'actionner de la même manière que les touches fixes : par un simple clic. Dans le cas où l'utilisateur souhaiterait accéder à la touche se trouvant sous une KeyGlass, il peut y parvenir par un double clic.

Ce système peut être récurrent : les KeyGlasses apparaissent après toute saisie de caractère, que ce soit à partir d'une touche fixe du clavier ou d'une KeyGlass (cf. fig. 4). Dans le cas d'une KeyGlass, cette dernière reste en arrière plan des 4 nouvelles KeyGlasses proposées.

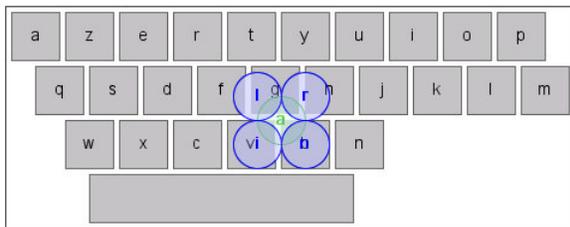


Fig. 4 : Cas de récurrence

Fonctionnement et architecture du système

Lorsqu'une touche est pressée, la position du clic est fournie au système qui détermine la position des KeyGlasses suivantes. En parallèle, la lettre saisie est envoyée au système de prédiction qui détermine les 4 lettres les plus probables et les renvoie au système d'affichage des KeyGlasses. Celui-ci attribue chacune d'elles à une KeyGlass et affiche le tout.

Ce système est indépendant de l'organisation des touches du clavier. La disposition des touches ainsi que les

caractéristiques de celles-ci sont décrites dans un fichier au format XML. Ainsi, il est possible de modifier la disposition des touches du clavier très simplement.

De plus, le système d'affichage des KeyGlasses est indépendant du système de prédiction. Il est donc possible de changer ce dernier pour, par exemple, adapter très rapidement ce système à une autre langue.

RESULTATS A PRIORI

A l'heure actuelle, nous n'avons pas mené d'expérimentations poussées avec des utilisateurs. Afin de tester rapidement l'efficacité de notre système, nous avons simulé la saisie d'un ensemble de mots sur celui-ci. Pour permettre la comparaison, nous avons réalisé la simulation sur un clavier logiciel de type AZERTY et ce même clavier avec notre système KeyGlasses. Pour ce dernier, nous avons simulé deux fois cette saisie en essayant deux systèmes de prédiction différents : un sous forme d'arbre lexicographique et l'autre sous forme de bigramme.

La simulation a été faite sur un peu plus de 30 000 mots différents pris dans des articles du Monde et de l'encyclopédie Universalis. Pour chacun de ces mots, nous avons calculé la distance qu'aurait normalement dû parcourir l'utilisateur via son dispositif de pointage pour passer d'une touche à l'autre. Cette distance a été calculée en supposant que l'utilisateur appuie systématiquement au centre de la touche. A chaque fois que la lettre à saisir faisait partie des KeyGlasses, c'est cette dernière qui a été sélectionnée.

Le tableau 1 présente les résultats des simulations réalisées sur les 3 systèmes. En plus de la distance moyenne (exprimée en pixels), nous apportons une estimation de la vitesse moyenne (en pixels/seconde) nécessaire pour saisir ces mots. Celle-ci a été calculée en utilisant la fonction de MacKenzie présentée précédemment.

	AZERTY	KeyGlass	
		arbre lexico	Bi-gramme
Distance	1184	453	589
Vitesse	3.22	2.14	2.34

Tab. 1 : Performance des différents systèmes

Le tableau 2 présente les gains (en pourcentage) pour la distance et la vitesse par rapport au clavier AZERTY. La dernière ligne donne le taux d'utilisation des KeyGlasses, si celle-ci ont été utilisées à chaque fois que cela est possible.

	Arbre Lexicographique	Bi-gramme
Distance	61.74	50.25
Vitesse	33.54	27.33
Utilisation	68,03	56.35

Tab. 2 : Performance des systèmes de prédiction

Les résultats que nous apportons dans le tableau 1, sont les résultats d'une utilisation optimale de ce système. Ils laissent envisager de bonnes performances. Cependant ils ne prennent pas en compte ni le fait que l'utilisateur ne prendra pas forcément la KeyGlass à chaque fois, ni le temps de recherche visuelle d'une touche.

Nous pouvons observer dans le tableau 2 l'importance du système de prédiction. En effet, même si les deux systèmes de prédiction donnent de bons résultats, ceux-ci varient de manière assez importante.

TRAVAUX FUTURS

Afin de confirmer ces premiers résultats et d'évaluer la surcharge cognitive engendrée par le système, nous allons maintenant tester notre système auprès d'utilisateurs finaux. Nous viserons plus particulièrement deux types de populations : d'une part les personnes handicapées moteur et d'autre part les utilisateurs de dispositifs mobiles.

En plus de ces expérimentations, il nous reste à valider certains points de notre système. Par exemple, il serait intéressant de déterminer le nombre optimal de KeyGlasses affichées. Les différents systèmes de prédiction seront aussi à tester pour voir quel est celui qui sera le plus performant.

Nous étendrons aussi nos tests sur différents types de claviers, pour déterminer si le gain s'atténue avec un clavier avec une disposition des touches optimisée.

CONCLUSION

Notre système qui peut être considéré comme un complément pour un clavier logiciel, apporte au vu des premiers résultats a priori un gain significatif sur la distance et donc sur la vitesse de saisie de texte.

BIBLIOGRAPHIE

1. Fitts P.M., The information capacity of the human motor system in controlling the amplitude of the movement, *Journal of experimental psychology* 47, pp.381-391, 1954.
2. Green N, Kruger J., Faldu C., St. Amant R., A reduced QWERTY keyboard for mobile text entry, *Proceedings of the ACM Conference on Human Factors in Computing Systems - CHI '04*, pp. 1429-1432, Vienne, 2004.
3. Isokoski P. Performance of menu-augmented soft keyboards, *Proceedings of the ACM Conference on Human Factors in Computing Systems - CHI '04*, pp. 423-430, Vienne, 2004.
4. MacKenzie, I. S., & Zhang, S. Z. The design and evaluation of a high performance soft keyboard. *Proceedings of the ACM Conference on Human Factors in Computing Systems - CHI '99*, pp. 25-31. New York: ACM, 1999.
5. Masui T., An efficient text input method for pen-based computers, *Proceedings of the ACM Conference on Human Factors in Computing Systems - CHI '98*, pp. 328-335, Los Angeles, 1998
6. Schadle I., Le Pevedic B., Antoine J.-Y., Poirier F., Sybillette : système de prédiction de lettre pour l'aide à la saisie de texte, *TALN'2001, atelier thématique "Handicap et Ingénierie Linguistique*, Tours, France, vol 2., pp. 233-242, 2001.
7. Soukoreff W., MacKenzie I.S., Theoretical upper and lower bounds on typing speeds using a stylus and keyboard, *Behavior & Information Technology*, 14, 370-379
8. Vigouroux N., Vella V., Truillet Ph., Raynal M., Evaluation of AAC for text input by two populations: normal versus handicapped motor persons (8th ERCIM UI4All, Vienna (Österreich), 28-29 June 2004
9. Zhai, S., M. Hunter, and B.A. Smith. The Metropolis Keyboard - an exploration of quantitative techniques for virtual keyboard design. *Proceedings of The 13th Annual ACM Symposium on User Interface Software and Technology (UIST)*. San Diego, California: ACM. pp. 119-218, 2000.

KeyGlasses : Des touches semi transparentes pour optimiser la saisie de texte

Mathieu Raynal
raynal@irit.fr

Problématique

- Augmenter la vitesse de saisie de texte sur un clavier logiciel
- Ne pas ajouter une trop forte charge cognitive pour l'utilisation du clavier logiciel afin de ne pas perturber l'utilisateur

Principe

- Proposer des touches supplémentaires après chaque lettre saisie, pour limiter les déplacements
- Lettres proposées grâce à un système de prédiction basé sur les lettres précédemment saisies
- Touches supplémentaires en semi-transparence : les KeyGlasses
- Possibilité de récurrence : après une pression sur une KeyGlass, d'autres apparaissent (cf. Fig. 2)

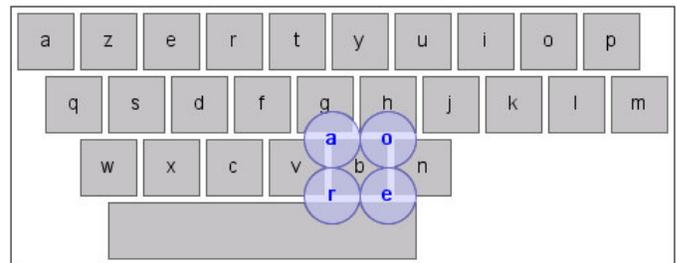


Fig. 1 : Disposition des touches après avoir saisi la lettre 'b'

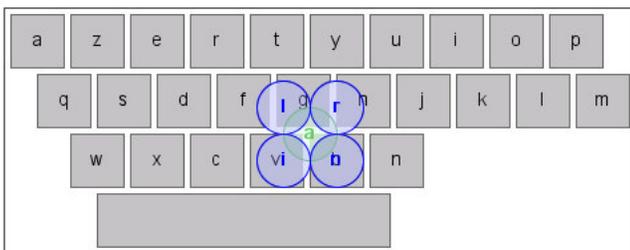


Fig. 2 : Cas de récurrence

	Arbre Lexico	Bi-gramme
Gain distance	61.74 %	50.25 %
Gain vitesse	33.54 %	27.33 %
Tx utilisation	68.03 %	56.35 %

Tab. 1 : Performance des systèmes de prédiction

Caractéristiques

- Indépendance entre le clavier, les KeyGlasses et le système de prédiction
- Disposition des touches et caractéristiques décrites dans un fichier XML

Premiers résultats

- Test en simulant une saisie « parfaite » sur plus de 30 000 mots
- Utilisation des KeyGlasses dans plus de 55 % des cas
- Gain assez important surtout pour la distance et amélioration de la vitesse de saisie
- Importance du choix du système de prédiction : écart important entre les 2 utilisés

Perspectives

- Concevoir une version utilisable sur des assistants personnels (Pocket PCs)
- Expérimenter sur deux populations de sujets : handicapés moteur et utilisateurs en situation de mobilité

Introduire des techniques de Programmation sur Exemple dans une boîte à outils : une étude de besoin

Loé SANOU, Patrick GIRARD, Laurent GUITTET

LISI/ENSMA

Téléport 2-1, avenue Clément Ader
BP 40109, 86961 FUTUROSCOPE Cedex
{loe.sanou, girard, guittet}@ensma.fr

RESUME

Les systèmes de Programmation sur Exemple (Programming by Demonstration ou PbD) ont à ce jour démontré leur intérêt dans de nombreux cas. Cependant, l'introduction systématique des techniques de PbD dans les applications n'est pas aisée en raison d'un manque d'outils. Nous avons entrepris de combler cette lacune en définissant des outils susceptibles d'être incorporés aux boîtes à outils classiques. Cette contribution constitue une première étape, par l'identification des classes d'applications de PbD, et par la caractérisation des besoins essentiels en terme d'enregistrement et de jeu.

MOTS CLES : Programmation sur Exemple, Boîtes à outils

CATEGORIES AND SUBJECT DESCRIPTORS: D.2.3 [Software Engineering]: Coding Tools and Techniques

GENERAL TERMS: Experimentation, Standardization

ABSTRACT

Programming by Demonstration (PbD) proved for a long time its interest in many cases. In the meantime, introducing such techniques into programs remains difficult, due to the lack of tools. We started to fill this gap by defining tools to be incorporated into classical toolboxes. This paper is a first step, with the classification of PbD applications, and the characterization of the essential needs for recording and replaying interactions.

KEYWORDS : Programming by Demonstration, Toolkits

INTRODUCTION

Les systèmes de Programmation sur Exemple (Programming by Demonstration ou PbD) sont assez nombreux dans la littérature, comme en témoignent les deux

recueils qui y sont consacrés ([1] et [2]). Dans certains cas particuliers, l'intérêt de cette technique a été largement démontré. Cependant, la PbD requiert des techniques de programmation approfondies (différences entre variables et constantes, structures de contrôles, nomination des objets, niveau d'enregistrement des actions, ...) et l'intégration des mécanismes de PbD nécessite le plus souvent une recomposition de l'application afin de pouvoir capturer les actions utilisateur à un niveau proche du noyau fonctionnel.

Jusqu'à ce jour, il n'existe pas de plate-forme robuste pour la construction de système de PbD et on ne dispose pas d'outils spécifiques permettant la réimplantation ou la réalisation de ces systèmes. La présente contribution se veut une première étape vers la réalisation d'une telle plate-forme, susceptible de mettre la technique de la PbD à la portée de tous les programmeurs. À partir d'une classification des systèmes de PbD, et en nous appuyant sur un exemple bien connu, nous définissons les principaux besoins d'une telle plate-forme en terme d'enregistrement et de jeu, qui constituent les bases de la technique.

CLASSIFICATION DES SYSTEMES DE PBD

Le concept de la programmation sur exemple trouve son origine dans Pygmalion ([1]). Il a été approfondi à travers Smallstar ([1]). En 1980, Brad Myers le formalise dans ses travaux et lui donne le nom de Example-based programming ([3]). En 1993, A. Cypher fait un récapitulatif des systèmes existants et introduit la notion de Programming by Demonstration ([1]).

Objectifs des systèmes de PbD

Selon les auteurs de ces systèmes, l'objectif principal est de générer interactivement des programmes. Cela est explicite dans les systèmes Peridot, Mondrian, SmallStar, ou encore ToonTalk ([1]). Aussi, les systèmes de PbD cherchent-ils à fournir aux non programmeurs le pouvoir d'expression d'un environnement de programmation classique (éditeur, structuration de programmes, ...). Le plus souvent, ils s'attachent également à enrichir et à personnaliser l'interface de dialogue du système interactif, en permettant à l'utilisateur d'associer l'exécution

d'un programme construit interactivement à une commande du système.

Par exemple, Eager [4] automatise les tâches répétitives entre applications différentes. À l'inverse Tinker ([1]) a pour but de favoriser l'apprentissage de la programmation elle-même. Les programmes produits n'ont pas d'intérêt particulier. Pygmalion ([1]) se focalise sur la création interactive de programmes et StageCast Creator ([2]) génère des applications. Enfin, le système EBP ([2]) crée des outils de conception technique.

Classes des systèmes

Beaucoup de systèmes de PbD n'ont eu pour but que de démontrer la validité du concept même de programmation sur exemple. Afin de caractériser les besoins en terme de PbD, nous avons préféré classer les systèmes en fonction de l'utilisation de la PbD elle-même. De ce point de vue, on peut répartir les systèmes en trois catégories : **l'assistance, les outils de conception et l'apprentissage de la programmation.**

L'assistance est la catégorie regroupant tout les systèmes permettant d'automatiser des tâches utilisateurs. C'est le but de la majorité des systèmes de PbD. Ils offrent une aide contextuelle à l'utilisateur afin de faciliter ou de réduire l'exécution de certaines de ses tâches. Le but global du système est très éloigné de la PbD. Dans cette classe on peut citer Eager (voir plus haut), Smallstar ([1]), qui cherche à résoudre les problèmes d'expressivité des langages iconiques en permettant de créer des macros puissantes, ou encore Tels ([1]), qui est spécialisé vers les tâches d'édition de texte.

Les outils de conception rassemblent les applications dont le but est de produire un résultat qui découle directement de l'utilisation de la PbD. Ces outils peuvent être classés dans la rubrique générale de la conception assistée par ordinateur. Qu'il s'agisse de pièces mécaniques comme pour EBP ([8]), d'applications à destination des enfants pour StageCast Creator ([2]) ou encore d'un GUI-Builder amélioré pour PERIDOT([5]), ils ont en commun de s'appuyer sur la PbD pour atteindre leur objectif.

L'apprentissage de la programmation concerne les systèmes dont le but est l'apprentissage de la programmation. On peut citer Tinker ([1]) ou encore Toontalk ([1]) et MELBA le plus récent avec Guibert paru dans IHM'04.

DES SOLUTIONS EXISTANTES

Fournir des services pour utiliser la PbD au sein des applications est fortement dépendant de la classe de système. Les applications dont le but est l'apprentissage de la programmation ont des besoins extrêmement spécifi-

ques ; la programmation sur exemple est le cœur même du système interactif. De même, les outils de conception demandent en règle générale une très forte imbrication du système de PbD dans le noyau fonctionnel même du système global. En revanche, les applications de la première catégorie n'utilisent la PbD que pour ajouter des fonctionnalités ou aider l'utilisateur dans sa tâche. Cependant, rares sont les travaux qui ont essayé d'apporter une solution générale permettant de développer à moindre coût des solutions incluant la PbD.

Deux solutions ont été développées selon des approches radicalement différentes : une approche interne, au travers du système AIDE ([6]), et une approche externe, avec PbDScript ([7]). Ces systèmes ne peuvent être classés dans les trois catégories présentées ci-dessus, leur objectif étant de construire des applications de PbD. Ils constituent une quatrième catégorie, que l'on peut qualifier d'**outils de PbD.**

L'utilisation de la voie interne consiste à intégrer la PbD au cœur de l'application, pendant sa construction. À l'inverse, l'approche externe consiste à espionner une application déjà développée, pour permettre l'application de la PbD. Elle peut faire le choix de l'établissement d'hypothèses préalables sur les noyaux fonctionnels (ce qui réduit le champs des applications) ou ne considérer que des interactions de bas niveau (de manière à étendre le choix des applications).

AIDE vise à faciliter la création de systèmes en fournissant une couche logicielle sur laquelle les développeurs peuvent bâtir des applications mettant en œuvre des techniques démonstrationnelles. L'utilisateur définit des macros opérant sur une collection ordonnée d'arguments. Afin d'enseigner une macro, l'utilisateur sélectionne les objets formant ses arguments et démontre les étapes de la macro en effectuant les commandes appropriées. Les objets sélectionnés à la fin de la macro constituent la valeur retournée par la macro. L'architecture d'AIDE s'articule autour des commandes de haut niveau et d'un gestionnaire de commandes gérant l'historique et la création de macros. Le processus de généralisation s'effectue par la détection de boucles non imbriquées, la généralisation des arguments et la détection de séquences (numérique, chaîne de caractères, points et classes). AIDE présente aux développeurs une structuration d'application mettant en œuvre la PbD mais n'est pas un outil permettant d'ajouter facilement des capacités de PbD à une application. Il suppose de respecter une architecture particulière, et impose le langage Smalltalk pour le développement.

PbDScript est un système qui permet d'enregistrer de façon totalement externe les interactions d'une application. PbDScript récupère l'arbre des widgets de présen-

tation d'une application Java par un mécanisme d'introspection, puis l'affiche. Il demande à l'utilisateur d'ajouter de l'information aux widgets et aux événements provenant de ces widgets ; la sémantique de l'application est ainsi décrite par l'utilisateur. PbDScript est un système très souple, qui permet de créer très simplement des macros sur une application quelconque. Cependant, ces fonctionnalités de PbD ne peuvent être incluses directement dans le programme. D'autre part, la tâche d'apprentissage de l'application est relativement longue car elle nécessite d'activer tous les écrans de l'application afin que l'introspection puisse se faire correctement.

Bien que présentant des intérêts certains, ces deux systèmes ne répondent pas au besoin que nous avons exprimé : PbDScript ne permet pas à un concepteur d'introduire des fonctionnalités de PbD dans l'application qu'il réalise. Quant à AIDE, il est limité de par ses choix d'implémentation (Smalltalk) et par sa technique d'espionnage, point sur lequel nous reviendrons plus loin.

L'EXEMPLE EAGER

Afin de rendre plus concrète l'expression des besoins en terme de PbD, nous avons choisi de nous appuyer sur une application typique de la catégorie « assistance », Eager.

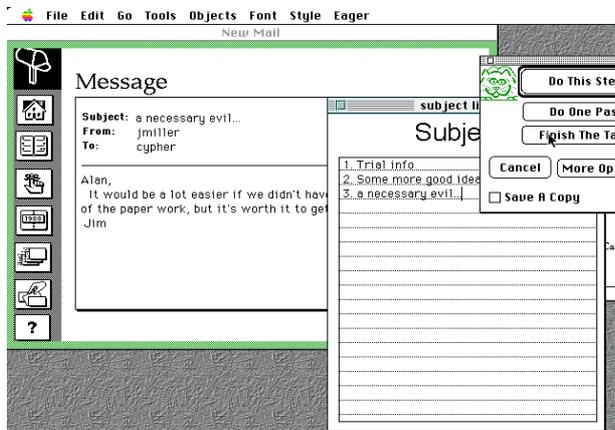


Figure 1 : Eager

Eager permet d'aider l'utilisateur dans ses tâches répétitives. Il prend en compte le fait que l'utilisateur ne réalise pas toujours immédiatement qu'il peut créer un programme pour accomplir sa tâche. Le système enregistre en permanence les actions de l'utilisateur, à la recherche de potentielles répétitions. Lorsqu'il détecte l'une d'entre elle, il demande à l'utilisateur de valider sa « déduction » en anticipant la commande suivante de l'utilisateur sans l'exécuter. Si l'inférence est incorrecte, la commande est considérée comme un contre exemple. L'utilisateur confirme la bonne inférence par clic sur une icône particulière de menu afin que Eager finalise la tâche.

che. Il a le choix entre l'exécution d'un seul coup (un seuil de confiance égal au moins à trois itérations), ou de faire du pas à pas. Eager n'infère pas de conditions, ni de boucles imbriquées, ni de répétitions temporelles.

Notre objectif consiste à identifier les besoins directement liés à la PbD lors de la conception d'une application comme Eager. Eager permet d'automatiser des tâches effectuées dans des applications classiques. L'exemple d'Eager s'appuie sur deux applications simples que sont un mini-éditeur de texte et un mini-mailer. La tâche automatisable consiste à recopier les en-têtes de mails à partir du mailer dans l'éditeur de texte, en les numérotant.

D'un point de vue utilisateur, la tâche consiste à sélectionner le titre du mail, le copier, activer l'éditeur, numéroté la ligne, puis coller le contenu du presse-papiers. Le retour dans le mailer et l'activation du mail suivant constituent la fin de la séquence qui doit être répétée pour tous les mails.

BESOINS POUR UNE BOITE A OUTILS

Du point de vue du programmeur, il s'agit de recenser et de minimiser les actions de programmation explicite nécessaire pour construire ces fonctionnalités de PbD. Les boîtes à outils possèdent aujourd'hui des widgets évolués qui permettent de s'affranchir de beaucoup de programmation. Ainsi, le « Copier-Coller » est-il fonctionnel sans effort particulier dans des widgets de type texte. Les applications de type WIMP, qui représentent la grande majorité des applications actuelles, sont organisées à base de conteneurs rassemblant des widgets actifs, comme des boutons ou des champs de texte. Dans le cas de notre exemple, l'éditeur de texte possède les fonctions de couper-coller par l'intermédiaire de menus, et le mailer possède en plus des boutons « Suivant » et « Précédent » pour naviguer dans les mails.

Quels services fournir ?

Une application de PbD doit permettre l'enregistrement et le rejeu des interactions de l'utilisateur, auxquelles s'ajoutent des capacités de généralisation. Plusieurs travaux ont défini les besoins de couplage entre les applications hôtes et les systèmes de PbD, en se focalisant particulièrement sur la généralisation. On trouvera ainsi une étude comparée dans [8]. Mais dans l'optique de fournir des outils généraux et indépendants des applications, il faut s'intéresser beaucoup plus finement aux opérations de base que constituent l'enregistrement et le rejeu

Enregistrement

Dans AIDE, l'enregistrement s'appuie sur la définition a priori de commandes. Dans notre exemple, il s'agirait de commandes telles que « Copier », « Coller » ou « Mail Suivant ». Dans le dernier cas, il s'agit d'espionner un

événement du type du clic sur un bouton, qui doit être explicitement programmé par le concepteur de l'application. Mais dans les deux autres cas, on doit tenir compte de deux possibilités d'interaction : l'utilisateur peut utiliser les menus, ou bien les raccourcis clavier qui sont généralement implantés par défaut dans les widgets, sans intervention explicite du programmeur.

Se restreindre aux actions de type « commande » n'est cependant pas suffisante pour permettre d'espionner en totalité l'utilisateur. Dans le cas d'Eager, l'utilisateur effectue des actions de frappe au clavier à l'intérieur de l'éditeur de texte. Lorsque l'on utilise des widgets évolués, ce type d'interaction n'a pas besoin d'être programmé. Le widget le gère tout seul. On voit ici la limite du système AIDE qui impose la définition de commandes, qui seules peuvent être enregistrées. À l'inverse, n'espionner que les opérations de très bas niveau ne permet pas de s'abstraire des tâches articulatoires qui constituent une part importante du dialogue homme-machine.

Rejeu

Au niveau du rejeu, Eager est un bon exemple des différents besoins qui se font jour. Lorsque Eager souhaite faire valider son analyse de la tâche exemple, il doit réactiver les interactions de l'utilisateur en les mettant en évidence. Par exemple, lorsque l'utilisateur est invité à confirmer une commande activée par un item de menu, Eager modifie la présentation du menu pour ajouter à l'item choisi une icône spécifique (Figure 2)

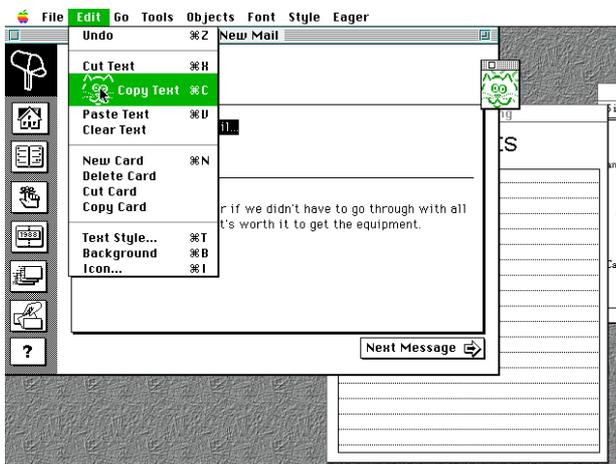


Figure 2 : Rendu spécifique à Eager

À l'inverse, lorsque la tâche automatisée doit être complétée, il n'est plus forcément nécessaire de rejouer très exactement les interactions de l'utilisateur. On constate donc que la boîte à outil doit permettre ces deux modes de fonctionnement selon le besoin de rejeu. Là encore, AIDE se révèle insuffisant dans son approche, car le rejeu des commandes est loin de satisfaire tous les besoins d'une application comme Eager.

Principes de réalisation

À partir de l'étude que nous avons réalisée, nous développons actuellement un prototype de boîte à outils sur la base de Swing. L'utilisation de l'abonnement, qui a servi de base à la réalisation de PbDScript, est faite pour satisfaire le besoin d'espionnage des commandes explicites. Dans une première approche, nous n'avons pas retenu la solution des hiérarchies d'action préconisée par AIDE, car elle impose une architecture d'application particulière, qui nous semble inappropriée à ce niveau.

Pour gérer les actions implicites, nous pensons qu'il est nécessaire de sous-classer les widgets de la boîte à outils support. Ceci permet d'apporter toute la puissance nécessaire tout en limitant le travail du programmeur d'application. Ce sous-classement permet également de contrôler finement le rendu des widgets pour ajouter les primitives nécessaires au rejeu.

CONCLUSION

La présente étude constitue un premier pas vers la réalisation d'une véritable boîte à outils susceptible de simplifier la mise en œuvre de la programmation sur exemple dans les applications interactives. La prochaine étape consiste à définir les solutions d'implémentation et à les expérimenter.

BIBLIOGRAPHIE

1. Cypher, A., ed. *Watch What I Do: Programming by Demonstration*. 1993, The MIT Press: Cambridge, Massachusetts. 604.
2. Lieberman, H., *Your Wish is my command*. 2001: Morgan Kaufmann. 416.
3. Myers, B.A., *Taxonomies of Visual Programming and Program Visualization*. *Journal of Visual Languages and Computing*, 1990. **1**(1): p. 97-123.
4. Cypher, A. *Eager: Programming Repetitive Tasks by Example*. in *Human Factors in Computing Systems (CHI'91)*. 1991. New Orleans, Louisiana: ACM/SIGCHI.
5. Myers, B.A., *Creating User Interfaces Using Programming by Example, Visual Programming, and Constraints*. *ACM Transactions on Programming Languages and Systems*, 1990. **12**(2): p. 143-177.
6. Piernot, P.P. and M.P. Yvon, *The AIDE Project: An Application-Independent Demonstration Environment*, in *Watch What I Do: Programming by Demonstration*, A. Cypher, Editor. 1993, The MIT Press: Cambridge, Massachusetts. p. 383-402.
7. Depaulis, F., L. Guittet, and C. Martin. *Apprends ce que je fais*. in *15ème Conférence Francophone sur l'Interaction Homme-Machine*. 2003. Caen: ACM Press.
8. Girard, P., *Ingénierie des systèmes interactifs : vers des méthodes formelles intégrant l'utilisateur*, in *LI-SI/ENSMA*. 2000, Université de Poitiers: Poitiers. p. 92.

Introduire des techniques de Programmation sur Exemple dans une boîte à outils : une étude de besoin

État de l'art

La Programmation sur Exemple (PbD) requiert des techniques de programmation approfondies (différences entre variables et constantes, structures de contrôles, nomination des objets, niveau d'enregistrement des actions, ...). L'intégration des mécanismes de PbD nécessite le plus souvent une recombinaison de l'application ciblée afin de pouvoir capturer les actions utilisateur à un niveau proche du noyau fonctionnel.

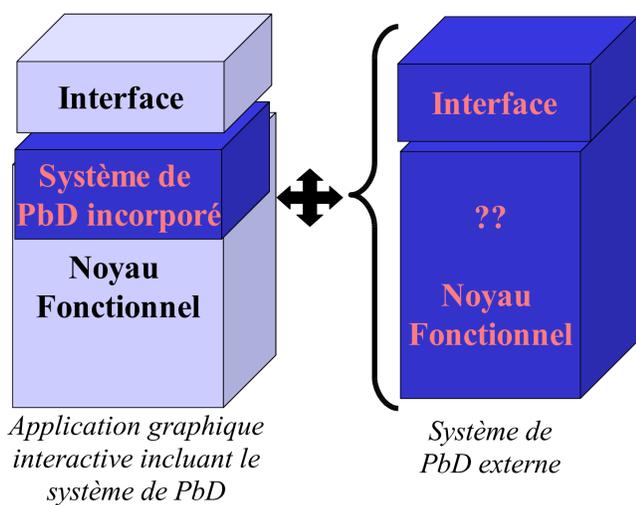
Il n'existe pas de plate-forme robuste (système de création ou de gestion de programmes, architecture déployée) pour la construction de système de PbD. On ne dispose pas d'outils spécifiques permettant la réalisation ou la réimplantation de ces systèmes.

Notre contribution

La présente contribution est une première étape vers la réalisation d'une telle plate-forme, susceptible de mettre la technique de la PbD à la portée de tous programmeurs.

- ✓ Classification des systèmes de PbD,
- ✓ Définir les principaux besoins,
- ✓ Appui sur l'exemple du système Eager[Cypher 91] pour fournir les besoins d'une boîte à outils.

Vision architecturale



Enregistrement

S'appuie sur la définition à priori de commandes (en tant que design pattern) tout en s'abstrayant des tâches articulatoires constituant une part importante du dialogue homme-machine.

Deux types d'actions à espionner : commandes explicites programmées par le programmeur (exemple : clic sur un bouton) et commandes implicites généralement implantées par défaut dans les widgets (exemple : utilisation de menu ou de raccourcis clavier).

Principe de réalisation à base de Swing :

Nous développons actuellement un prototype de boîte à outils sur la base de Swing.

L'utilisation de l'abonnement est faite pour satisfaire le besoin d'espionnage des commandes explicites.

Sous-classer les widgets de la BO support serait nécessaire pour gérer les actions implicites. Contrôler finement le rendu des widgets pour ajouter les primitives nécessaires.

Deux types de solutions existent actuellement

développés selon des approches radicalement différentes :

- une **approche interne** (intégrer la PbD au cœur de l'application pendant sa construction) : AIDE Project

- une **approche externe** (espionner une application déjà développée) faisant le choix de l'établissement d'hypothèses préalables sur les noyaux fonctionnels ou ne considérer que des interactions de bas niveau : PbDScript.

Auteurs :

- Loé SANOU (LISI/ENSMA)
loe.sanou@ensma.fr
- Patrick GIRARD (LISI/ENSMA)
- Laurent GUITTET (LISI/ENSMA)

Trois classes de systèmes (en fonction de l'utilisation de la PbD)

- **L'assistance** : automatiser des tâches utilisateurs.
- **Les outils de conception** : produire un résultat qui découle directement de l'utilisation de la PbD.
- **L'apprentissage de la programmation** : apprendre à programmer.

2 systèmes non classifiés car leur objectif est de construire des applications de PbD : The AIDE Project [Pernot 1993] et PbDScript [Christophe 2003]

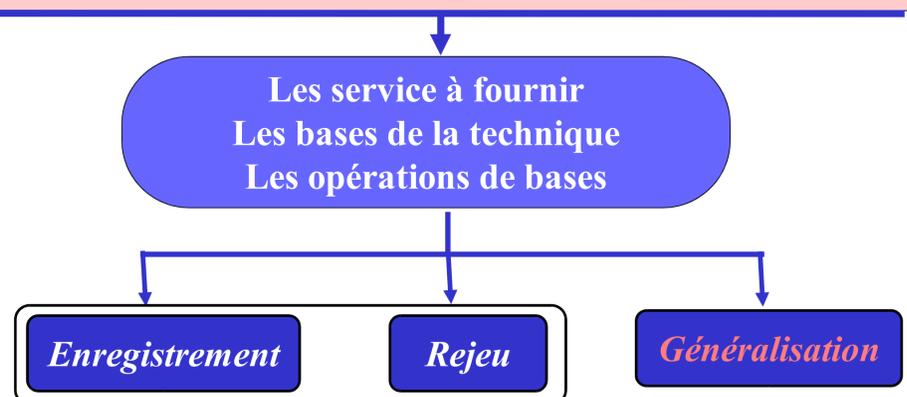
Ces 2 systèmes peuvent constituer une 4^{ème} catégorie : **les outils de PbD**

BESOINS POUR UNE BOITES A OUTILS (BO)

Du point de vue du programmeur : recenser et minimiser les actions de programmation explicite nécessaire pour construire ses fonctionnalités de PbD.

Une application de PbD devra permettre l'enregistrement et le rejeu des interactions de l'utilisateur, auxquelles s'ajoutent des capacités de généralisation.

Opérations de base : l'enregistrement et le rejeu.



Rejeu

La BO doit permettre deux modes de fonctionnement : ré-activation des interactions de l'utilisateur en les mettant en évidence et aussi la non nécessité de rejouer très exactement ces interactions lorsque la tâche automatisée doit être complétée.

Web :

- <http://www.lisi.ensma.fr/members/sanou/>
- <http://www.lisi.ensma.fr/ihm/>

Etude de l'optimisation de claviers virtuels au travers des sujets handicapés versus valides

Frédéric Vella, Nadine Vigouroux

IRIT (Institut de Recherche en Informatique de Toulouse, Université Paul Sabatier)
Equipe Dialogue InterAction Multimodalité Accessibilité et Nouvelles Technologies (DIAMANT)
118, Route de Narbonne 31062 Toulouse Cedex
{[vella.vigourou](mailto:vella.vigourou@irit.fr)}@irit.fr

RESUME

L'objectif de ce papier est de présenter les problèmes de saisie de texte concernant les sujets handicapés. Deux expérimentations ont permis de montrer que les modèles neurophysiologiques conçus pour la saisie de texte ne sont pas adaptés pour les sujets handicapés. A partir de ces résultats nous avons été obligé de concevoir un environnement de conception de claviers virtuels qui évalue de manière théorique leurs performances.

MOTS CLES : Modèles neurophysiologiques, handicapés moteurs, clavier virtuel.

ABSTRACT

The aim of this paper is to present the problems of text input concerning the disabled subjects. Two experiences have showed that the neurophysiological models designed for the text input are not adapted for the disabled subjects. From these results we were obliged to elaborate an environment of design for virtual keyboards which evaluates in a theoretical way their performances.

CATEGORIES AND SUBJECT DESCRIPTORS: H.4 [Information Systems Applications]: Miscellaneous; D.2.8 [Software Engineering]: Metrics/complexity measures.

KEYWORDS : Neurophysiological model, diseabled motor, virtual keyboard.

INTRODUCTION

La communication écrite est une activité cognitive essentielle pour l'échange d'informations numérique (rédactions de documents textuels, accès aux serveurs d'informations, communication inter-personnes via des boîtes de communication (fenêtre chat, etc.) ou encore l'écriture de SMS¹ avec l'arrivée des dispositifs portables, etc. Depuis les années 1980, de nombreuses recherches sur les aides technologiques d'assistance à la saisie de texte ont été conduites dans divers contextes pluridisciplinaires : en perception sur l'étude du mouvement [8], dans une tâche de saisie manuscrite instrumentée ou au travers de claviers logiciels et plus récemment en interaction homme-machine sur le processus d'interaction [1], optimisation de la configuration spatiale des touches [8], etc.

Les claviers virtuels, WiVik² et QualiKey³ sont utilisés par

des personnes handicapées moteurs des membres supérieurs. Durant la saisie, ces personnes effectuent des parcours d'une touche à une autre. Du fait que les distances parcourues soient longues et répétées dans le temps, elles éprouvent fréquemment de la fatigue [7]. Les conséquences visibles sont l'apparition de fautes d'orthographe et une saisie de texte plus longue. Afin d'éviter ces deux conséquences, nous devons rechercher une configuration qui optimise : 1) le déplacement du pointeur de la souris et 2) le temps de recherche d'une touche. Pour réaliser cette optimisation nous avons besoins de modèles évaluant les performances humaines pour une tâche de saisie de texte. D'ailleurs certains ont été élaborés pour étudier les temps de réaction ou de production de trajectoires de la souris ou au clavier : on peut mentionner, les lois de Fitts [3], de Hick-Hyman [4, 5]. Ces lois ont permis la modélisation des performances d'un utilisateur pour la réalisation d'une tâche comme avec le modèle KLM [1] (temps d'une tâche d'exécution en tenant compte de la conception et spécification de cette dernière) ou encore dans le cadre de la saisie de texte [8]. Cependant, l'expérience effectuée par Edmund [2] sur l'utilisation des commandes de l'ordinateur à l'aide des mouvements de la tête a prouvé que les modèles neurophysiologiques ne sont pas adaptés aux personnes handicapées. Les estimations théoriques de ces taux de performance du mouvement semblent être éloignées de la réalité pour les personnes handicapées. Ces problématiques nous ont conduit à deux grandes tâches qui sont : 1) la modélisation de la tâche de saisie de texte pour des personnes handicapées et 2) l'élaboration d'une plateforme de conception de claviers virtuels.

Dans ce papier nous décrivons un premier travail. Tout d'abord, un bref récapitulatif des lois et des modèles sera réalisé. Puis nous décrivons deux évaluations sur deux populations (personnes handicapées moteurs (type Myopathe) et personnes valides (en tant que population de référence)). Ces expérimentations se sont appuyé sur les trois actions permettant la saisie au clavier virtuel (visualiser, déplacer et sélectionner). Ces dernières ont permis d'observer les comportements différents entre personnes valides et handicapées. Enfin, ces résultats et les besoins des utilisateurs nous ont conduit à la conception de l'environnement SOKEYTO dont nous décrivons les principes.

PRESENTATION DES LOIS ET DES MODELES

Depuis les années 50, bon nombre de recherches ont eu lieu sur les différentes méthodes modélisant les processus humains. La première modélisation fut abordée par Hick [4] et Hyman [5], elle concerne le temps de prédiction de sélection d'un item dans un ensemble de choix possibles (caractère ou mots par exemple). La seconde modélisation fut abordée par Fitts [3], elle concerne le temps pour se déplacer d'une cible à une autre. A partir de ces deux lois des

Réserver cet espace pour la notice de copyright

¹

² <http://www.wivik.com/>

³ <http://www.qualilife.com>

méthodes semi-intensives⁴ (comme par exemple Keystrokes-Level-Model (KLM) [1]) ont été élaborés.

Le modèle KLM [1] est un modèle donnant la prédiction de la durée d'une tâche à exécuter selon un scénario donné. L'approche de KLM [1] consiste à spécifier d'abord un scénario définissant une action à effectuer. Ensuite, dans ce scénario il faut décomposer cette action en plusieurs sous activités qui va ensuite donner lieu à un modèle de tâche. Ce modèle va être obtenu grâce à des opérateurs de temps qui sont les suivants :

$$T = T_K + T_P + T_H + T_D + T_M + T_R \quad (1)$$

Où :

T: temps total nécessaire à l'exécution d'une tâche; T_K : temps nécessaire pour effectuer un click; T_P : temps de pointage; T_H : temps nécessaire pour prendre un dispositif; T_D : temps de déplacement de curseur d'un point à un autre; T_M : temps de l'activité mentale de l'utilisateur; T_R : temps de réponse du système.

Dans le scénario de tâche – saisie d'un caractère – l'estimation est réalisée avec les valeurs empiriques de KLM [1]:

- Prendre en main du dispositif de pointage (H)
- Visualiser une lettre (M)
- Déplacer un dispositif de pointage (P)
- Sélectionner le caractère (BB)

Le modèle prédictif KLM pour les trois types d'utilisateur :

$$T_{\text{débutant}} = H+M+P+BB = 0.4+0.0135+1.5+0.2 = 2.113 \text{ s}$$

$$T_{\text{moyen}} = H+M+P+BB = 0.4+0.012+1.1+0.2 = 1.712 \text{ s}$$

$$T_{\text{expert}} = H+M+P+BB = 0.4+0.006+0.8+0.2 = 1.406 \text{ s}$$

Ces estimations de KLM ont été élaborées pour des personnes valides. Néanmoins, les travaux effectués par Edmund [2] et Keates [6] a prouvé que les modèles neurophysiologiques ne sont pas adaptés aux personnes handicapées. On peut s'interroger sur l'adaptation des valeurs estimées de KLM dans le cas de la saisie de texte pour les personnes handicapées. C'est ce que nous allons montrer maintenant avec les deux études de cas que nous avons réalisés.

LES ETUDES DE CAS

Nous avons effectué un premier travail dont les objectifs est d'une part de montrer que les lois et les modèles neurophysiologiques ne sont pas adaptés aux personnes handicapées dans une tâche de saisie de texte et d'autre part que les comportements moteurs des personnes valides et des personnes handicapées sont différents.

Première expérience

Protocole

L'expérience décrite dans [9] consiste à comparer deux types de populations dans une tâche de saisie de texte au moyen de deux types de claviers virtuels (Figure 1) : ambigu (clavier type téléphone) et non ambigu (clavier type AZERTY). La saisie à l'aide de ces claviers s'effectue selon trois actions décrites auparavant. Cependant, ces trois actions ne sont pas réalisées de la même manière selon le type de clavier. Pour le clavier non ambigu, l'utilisateur visualise la touche, déplace le pointeur vers cette dernière et la sélectionne. Par contre, pour le clavier ambigu, l'utilisateur visualise la touche contenant le caractère voulu puis déplace le pointeur vers cette dernière et réalise sur le dispositif plusieurs clicks pour sélectionner le bon caractère. La raison pour laquelle nous avons choisi ces deux types de claviers est d'observer lesquels des deux est le plus adapté aux personnes handicapées.



Figure 1 : Clavier type AZERTY et type Téléphone

Pour vérifier cela nous avons émis deux hypothèses :

Hypothèse 1 : La distance parcourue à l'aide du pointeur est plus longue sur le clavier AZERTY que sur le clavier téléphone.

Hypothèse 2 : Le temps de saisie est plus long sur le clavier AZERTY (nommé C1) que sur le clavier téléphone sans système de prédiction (nommé C2) et par conséquent le temps de saisie de ce dernier est plus long que le clavier téléphone avec prédiction (nommé C3).

Deux types de populations ont participé à cette expérience : 2 sujets handicapés et 5 sujets valides mais tous sont familiers avec le clavier type AZERTY. Chacune des deux populations a pour consigne de saisir le plus rapidement possible durant 9 sessions un texte s'affichant sur un bandeau. Ce texte est constitué de 1022 caractères qui est lui même divisé en 9 parties c'est à dire, une partie pour chaque session. Une session correspond à un jour.

Résultats

A partir de cette expérimentation nous avons obtenu deux résultats.

Le premier résultat (Figure 2) montre que la première hypothèse est vérifiée par les deux populations. En effet, les distances couvertes pendant les saisies de textes sont clairement inférieures pour les claviers C2 que pour C1. Ceci est dû au nombre différent de touches des deux claviers. Les sujets parcourent plus de distance avec le clavier C1 (27 touches) qu'avec les claviers C2 et C3 (10 touches, deux fois plus pour C1 que C2).

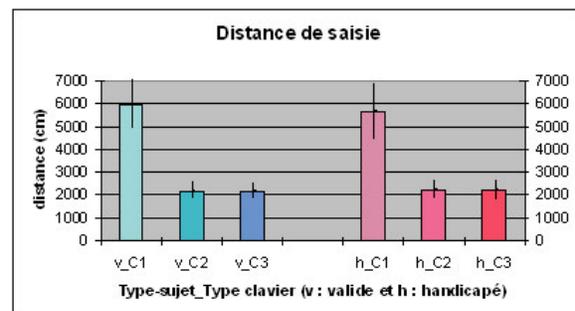


Figure 2 : Distance de saisie en cm

Le second résultat (Figure 3) montre que l'hypothèse 2 n'est pas vérifiée. En effet, pour les deux types de populations, le clavier C1 reste le plus performant concernant la vitesse de saisie des textes en comparaison avec les claviers C2 et C3. Toutefois, les performances de C3 sont meilleures que C2. Ceci pourra s'expliquer par le fait que le clavier C1 est utilisé quotidiennement par les sujets. Par conséquent, la configuration spatiale de C1 est connue. Cependant, nous observons une augmentation croissante des performances des sujets valides comme dans [8]. Il y aurait donc un effet d'apprentissage qui se produit progressivement pendant les neuf sessions pour C2 et C3.

Cependant, nous pouvons observer chez les sujets handicapés que les performances de ces deux claviers C2 et C3 rejoignent celui de AZERTY.

L'explication évoquée pourrait être que les clics successifs effectués sur les claviers ambigus sont plus difficiles à ré-

⁴ évaluation avec tâche réelle et utilisateur simulé

aliser que les mouvements du pointeur, pour les sujets handicapés.

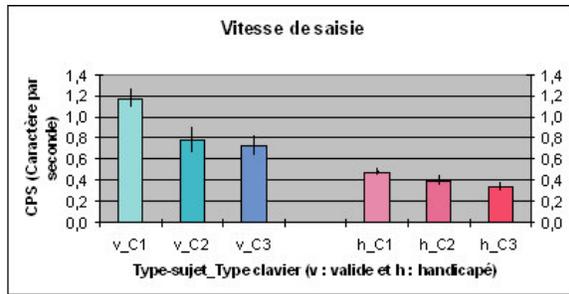


Figure 3 : Vitesse de saisie en CPS (Caractère / Seconde)

Etant donné que les deux populations ont une meilleure performance avec le clavier C1, nous avons calculé la valeur théorique de la saisie du texte de 1022 caractères avec le modèle KLM [1]. Lors de la comparaison avec la valeur théorique (Tableau 1), nous remarquons que l'écart avec cette dernière est plus important avec le résultat expérimental des sujets handicapés (- 0,513) qu'avec celui des sujets valides (+ 0,167). Les résultats rejoignent ceux de Keates [6] et Edmund [2]. Ceci confirme que les valeurs empiriques de KLM ne sont pas adaptées aux personnes handicapées.

	Expert
CPS correspondent KLM	0,993
CPS pour sujets valides	1,16
CPS pour sujets handicapés moteurs	0,48

Tableau 1 : Vitesse de saisie théorique en CPS comparée avec la vitesse de saisie expérimentale

Deuxième expérience

Protocole

Comme nous avons vu dans l'expérience précédente nous avons évoqué le problème des clics successifs, c'est la raison pour laquelle nous avons réalisé une deuxième expérience. Elle consiste à comparer le temps de réponse d'une série de clics entre les deux populations d'étude (handicapées et valides).

Nous avons formulé les hypothèses suivantes :

Hypothèse 1 : La vitesse moyenne des clics est plus importante chez les sujets valides que chez les sujets handicapés moteurs.

Hypothèse 2 : Les symptômes de fatigue sont plus ressentis par les personnes handicapées que par les sujets normaux.

Dans le cadre d'une pré-expérience, ces hypothèses ont été évaluées sur 4 personnes valides et 2 personnes handicapées. Les sujets sont familiers des ordinateurs, nous n'avons donc pas procédé à une phase de découverte du test.

Le sujet effectue durant 13 minutes une série de clics. La vitesse est obtenue par la mesure des clics successifs réalisée avec un dispositif de pointage (souris ou trackball). Cette expérience consiste à cliquer à l'aide du dispositif de test sur le rectangle de 3*10,5 centimètre qui apparaît au milieu de l'écran de l'ordinateur.

Résultats

La Figure 2 montre que les utilisateurs valides sont réguliers (pour v1,v2,v3 et v4). Ces résultats sont identiques à ceux de Keates [6]. Les résultats obtenus sur les sujets handicapés (h1 et h2) moteurs montrent qu'il est nécessaire d'avoir plusieurs profils d'utilisateur handicapé. Les résultats montrent la difficulté des personnes handicapées motri-

ces à produire des mouvements [7]. L'hypothèse 1 est confirmée : les personnes handicapées moteur sont approximativement (1,48 fois) plus lente que les personnes valides. L'hypothèse 2 est elle aussi confirmée, nous pouvons observer que que h1 et h2 "oscille" plus que v1,v2,v3 et v4, cela explique que les sujets handicapés éprouvent plus de fatigue que les sujets valides. Ces résultats préliminaires montrent qu'il est nécessaire de poursuivre les études afin d'obtenir des explications sur le comportement différent dans une tâche de saisie entre personnes valides et personnes handicapées. Une explication est suggérée par [6] qui a montré qu'une personne handicapée moteur doit faire un effort cognitif supplémentaire (comme se concentrer) pour réaliser des mouvements physiques.

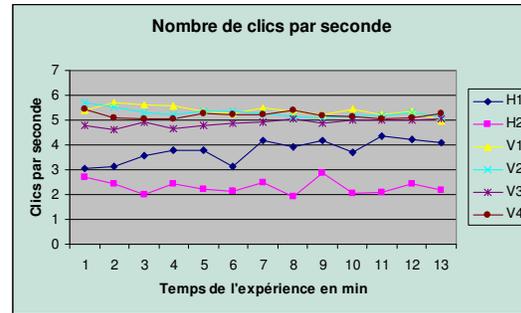


Figure 4 : Vitesse du nombre de clics

Les premiers résultats de ces deux expériences soulèvent la question de l'adéquation des modèles prédictifs de performance aux sujets handicapés. Ils en résultent plusieurs axes d'études. Faut-il: a) augmenter le nombre de sujets pour voir si les résultats préliminaires sont cohérents; b) élaborer des modèles de prévision prédictive qui représenteraient mieux les comportements moteurs des sujets en situation d'écriture; c) déterminer les facteurs de la fatigabilité ou l'effet de l'interaction.

Dans cet objectif d'étude, nous avons développé une première version de la plate-forme SOKEYTO de conception de claviers virtuels (agencement spatial des touches). Ainsi pour chacun des claviers virtuels, des taux de prédiction sont possibles selon les modèles de Fitts, Hick-Hyman et de KLM.

PLATE-FORME SOKEYTO (SOFT KEYBOARD TOOLKIT)

L'environnement SOKEYTO (Figure 5) permet de concevoir des claviers virtuels avec une configuration spatiale des touches différentes, des langages de saisie différents (alphanumérique, alphabétique phonétique international, etc.) ainsi que des fonctionnalités différentes (système de prédiction linguistique, clavier multi niveau). La représentation du clavier est décrite dans un langage au format XML (eXtended Markup Language).

Ainsi, nous pouvons disposer d'une base de ressources de claviers virtuels à des fins d'usages et d'études de performances suivant leur description.. Contrairement aux environnements WiVik et QualiKey, SOKEYTO intègre (comme dans [10] concernant d'autres types d'interfaces) des modèles et des lois neurophysiologiques : loi de Fitts [2], loi de Hick-Hyman [3] et le modèle KLM (Keystroke-level-Model) [1]. Cela permet de concevoir et de valider (de manière théorique) rapidement les performances de nouveaux claviers virtuels.

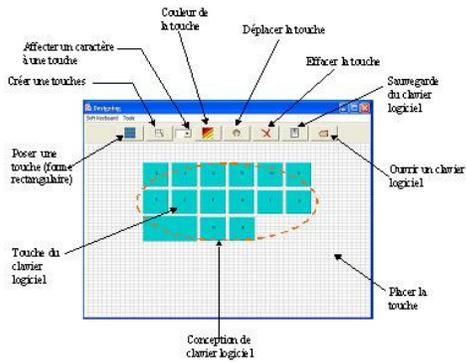


Figure 5 : Interface de conception de SOKEYTO

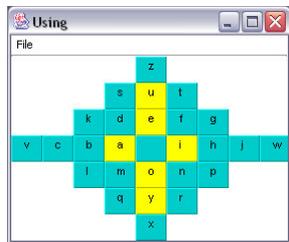


Figure 6 : Clavier "Annie" avec différenciation des voyelles et consonnes

Le Tableau 2 illustre les résultats théoriques obtenus grâce aux modèles neurophysiologiques intégré dans SOKEYTO qui permet d'estimer le temps de saisie théorique, (respectivement pour les claviers « Annie » (Figure 6) et « AZERTY »). Ce temps théorique est obtenu pour la saisie du texte suivant (exemple contenu dans les polices de caractère de windows) : « voix ambiguë d'un coeur qui au zephyr prefere les jattes de kiwis ».

C=31 E=10	Hick-Hyman	Fitts	KLM		
			Débutant	Moyen	Expert
Annie	0,659	27,77	112,26	86,1	66
AZERTY	0,659	31,11	112,26	86,1	66

Tableau 2 : Résultats de SOKEYTO (C(pixels) = largeur de la touche et E (pixels) = écart entre les touches)

CONCLUSION ET PERSPECTIVES

Nous avons vu, grâce aux résultats que les modèles neurophysiologiques ne sont pas adaptés aux personnes handicapées et que les comportements moteurs des sujets valides et handicapés sont différents. Par conséquent, notre future travail sera consacrée à l'élaboration de lois visant à mieux considérer les performances et les caractéristiques des sujets handicapés lors des tâches de saisie et de navigation. Nous envisagerons également de mesurer les effets du type de tâche, du dispositif de pointage (*trackball*, suivi du regard).

De plus, nous avons conçu l'environnement SOKEYTO permettant de générer une collection de claviers virtuel. Dans cette plateforme, nous ajouterons de nouvelles fonctionnalités permettant d'étudier l'effet des caractéristiques morphologiques de la touche (formes de touche, couleurs, espacement, etc.); la conception de claviers ambigus, l'intégration de nouvelles fonctionnalités comme l'arrangement spatial des touches ; l'ajout de plusieurs fonctionnali-

tés à chaque touche, la génération d'une configuration spatiale des touches à partir d'une analyse statistique d'un corpus de texte ; l'insertion des systèmes de prédiction linguistique; l'intégration de nouveaux modèles neurophysiologiques.

REMERCIEMENTS

Je remercie Karine Rigaldie d'avoir relu le résumé en anglais.

BIBLIOGRAPHIE.

1. Card, T.P. Moran, A. Newell, *The Psychology of Human-Computer Interaction*, 1983 Lawrence Erlbaum Associates.
2. Edmund F. LoPresti, PhD; David M. Brienza, PhD; Jennifer Angelo, PhD, OTR/L; Lars Gilbertson, PhD Neck range of motion and use of computer head controls Assistive Technology Sciences, Pittsburgh, PA; *Koester Performance Research*, Ann Arbor, MI; Department of Bioengineering and Rehabilitation Science and Technology Department, University of Pittsburgh, 5040 Forbes Tower, Pittsburgh, PA, 2003.
3. Fitts P.M., "The information capacity of the human motor system in controlling the amplitude of the movement", *Journal of experimental psychology* 47, 1954 pp.381-391.
4. Hick, W. E. On the rate of gain of information. *Quarterly Journal of Experimental Psychology*, 4, pp. 11-26, 1952.
5. Hyman R., "Stimulus information as a determinant of reaction time", *Journal of Experimental Psychology*, n°45, 1953, pp. 188-196.
6. Keates S, Clarkson J, Robinson P. Investigating the applicability of user models for motion-impaired users. *Proceedings of Assets 2000: Fourth International ACM Conference on Assistive Technologies*; 2000 Nov 20-22. pp. 129-36, 2000.
7. Leshner, G.W. & Moulton, B.J. "A method for optimizing single-finger keyboards." *Proceedings of the RESNA 2000 Annual Conference*, pp 91-93.
8. MacKenzie, I. S., & Zhang, S. Z. The design and evaluation of a high performance soft keyboard. *Proceedings of the ACM Conference on Human Factors in Computing Systems - CHI '99*, pp. 25-31. New York: ACM, 1999.
9. N. Vigouroux, F. Vella, Ph. Truillet, M. Raynal, Evaluation of AAC for text input by two groups of subjects: Able-bodied Subjects and Disabled Motor Subjects, *8th ERCIM UI4All*, Vienna (Österreich), 28-29 June 2004.
10. Scott Hudson, Bonnie John, Keith Knudsen, Michael Byrne, "A Tool for Creating Predictive Performance Models from User Interface Demonstrations", *Proceedings of the ACM Symposium on User Interface Software and Technology*, v1, n1, pp. 93-102, November 1999.

Etude de l'optimisation de claviers virtuels au travers des sujets handicapés versus valides

Frédéric Vella, Nadine Vigouroux



IRIT, DIAMANT
118, Route de Narbonne,
31062 Toulouse Cedex 4, FRANCE
{vella, vigouroux}@irit.fr



PROBLEMATIQUES

- Fatigue éprouvée par les personnes handicapées → fautes d'orthographe
- Optimisation de la saisie
 - Optimiser le déplacement du pointeur de la souris
 - Optimiser le temps de recherche d'une touche

OBJECTIFS

- Modéliser la tâche de saisie de texte pour des personnes handicapées
- Elaborer une plateforme de conception de claviers virtuels

Lois et modèles

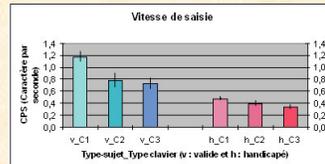
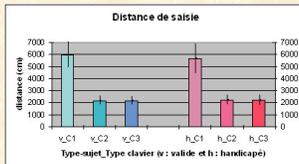
- Loi de Fitts
- Loi de Hick-Hyman
- Modèle Keystrokes-Level-Model

$$T = T_K + T_P + T_H + T_D + T_M + T_R$$

Protocole de l'expérience 1

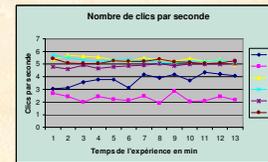
- Objectifs
 - Etudier l'usage de 3 types de claviers (AZERTY et téléphone avec et sans prédiction) sur 2 populations
 - Distance
 - Vitesse
- Hypothèses
 - H 1 : Distance AZERTY > Téléphone
 - H 2 : Temps de saisie : Clavier AZERTY > clavier téléphone sans prédiction > clavier téléphone avec la prédiction.
- Sujets
 - 5 valides
 - 2 handicapés (myopathes)
- Résultats

	Expert
CPS correspond KLM	0,993
CPS pour sujets valides	1,16
CPS pour sujets handicapés moteurs	0,48



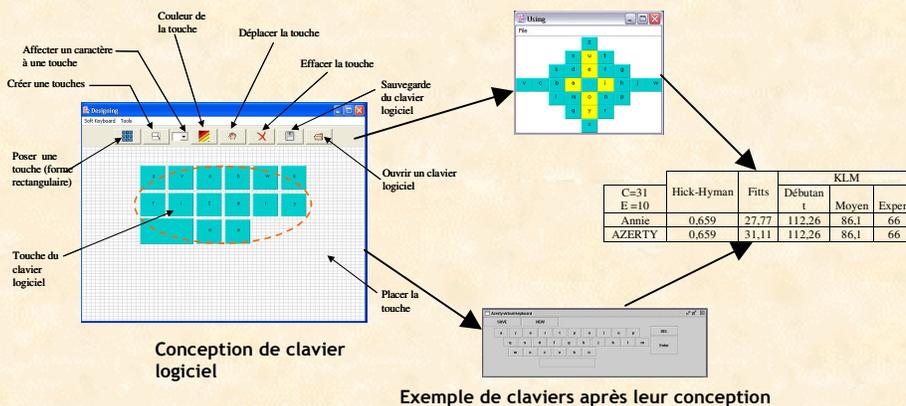
Protocole de l'expérience 2

- Objectif
 - Comparer la durée de clic sur 2 populations
- Hypothèses
 - H 1 : La durée moyenne des clics est plus importante pour les sujets handicapés moteur que pour les sujets valides
 - H 2 : Les symptômes de fatigabilité sont plus précoces pour les sujets handicapés que pour les sujets valides pendant l'expérience
- Sujets : 2 handicapés et 4 valides
- Résultats



Environnement de conception SOKEYTO

- Objectifs :** Disposer d'un outil de conception de clavier logiciel répondant aux besoins de personnes handicapées moteur ;
Disposer d'une base de ressources de clavier pour le laboratoire des usages (démonstration, apprentissage, test) ;
Disposer de divers « plug-ins » de dispositifs d'interaction adaptés à cette population ;
Et mise en œuvre d'une **approche de conception centrée utilisateurs** (finaux et cliniciens).



Perspectives :

- Étudier et améliorer la représentation du clavier : formes de touche (cercle, triangle,...), arrangement spatial des touches ;
- Ajout de la fonction recouvrement entre les touches ;
- Étendre la conception à des claviers de navigation et à d'autres langages ;
- Insérer des systèmes de prédiction linguistique et de configuration spatiale automatique ;
- Ajouter un système d'oculométrie ;
- Rendre disponible sous plate-forme PC et Mac.

CONCLUSION

- Le clavier C1 plus performant que C2 et C3
- Modèle neurophysiologique non adaptés ↔ Comportement des sujets valides différent par rapport aux sujets handicapés
- Conception de l'environnement SOKEYTO

PERSPECTIVES

- Elaborer de lois visant à mieux considérer les performances et les caractéristiques des sujets handicapés
- Mesurer les effets du type de tâche, du dispositif de pointage (*trackball*, suivi du regard)



Présentations courtes

Réalité Mixte et Co-Adaptation

Pascal Costa-Cunha

Laboratoire de Recherche en Informatique (LRI) ¹
Bâtiment 490 – Université Paris Sud
91405 Orsay Cedex, France
costa@lri.fr

RESUME

Mon travail sur la conception d'un cahier de laboratoire augmenté pour les biologistes m'a amené à associer deux concepts de l'IHM : la réalité mixte et la co-adaptation. Ce papier présente le I-Book, le premier prototype que j'ai réalisé en insistant sur deux points. Tout d'abord, comment, en utilisant la technologie Anoto, j'ai pu réaliser un système de réalité mixte simple à mettre en place et à utiliser dans des conditions réelles. Et comment, en utilisant l'interprétation, un moyen simple de collecter les besoins et envies des utilisateurs, toujours dans des conditions réelles d'utilisation, le I-Book est pensé afin de favoriser la co-adaptation entre un système et son utilisateur.

MOTS CLES : Réalité Mixte, Co-adaptation, Interface Papier, Technologie Anoto, Conception Participative.

INTRODUCTION

La Réalité Mixte

Lors de la conception d'un nouveau système, l'approche suivie est bien souvent de remplacer l'existant par un nouveau système complètement informatisé. La Réalité Mixte (ou réalité augmentée) peut être vue comme une alternative qui consiste à conserver au maximum l'interface et les interactions physiques existantes tout en les augmentant, c'est-à-dire, en y ajoutant des fonctionnalités informatiques.

Cette voie a été ouverte par le Digital Desk de Wellner [11], qui, à l'aide d'une caméra et d'un projecteur situé au-dessus d'un bureau, permettait de matérialiser la métaphore du bureau informatique et de travailler conjointement avec des outils virtuels et des documents papier. Ce travail a inspiré de nombreux autres travaux principalement sur des tableaux blancs augmentés [8, 10] ou des bureaux augmentés [2] qui tous se fondent sur l'utilisation d'interfaces tangibles. Leur mise en place est bien souvent fastidieuse car ces systèmes mettent en jeu des technologies peu flexibles, par exemple le couple caméra projecteur très utilisé doit être parfaitement calibré et donc immobile.

La Co-adaptation

La co-adaptation est un phénomène observé a posteriori [6]. Un nouvel outil change le comportement de son utilisateur en lui apportant de nouvelles possibilités, et, parallèlement, l'utilisateur adapte le système pour qu'il concorde avec ses attentes, parfois même de manière non prévue par le concepteur du système.

Au cours de mon travail sur la conception d'un cahier de laboratoire augmenté pour les biologistes, j'ai vite pu constater que les cahiers de laboratoire, bien qu'étant un outil basique (ce n'est qu'un banal cahier), étaient utilisés d'une manière personnelle par chaque biologiste. Pour faciliter les recherches ultérieures dans le cahier, certains utilisent des post-its comme marque-pages, d'autres des codes couleurs, certains font entièrement confiance à leur mémoire. Et bien qu'à l'Institut Pasteur, il soit interdit de coller quoi que ce soit sur le cahier ou d'écrire au crayon de papier, la plupart des biologistes passent outre ces interdictions. Bien qu'ayant des problèmes similaires, les solutions qu'ils utilisent sont tellement variées qu'il semble peu probable de faire un cahier de laboratoire augmenté qui soit adapté à chaque biologiste, surtout si ils souhaitent garder leur liberté face à leur outil. Le cahier peut être vu comme un outil co-adaptatif, le cahier étant personnalisé par son utilisateur qui en crée de nouvelles utilisations, mais le cahier modifie peu le comportement de son utilisateur. Augmenter le cahier permet de garder la flexibilité du papier et d'apporter la puissance de calcul d'un ordinateur, qui peut, elle, réellement modifier les habitudes de travail des biologistes.

L'article présente le premier prototype que j'ai réalisé, le I-Book, en insistant sur l'utilisation qui en est faite. Utilisation rendue possible par sa simplicité d'utilisation, malgré le fait qu'il s'agit d'un outil de Réalité Mixte, et utilisation rendue utile pour la conception future du prototype, car pensé dès sa conception comme un outil favorisant la co-adaptation. Et on conclura par le début d'évaluation qui en a été fait en tant que sonde technologique.

¹ projet In Situ, Pôle Commun de Recherche en Informatique, CNRS, Ecole Polytechnique, INRIA, Université Paris Sud

LE I-BOOK

Les Outils

Mon travail s'inscrit dans la prolongation de travaux [7] déjà effectués dans l'équipe, au cours duquel trois prototypes de cahier de laboratoire augmenté ont déjà été réalisés. Comme précédemment, la méthode de conception participative a été suivie, et cela avec des biologistes de l'Institut Pasteur et de l'université d'Orsay. J'ai essentiellement réalisé des interviews de nombreux biologistes et j'ai également organisé un brainstorming avec quelques biologistes. Le but étant ici d'introduire une nouvelle technologie, le stylo Anoto (www.anoto.com) dans leur cadre de travail. Mais ici, comme on le verra dans la dernière section de l'article, la méthode utilisée n'est pas uniquement d'impliquer les utilisateurs finaux pendant le design du système et lors des phases de tests, mais également de leur permettre de modifier le prototype, eux-mêmes et pendant une utilisation dans le contexte réel, comme bon leur semble. Tout ce travail a été fait en gardant à l'esprit le but qui est de faire un outil co-adaptatif.

La technologie Anoto permet d'avoir une copie digitale, que l'on nommera *cahier digital*, de tout ce qui est écrit sur un cahier. Comme si on disposait d'une tablette graphique sur chaque page d'un cahier, mais ici, on a uniquement un stylo et un cahier, sans aucune connectique particulière. Pour chaque trace faite sur le cahier, le stylo conserve sur quelle page il a été fait, la position sur cette page, et quand il a été fait (comme les données se sont pas envoyées à un PC en temps réel, l'information temporelle est très importante). Etant donné le fonctionnement du stylo Anoto, qu'on peut qualifier d'asynchrone, mon prototype se scinde en deux parties distinctes. D'une part, la partie off-line, pour l'édition, qui se compose du cahier de laboratoire, du stylo et d'une *feuille de commande*. Et la partie on-line, qui se trouve sur l'ordinateur, et facilite la recherche d'informations dans la version digitale du cahier.

La Partie Off-Line

Comme le montre la figure 1, la partie off-line du I-Book se compose d'un cahier et d'un stylo ayant les fonctionnalités Anoto. Le nouveau système n'apporte donc qu'une unique contrainte par rapport à l'ancien système : l'utilisation d'un seul stylo, spécifique. Tout ce qui sera écrit avec un autre stylo, comme par exemple avec un stabylor pour souligner des titres ne sera pas pris en compte dans la version digitale du cahier. Mais détail important, rien n'empêche le biologiste de le faire. En fait, il a autant de liberté face à ce nouvel outil qu'avec son cahier habituel. Et surtout, malgré l'augmentation du cahier que l'on décrira au prochain paragraphe, le I-Book peut être utilisé comme un cahier normal, avec tout ce que cela implique en terme de robustesse comparé à un outil informatique toujours capricieux, aux bugs et pannes très problématiques dans le cadre professionnel.



Figure 1: la partie off-line du I-Book, avec le cahier, le stylo Anoto et la *feuille de commande*.

L'augmentation du cahier consiste à permettre au biologiste de structurer le *cahier digital* mais cela non pas dans la partie on-line du prototype, ce qui entraînerait une double phase d'édition, mais directement sur le cahier papier. Et cela, à l'aide des boutons situés sur la *feuille de commande*, que l'on voit au bout du stylo sur la figure 1.

Le I-Book peut faire penser au PaperLink [1]. En fait, l'interaction avec le système est très similaire. Mais le PaperLink ne permettait de travailler que sur des documents en caractères imprimés reconnus par OCR, et non pas d'éditer complètement et en parallèle un document papier et sa copie digitale.

Les boutons de la *feuille de commande* sont de trois sortes : *lien*, *titre* et *annexe*. Leur fonction sera détaillée dans la section sur la partie on-line du cahier étant donné que c'est là qu'elle prend effet. Une dernière chose à souligner sur l'interface off-line de ce prototype, c'est qu'elle est très robuste et simple à utiliser. Comme il a été dit plus haut, il peut être utilisé comme un cahier papier ordinaire mais aussi, l'ajout de cette feuille de commande, qui s'utilise par click et sélection à l'aide du stylo ne rompt pas cette robustesse. En effet comme détaillé dans [3], l'interaction avec la feuille de commande est très intuitive et la reconnaissance qu'en fait le système est suffisamment souple pour générer très peu d'erreur quoi que fasse l'utilisateur.

La Partie On-Line

La partie logicielle du I-Book se divise en deux zones : le sommaire (à gauche dans la figure 2) et la zone d'affichage du cahier au centre.

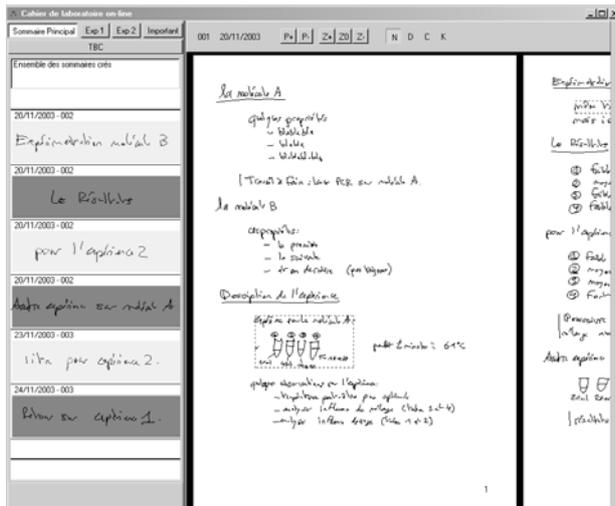


Figure 2: la partie on-line du I-Book.

Comme son nom l'indique la zone d'affichage du cahier affiche la copie digitale du cahier. Elle est de très bonne qualité, la technologie Anoto permettant d'avoir une copie quasi conforme de tout ce qui a été écrit, seule quelque perte de précision ont été constatée et sur des écritures effectuées très rapidement. Dans cette affichage se distinguent les zones marquées par le bouton *lien*, par l'intermédiaire d'un encadré et du curseur de la souris qui change de forme pour indiquer le lien. Un lien s'établissant toujours entre deux zones du cahier, un click sur l'une des zones permet de naviguer rapidement vers l'autre et cela de manière bi-directionnelle.

En haut du sommaire est situé un onglet pour chaque bouton de la feuille de commande de type *titre* ou *annexe* plus un onglet sommaire principal. Dans la figure 2, c'est ce dernier qui est sélectionné et on peut y voir que les éléments du sommaire sont de deux couleurs différentes. En fait, ce qui est appelé sommaire principal contient toutes les zones du cahier qui ont été sélectionnées avec un bouton de type *titre*. Dans l'exemple, il y avait deux boutons de ce type, d'où les deux couleurs.

En sélectionnant un onglet correspondant à un bouton *titre*, seul les éléments associés à ce bouton apparaîtront dans le sommaire. Bien évidemment, ce sommaire permet de naviguer rapidement dans le *cahier digital*, par simple click, comme pour les liens. De la même manière, en sélectionnant un onglet associé à un bouton de type *annexe*, les zones du cahier indiquées par le biologiste comme faisant partie de cette *annexe* apparaissent dans le sommaire. A part d'apparaître dans l'onglet sommaire principal, la différence entre un *titre* et une *annexe*, c'est que les titres structurent le cahier en parties de différentes catégories. Tout ce qui suit un *titre* jusqu'au prochain *titre*, est de la catégorie de ce premier *titre*. Le cahier est donc structuré en parties de différentes catégories et l'utilisateur peut choisir de ne visualiser qu'une seule de ces catégories à la fois.

Ces trois types de boutons ont été choisis en fonction des informations recueillies auprès des biologistes, avec pour but de minimiser le nombre de type différents. Le type *titre* correspond principalement aux expériences menées par les biologistes. En effet, les biologistes écrivent de manière chronologique dans leur cahier, mais ils effectuent parallèlement plusieurs expériences qui s'étendent sur des semaines ou des mois. L'utilisation de bouton *titre* leur permet d'avoir une vision thématique du cahier et non plus une vision strictement chronologique. Les boutons *annexe* peuvent avoir différentes utilisations, comme par exemple permettre de répondre à une recherche du type : où ai-je utilisé la molécule M ? Où est-je suivi le protocole P ? Quels sont les schémas que j'ai réalisés ? Ils permettent de réaliser toute sorte d'annexe.

Aspects Co-Adaptatifs

On voit que les possibilités offertes par les trois types de bouton permettent de créer un grand nombre de boutons différents. Mais au lieu d'avoir un outil complexe avec de nombreuses fonctionnalités difficiles à assimiler et, de plus, pas forcément pertinentes pour chaque utilisateur, la *feuille de commande* est initialement vide et c'est au biologiste de créer les boutons qu'il souhaite utiliser. Par exemple, les boutons *titre* doivent correspondre avec le nombre d'expériences en cours du biologiste, et des noms comme 'expérience 1', 'expérience 2' sont sans comparaison avec un nom choisi par le biologiste pour décrire son travail. Cette personnalisation du prototype par son utilisateur est très simple à réaliser, il suffit de dessiner un bouton sur la feuille de commande en lui donnant un nom ou un symbole. Le biologiste donne le sens des boutons qu'il crée, mais ne communique au système que le type du bouton parmi les trois possibles. Le système ne cherche pas à comprendre la sémantique du bouton, ce qui en fait, ici encore, un système très robuste et très flexible comme expliqué plus longuement dans [4]. Ce type de personnalisation, où, à la manière d'un tableur que l'utilisateur peut remplir avec un relevé de compte ou un emploi du temps sans que le système ne le sache, peut être rapproché de l'*interpretive flexibility* de Orlikowski [9] et est la raison du nom I-Book, pour Interpretive Book.

Mis à part cet aspect qui a été pensé pour favoriser la co-adaptation dès la conception du prototype, une autre particularité du système peut être détournée par les futurs utilisateurs. J'ai implémenté dans la partie on-line un dispositif de recherche dans le cahier par date, qui pour une date donnée affiche ce qui a été écrit à cette date de couleur vive et ce qui a été écrit quelques jours plus tôt ou plus tard de plus en plus fade, cela afin de pouvoir affiner interactivement la recherche. Cela a été implémenté pour permettre aux biologiste de continuer à utiliser les recherches chronologiques qu'ils utilisent déjà sur leur cahier papier mais cela a été vu par certains comme un moyen d'établir des liens entre des parties du cahier. En

effet ce système qui affiche d'une couleur vive tout ce qui a été écrit le même jour permet d'établir un lien entre une annotation à un endroit du cahier et ce qui a été fait ce jour-là expliquant cette annotation. Le cahier devenant alors un outil réflexif permettant à un biologiste d'analyser son propre travail, ce que le cahier papier, parfois un peu brouillon fait très peu.

PERSPECTIVES

Grâce à sa robustesse, le I-Book peut être utilisé dans un contexte réel, et grâce à sa flexibilité, il favorise (et même force) son appropriation par son utilisateur. C'est pourquoi il a été utilisé comme sonde technologique [5], afin de recueillir les personnalisations faites par ses utilisateurs durant leur travail. Le I-Book n'a pour l'instant été confié qu'à une seule biologiste et pour une courte durée de trois semaines, mais ce premier test, qui sera suivi par d'autres, a été très encourageant. Le I-Book s'est bien montré simple d'utilisation et a été utilisé activement par cette biologiste.

La partie off-line du prototype a été très appréciée et les recommandations faites par les biologistes ayant vu ou utilisé ce prototype portent pour l'essentiel sur la partie on-line du prototype. Et donc, parallèlement aux futurs tests, mon travail portera sur comment porter les concepts favorisant la co-adaptation de la partie off-line vers la partie on-line, et également en trouver d'autres pour ne pas limiter les capacités d'adaptation de l'outil informatique.

CONCLUSION

De par son utilisation d'outils physiques familiers, la réalité mixte semble très appropriée pour réaliser un outil co-adaptatif. Le I-Book est un outil qui s'intègre très facilement au milieu de travail pour lequel il a été conçu, nécessitant très peu d'apprentissage, ne perturbant pas les habitudes de travail existantes et favorable à l'appropriation de l'outil par son utilisateur, prérequis à l'apparition de comportements co-adaptatifs.

REMERCIEMENTS

Merci à Wendy Mackay pour ses réflexions toujours pertinentes et à tous les biologistes ayant participé à l'élaboration de ce prototype durant cette première année, en particulier Laurence Decourty et Florence Hantraye de l'Institut Pasteur.

BIBLIOGRAPHIE

1. Arai, T., Aust, D. & Hudson, S (1997) PaperLink: A Technique for Hyperlinking from Real Paper to Electronic Content. *In Proc. CHI 1997* pp. 327-334.
2. Brygg, U. and Hiroshi, I. (1997) The metaDESK : Models and Prototypes for Tangible User Interfaces. *In Proc. UIST 1997* pp. 223-232.
3. Costa-Cunha, P. & Mackay, W.E. (2003) Papier augmenté et stylo Anoto. *In Proc. of IHM 2003*, pp. 232-235.
4. Costa-Cunha, P. (2004) The I-Book: a freely interpretive notebook for biologists. *In Proc. of PDC 2004, Volume II*, pp 159-162.
5. Hutchinson, H., Mackay, W.E., Westerlund, B. et al (2003) Technology probes: Inspiring design for and with families. *In Proc. of ACM CHI 2003*, p. 17-24. ACM Pres
6. Mackay, W.E. (1990). Users and Customizable Software: A Co-Adaptive Phenomenon, Ph.D. Thesis. Massachusetts Institute of Technology.
7. Mackay, W.E., Pothier, G., Letondal, C., Bøegh, K., Sørensen, H. (2002) The Missing Link : Augmenting Biology Laboratory Notebooks. *In Proc. User Interface Software and Technology (UIST2002)*, Paris, France, ACM Press, p. 41-50.
8. McGee, D., Cohen, P. & Wu, L. (2000) Something from nothing: augmenting a paper-based work practice with multimodal interaction. *In Proc. of DARE'00*, Copenhagen: ACM. p. 71-80.
9. Orlikowski, W.J. (1992) The Duality of Technology: Rethinking the Concept of Technology in Organizations. *In Organization Science*, Vol. 3, No. 3, 1992, p. 398-427.
10. Stafford-Fraser, Q., Robinson, P. (1996) BrightBoard: A Video-Augmented Environment. *In Proc. Human Factors In Computing Systems (CHI'96)*, Vancouver, ACM Press, p. 134-141.
11. Wellner, P. (1993) "Interacting with paper on the DigitalDesk." *Communications of the ACM*, 36(7):86-96, July 1993.

Le I-Book : un nouvel outil pour la Conception Participative

Pascal Costa-Cunha (costa@lri.fr)

Laboratoire de Recherche en Informatique de Paris-Sud - Projet In Situ

But : Réaliser un cahier de laboratoire 'augmenté'

Méthode : Conception Participative avec les biologistes

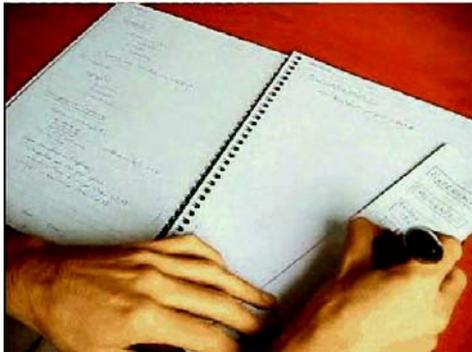
Placer les futurs utilisateurs au centre du processus de conception du système.

Interviews sur leur lieu de travail, brainstorming avec les biologistes.

Comprendre leurs **besoins** mais aussi leurs **habitudes** de travail.



Le Prototype : Le I-Book (Interpretive Book)



Interface papier **robuste** et **flexible**.

Grâce à la technologie Anoto¹, outil identique à leur cahier habituel permettant de construire une copie digitale du contenu du cahier.

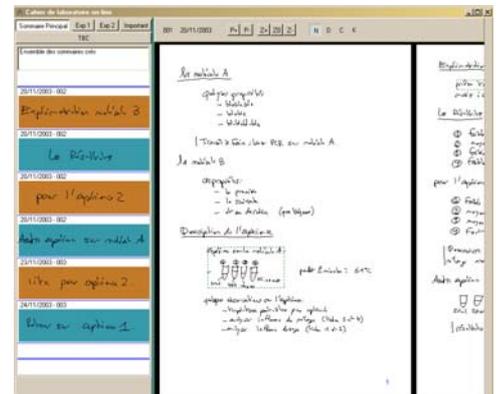
Ajout d'une *feuille de commande*, contenant des boutons pour structurer cette version digitale du cahier :

- structuration thématique
- gestion d'annexes
- lien entre parties du cahier

Version digitale pour **visualisation** du cahier, **recherche** et **partage** d'informations (édition uniquement sur papier).

Robustesse : interaction avec la *feuille de commande* ne générant que très peu d'erreurs.

Flexibilité : le biologiste crée les boutons sur sa *feuille de commande*, en leur donnant le sens qu'il souhaite.



Le I-Book utilisé comme sonde technologique²

Outil simple à prendre en main et non sujet aux pannes ou bugs (la version papier fonctionne toujours).

⇒ peut être déployé dans le contexte de travail

Outil personnalisable, les biologistes créent leurs boutons sur la *feuille de commande* (initialement vide).

⇒ force les biologistes à réfléchir sur leur outil

Permet de capturer, **dans** et **pendant** le contexte réel d'utilisation, les besoins et envies des biologistes.

Conception participative minimisant l'influence du concepteur non-utilisateur, non présent pendant la phase créative de sens par les utilisateurs réels.

¹ Anoto Group : www.anoto.com

² H. Hutchinson, W. Mackay, B. Westerlund, et al. **Technology probes: Inspiring design for and with families.** In *Proceedings of CHI 2003*.

Modélisation de tâches contextualisées pour la construction d'applications collaboratives

Olivier Delotte, Bertrand David

Laboratoire ICTT
Ecole Centrale de Lyon
36, av. Guy de Collongue
69134 Ecully Cedex
olivier.delotte@ec-lyon.fr

RESUME

Cet article décrit une méthode pour la construction d'un modèle comportemental pour l'élaboration de systèmes collaboratifs capillaires. Le point de départ de cette démarche est la constitution d'un ensemble de scénarios d'usage. L'activité et les préférences des acteurs sont au cœur du modèle, et sont prises en compte en amont de l'intégration ainsi qu'au cours de l'utilisation. De plus, nous détaillons notre outil CBME (Contextualized Behavioral Model Environment) qui permet la création et la vérification d'un modèle comportemental comprenant un éditeur de tâches contextualisées. Pour les concepteurs de systèmes capillaires sensibles au contexte, il permet de visualiser et de manipuler les différents éléments du modèle comportemental pour contrôler la cohérence du modèle.

MOTS CLES : Scénarios, Systèmes Capillaires, CSCW, Modélisation des tâches, Approche Model-Based.

INTRODUCTION

Le monde du TCAO est encore aujourd'hui essentiellement tourné vers des dispositifs fixes disposant de connexions haut-débit. Cependant, le nomadisme des travailleurs, et donc des utilisateurs de systèmes collaboratifs change la donne. Dans nos précédents articles présentés à IHM [6] [7], nous avons positionné nos travaux au niveau bibliographique et par rapport aux projets existants. Le travail coopératif capillaire [4] qui marie le synchrone et l'asynchrone dans la mobilité des personnes et des informations, tente de prendre en compte le contexte des sujets et des objets de l'action. La diversité des dispositifs fixes et mobiles et leurs caractéristiques, notamment d'affichage et d'interaction, doivent être prises en compte par une démarche de développement multi plates-formes. Dans [6], nous décrivons une démarche théorique pour la construction et l'évolution de systèmes collaboratifs sensibles au contexte (figure 1). Nous nous situons dans la mouvance Model-based Approach, MDA ou MDE [10]. Il s'agit de collecter des scénarios exprimant de façon parcellaire, souvent tâche par tâche et acteur par ac-

teur, des situations d'utilisation souhaitées. La première étape vise à construire, à partir d'un ensemble de scénarios d'utilisation collectés, un modèle comportemental qui synthétise ces informations afin de caractériser le comportement du système coopératif visé. La seconde étape est une phase d'adaptation du modèle comportemental à l'architecture collaborative capillaire en tenant compte des caractéristiques des matériels utilisés et des outils logiciels déjà disponibles. La dernière étape concerne l'évolution déclenchée lors de la modification du système conformément aux souhaits des utilisateurs exprimés à travers de nouveaux scénarios. Le système collaboratif peut co-évoluer avec le modèle comportemental afin de proposer de nouveaux services et de nouvelles manières d'interagir aux utilisateurs.

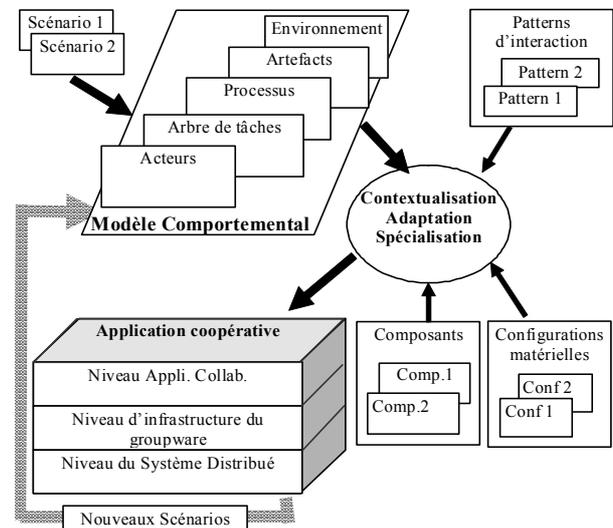


Fig. 1. Processus de conception d'applications collaboratives sensibles au contexte d'usage.

DEFINITION DU CONTEXTE D'UTILISATION

Nous nous appuyons sur la définition de A.K. Dey [8], pour qui le contexte regroupe toutes les informations qui peuvent être utilisées pour caractériser la situation d'une entité. Une entité est une personne, un endroit ou un objet qui relève de l'interaction entre un utilisateur et une application, eux-mêmes inclus. Nous choisissons cette définition avec comme objectif d'associer

au contexte d'usage d'un utilisateur les activités des autres acteurs. Ainsi la première étape dans le processus d'intégration du contexte est la recherche des informations et, ensuite, leur interprétation dans le système collaboratif. Beaucoup de recherches [5] se font actuellement autour du captage de l'environnement de l'utilisateur avec l'aide de puces électroniques (bruit, position,...). Pour utiliser ces informations dans un processus de conception d'applications sensibles au contexte, nous pensons qu'elles doivent être couplées avec l'arbre des tâches afin de faciliter l'adaptation (des IHM et des Services). Dans le cadre de systèmes collaboratifs, la conscience des autres (awareness) est un élément supplémentaire nécessaire à l'activité de groupe et le système doit proposer à chaque utilisateur une interface adaptée en fonction des tâches qu'il est en train d'effectuer et des possibilités des autres utilisateurs. C'est pourquoi, nous avons opté pour l'utilisation de scénarios d'utilisation comme formalisme de description des activités dans leur contexte d'utilisation.

Les scénarios d'utilisation comme point de départ

Un scénario peut prendre plusieurs formes : soit un texte libre décrivant une activité, soit un texte structuré avec un certain nombre de champs, soit une forme graphique ou textuelle telle que proposée par les cas d'utilisation d'UML [9]. Carroll [2] [3] définit les scénarios comme des récits d'utilisation et comme une description narrative d'une activité, située dans un contexte précis. Le système capillaire, que l'on souhaite construire, a pour but de prendre en considération les différents scénarios ponctuels collectés. Ceux-ci sont donc progressivement repris et « recoupés » pour servir lors de la construction du modèle comportemental. Nous allons donc détailler ses principaux constituants de ce dernier.

Le modèle comportemental

La section Acteurs contient la liste de tous les acteurs évoluant dans le système avec leurs caractéristiques qui, outre leurs noms (génériques), comportent leurs rôles potentiels. Ces derniers expriment la façon dont ils interviennent dans les activités et agissent avec et sur les artefacts.

La section Processus regroupe à la fois les arbres des tâches avec leurs pré-conditions et les processus donnant l'organisation (succession, parallélisme, dépendances) logique, fonctionnelle ou temporelle de ces tâches.

Quant à la section Artefacts, elle identifie les outils logiciels (avec quoi on travaille) et les classes d'objets (sur quoi on travaille).

La section Environnement réunit les informations relatives à la localisation potentielle des acteurs (position géographique, situation d'utilisation), la situation de l'activité dans le temps ou l'action déclenchante. En effet, la situation « chez le client » (localisation) est cataloguée comme contexte d'utilisation, tout comme « après la réparation » (position dans la séquence de tâches) ou « en se connectant au serveur de l'entreprise » (événementiel).

DES SCENARIOS VERS LE MODELE COMPORTEMENTAL

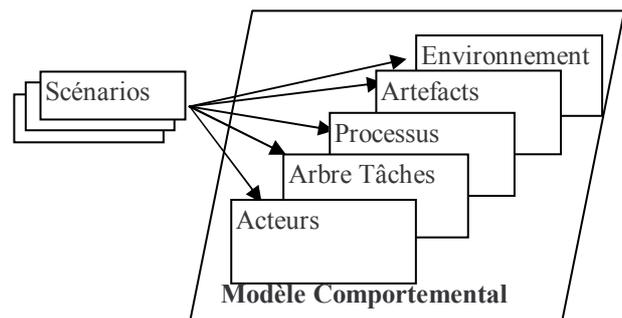


Figure 2 : Transformation des scénarios en modèle

La transformation des scénarios en modèle consiste donc à extraire les informations contenues dans les scénarios en les positionnant dans les différentes sections du modèle comportemental. Nous prenons chaque récit et nous appliquons un découpage.

Exemple de décomposition d'un scénario :

« Le matin, le technicien consulte sur son PDA la liste des opérations de la journée. »

Le matin devient une étiquette de l'environnement. Le technicien est l'acteur principal. De cette manière, on décompose intégralement le scénario. Comme dit précédemment, un scénario ne peut couvrir intégralement l'ensemble du modèle comportemental. C'est pourquoi, un corpus de scénario pour chaque activité doit être créé. Aujourd'hui, la décomposition d'un ensemble de scénarios a permis de fabriquer un corpus de tâches récurrentes pour la journée d'un technicien d'une société de dépannage. La mise à disposition de ces informations et leur hiérarchisation forment le modèle comportemental. Reste le problème de la mise en forme de plusieurs scénarios disjoints ou complémentaires. Comment reconnaître et éliminer les redondances au sein du modèle ? Quelle méthode de construction à employer ? C'est à ces réponses que je souhaite répondre dans le paragraphe suivant.

Construction du MC avec plusieurs scénarios

La construction du modèle comportemental se fait par recouplement de plusieurs scénarios et par réification.

Reprenons l'exemple 1 et un autre scénario pioché dans le corpus. La première étape reprend le découpage précédent avec le second scénario. Ainsi, on obtient 2 arbres de tâches. La mise en relation de ces 2 éléments est donc une partie importante dans la réification. Décomposons étape par étape :

- Découpage de tous les éléments qui ne sont pas des actions. Intégration de l'activité dans le module Arbre des tâches.
- Association de l'activité avec les éléments du modèle déjà insérés.

A travers la construction, on discerne 4 modes d'actions dans la journée d'un réparateur : transmission d'informations, d'interaction, d'organisation-supervision, approche métier.

Construction du modèle de tâches

De nombreux travaux se sont penchés sur les formalismes de modélisation des tâches. Peu ont aboutis à des outils utilisables [11]. Cependant, les travaux sur CTT [12] et sur la modélisation de tâches sensibles au contexte [13] sont des recherches sur lesquelles nous pouvons nous appuyer pour construire notre modèle de tâches. Nous utilisons actuellement le formalisme CTT pour sa complétude. L'outil d'édition et de visualisation graphique CTTE n'étant pas intégrable dans notre projet, nous avons décidé de mettre en place notre propre outil (édition et visualisation). Il permet d'exprimer visuellement les liens entre plusieurs tâches collaboratives, comme CTTE, mais en intégrant les modèles précédemment définis: les acteurs, les artefacts et leurs environnements. De plus, l'outil doit gérer les liens entre les autres éléments de chaque modèle du MC.

Validation du modèle comportemental

La validation du modèle comportemental se fait selon trois axes : complétude, exactitude et cohérence. Il s'agit donc de mener des validations formelles et informelles, avec recours aux utilisateurs. Les validations formelles portent sur la complétude de l'arbre des tâches et sur la complétude de l'utilisation d'opérations proposées par des classes. Les utilisateurs doivent intervenir lors de la validation sémantique qui peut s'appuyer sur différentes représentations. Ces représentations peuvent montrer pour chaque acteur (ou rôle) son implication dans le système, c'est-à-dire dans les opérations issues de l'arbre de tâches, les classes manipulées et les opérations autorisées et/ou utilisées.

Pour pouvoir manipuler correctement le modèle, il est nécessaire de conserver, tout au long de la construction, les liens entre les scénarios sources et les éléments

du modèle. De cette manière, il est possible de valider les différentes parties du modèle ou d'apporter des modifications pour le rendre cohérent. Par exemple, la liste des acteurs issus des scénarios peut être redondante avec des synonymes, ... On peut alors « nettoyer » cette liste et, par conséquent, modifier le modèle et les scénarios sources. Il en est de même pour les autres aspects du modèle.

Plus globalement, les représentations basées respectivement sur les acteurs, tâches, processus, artefacts et les environnements permettent d'étudier formellement et sémantiquement le modèle pour l'observer, le compléter, le modifier et finalement le valider. La transposition des scénarios vers le modèle comportemental est réalisée manuellement grâce à l'outil CBME (fig. 3), explicité ci-après.

L'OUTIL CBME

Contextualized Behavioral Model Environment (CBME) est un outil de visualisation graphique du modèle comportemental. Il permet le passage entre les scénarios (sous forme de texte) et le modèle comportemental (fig. 3). En effet, cet outil est capable de prendre un scénario et d'aider le concepteur à l'insérer dans le modèle comportemental.

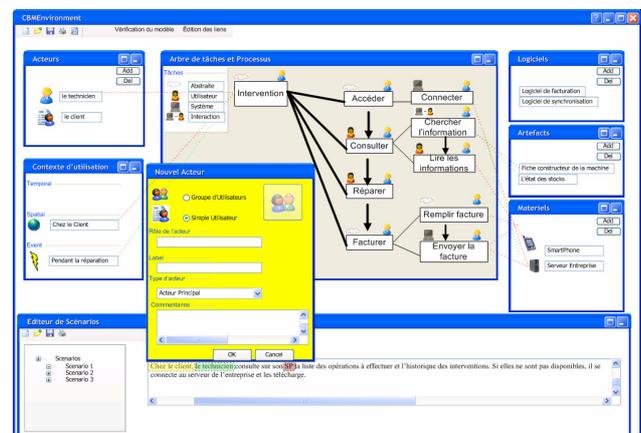


Fig.3 Prototype CBME

FONCTIONNALITES DU PROTOTYPE CBME :

Construction graphique : l'utilisateur sélectionne les mots du scénario et les dispose dans la section correspondante. Une zone de description permet de caractériser le nouvel élément du modèle.

Edition des liens entre les tâches : Il s'agit de construire l'arbre des tâches pour chaque activité et d'établir des liens appropriés entre les tâches, les processus et les artefacts concernés.

Vérification du modèle : L'outil permet de vérifier la complétude du modèle en s'assurant la cohérence des relations entre les éléments du modèle. La complétude

sémantique est contrôlée par observation des liens entre les éléments. Des modifications peuvent être effectuées afin de corriger le modèle.

VISUALISATION DU MODELE COMPORTEMENTAL DANS CBME

Le premier problème réside dans la visualisation d'informations hiérarchisées. Comment le concepteur peut visualiser l'ensemble des tâches disponibles dans telle situation et modifier le modèle comportemental ? C'est dans l'optique de répondre à cette question, que nous orientons nos recherches, en particulier, vers les travaux de visualisation d'information et d'ergonomie des logiciels. La manipulation des données est le second objectif et changer le modèle en insérant, déplaçant, supprimant des composants du modèle comportemental.

Afin d'illustrer notre démarche, nous travaillons sur un ensemble de scénarios d'utilisation tirés de l'activité d'une société de dépannage, dont voici un exemple : « Chez le client, le technicien consulte la liste des opérations à effectuer et l'historique des interventions. Si elles ne sont pas disponibles, il se connecte au serveur de l'entreprise et les télécharge. »

Nous utilisons l'éditeur de scénarios (fig. 3) de CBME, afin de le décomposer en sélectionnant chaque bloc de mots (avec conservation du marquage) vers l'une des quatre fenêtres représentant les 4 sections de la grille. Selon le type de données, une boîte de dialogue apparaît et permet une caractérisation plus fine du nouvel élément.

Le système conserve en mémoire les liens entre chaque élément du scénario et les représente graphiquement, selon une typologie de couleurs. Dans la partie Arbre de tâches et Processus, des icônes représentent les acteurs pour chaque tâche. L'édition manuelle de liens est nécessaire à la création du séquençement des actions et à la mise en relation des éléments du modèle.

CONCLUSION ET PERSPECTIVES

Dans cet article, nous avons détaillé l'étape de transformation d'un ensemble de scénarios en un modèle de comportement synthétisant toutes les activités à proposer dans un système collaboratif capillaire spécialisé en cours de construction. Nous avons également présenté l'outil chargé de cette transformation et de la validation du modèle bâti. Enfin, nos travaux sont en cours pour modifier la visualisation et l'utilisabilité de CBME. Parallèlement nos travaux futurs, comme déjà décrits dans [3], c'est l'utilisation de ce modèle comportemental et sa projection (ou celle de ses éléments) sur une architecture logicielle dont le modèle a été prédé-

fini sont dans nos perspectives prochaines. Des recherches se font déjà dans cette direction [1].

BIBLIOGRAPHIE

1. Abi-Aad R., Sinnig D., Radhakrishnan T.; Seffah A., CoU: Context of Use Model for User Interface Designing, in Proceedings of HCI International 2003, vol. 4, (Greece), pp. 8 – 12, LEA, June 2003. HCI International 2003, ISBN: 0262032791.
2. Carroll J.M., *Making Use: Scenario-Based Design of Human-Computer Interactions*. The MIT Press, 2000, ISBN: 0262032791.
3. Carroll J., Rosson M.B., McInerney P., Scenarios in Practice, CHI Workshop 2003.
4. David B.T., Chalon R., Vaisman G., Delotte O., Capillary CSCW. In *Proc. Of IHCI*, Crète, 2003.
5. De Guzman Edward S., Yau M., Gagliano A., Park A., Dey A.K., Exploring the Design and Use of Peripheral Displays of Awareness Information, CHI 2004. ISBN:1-58113-703-6
6. Delotte, O., David, B., Chalon, R., Environnement et démarche pour la construction et l'évolution de systèmes collaboratifs. *Conférence IHM'03*, Caen, 2003, ISBN : 1-58113-803-2.
7. Delotte, O., David, B., Chalon, R., Construction d'un modèle comportemental pour le travail coopératif capillaire. *Conférence IHM'04*, Namur, 2004, ISBN : 1-58113-926-8.
8. Dey A.K., Providing Architectural Support for Building Context-Aware Applications, PhD Thesis, Georgia Institute of Technology, 2000.
9. Larman, C., *UML et les Design Patterns (2^e Ed.)*. Campus Press, 2003, ISBN: 2744016233.
10. Object Management Group : www.omg.org
11. Paris C., Lu S., Vander Linden K., *Environments for the Construction and Use of Tasks Models*, In The Handbook of Task Analysis for Human-Computer Interaction. Edited by Prof. Dan Diaper & Prof. Neville Stanton. Lawrence Erlbaum Associates., 2003.
12. Paternò, F., *Model-Based Design and Evaluation of Interactive Application*. Applied Computing Series, Springer-Verlag (eds.), 2000, ISBN: 1-85233-155-0.
13. Pribeanu C., Limbourg Q., Vanderdonck J., Task Modelling for Context-Sensitive User Interfaces; DSVIS 2001;

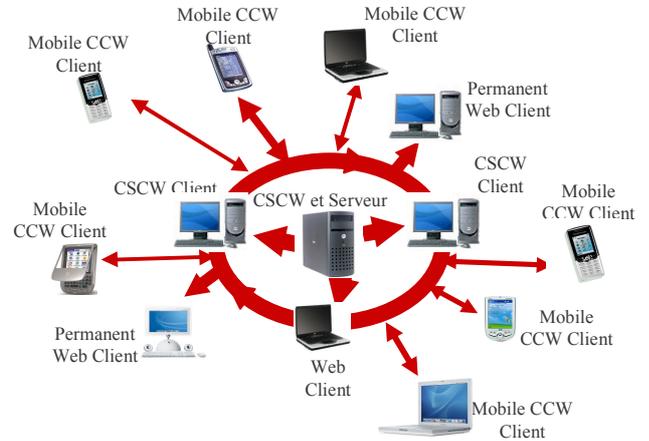
Conception et mise en œuvre de systèmes collaboratifs capillaires

Problématique du Travail collaboratif capillaire

Le TCAO « capillaire » (par analogie avec le réseau des vaisseaux sanguins) consiste à étendre les capacités fournies par les outils de travail coopératif en des ramifications de plus en plus fines, depuis leur utilisation sur des postes fixes jusqu'aux postes mobiles de différents types (PDA, Tablet PC, Téléphone mobile,...).

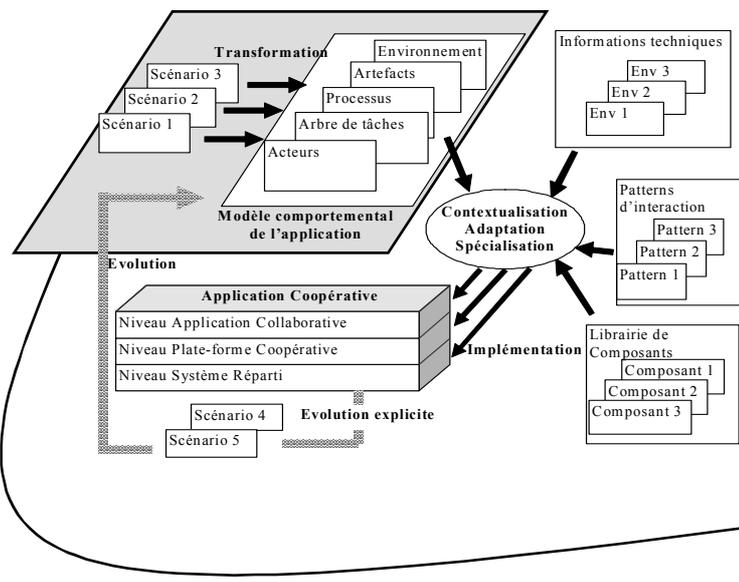
Caractéristiques:

- **Mobilité:** Permettre aux utilisateurs d'agir n'importe quand et n'importe où.
- **Coopération:** Supporter la collaboration (coproduction – coordination – conversation) entre acteurs distants.
- **Contextualisation:** Prendre en compte le contexte (lieu, temps, utilisateur et plateforme).



Framework, Concepts et Outil

FRAMEWORK

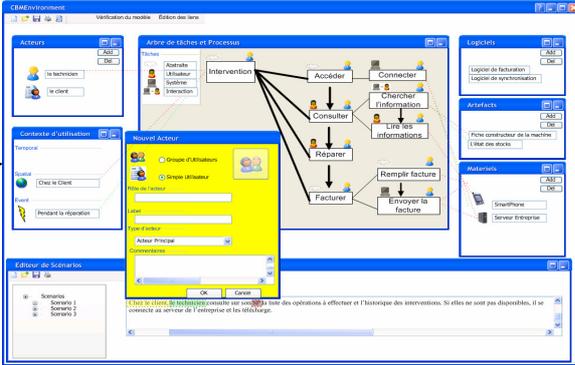


CONCEPTS

- **Scénarios:** Pour capturer les activités des utilisateurs dans des situations précises.
- **Approche « Model-Based »:** Construction d'un **Modèle Comportemental** de référence composé des modèles de tâches, des acteurs, des processus, des artefacts et des environnements.
- **MDA (Model Driven Architecture):** Transformation du modèle comportemental par réécriture pour l'adapter à l'architecture cible.
- **Patterns:** Utilisation de patterns pour l'adaptation des services et des interactions des utilisateurs aux contextes et aux dispositifs.
- **Évolution:** Prise en compte de nouveaux scénarios au cours de l'utilisation, avec modification du modèle comportemental et production de la mise à jour de l'application.

OUTIL

Contextualized Behavior Model Environment v1.0
Un outil pour l'élaboration et la Validation du modèle Comportemental.



Domaines d'application

- Systèmes de travail collaboratif des entreprises disposant de collaborateurs mobiles.
- Enseignement à distance, mobilité au sein des Campus Universitaires
- Bureaux à distance et mobiles

Modélisation d'un support de communication pour des personnes âgées à domicile

Abir Ghorayeb¹, Vincent Rialle, Norbert Nourry
Laboratoire TIMC-IMAG CNRS UMR 5525
Université Joseph Fourier, Grenoble, France
(Abir.Ghorayeb, Vincent.Rialle)@imag.fr
Norbert.Nourry@imag.fr

Joelle Coutaz
Laboratoire CLIPS-IMAG, équipe IIHM
Université Joseph Fourier, Grenoble, France
Joelle.Coutaz@imag.fr

RESUME

Cet article présente le concept de notre travail qui a pour but de proposer et de réaliser un support technique à la communication interpersonnelle dans les cas des personnes âgées à domicile, afin de réduire leur solitude et leur isolement. La notion de contexte conscient est définie non seulement pour faciliter le maintien à domicile et la relation de ces personnes avec l'extérieur, mais aussi pour assurer la protection de leur vie privée.

MOTS CLES : TCAO, télésurveillance, communication, contexte conscient, protection de la vie privée, scénario, médiaspace.

ABSTRACT

This paper presents the concept of our work that has for aim to propose and to realize a technical support to the interpersonal communication in the case of the frail people at home, in order to reduce their solitude and individually. The notion of context aware is designed to ease the autonomous life of these persons and enhance their relation with the outside. It aims also to assure them of the protection of private life.

CATEGORIES AND SUBJECT DESCRIPTORS: System Communication, context aware.

GENERAL TERMS: design, Human Factors

KEYWORDS: frailty, CSCW, telemonitoring, context aware, communication, privacy, scenario, mediaspace.

INTRODUCTION

Le sentiment de solitude est sans nul doute est le phénomène psychologique le plus fréquemment vécu par la personne humaine aux différents âges de la vie.

Le but de notre travail est de concevoir et réaliser un 'médiaspace conscient' pour aider les personnes en perte d'autonomie à vivre le plus possible chez elles d'une façon autonome, indépendante et sécurisée.

Antoinette Mayrat définit la solitude de deux façons: la solitude objective et la solitude subjective [4].

La solitude objective est un fait observable ; c'est l'isolement en tant que privation de compagnie humaine, et mise hors du circuit social. Cette solitude est parfois choisie par la personne, qui désire par exemple laisser son logement et se retirer dans une maison d'accueil, mais elle est subie si la personne est forcée de se retirer dans un endroit qu'elle n'a pas choisi.

La solitude subjective est un phénomène du vécu qui échappe à l'observation et au contrôle. Elle est de l'ordre du sensible. C'est un état d'âme ressenti sur un mode émotionnel. Ce sentiment peut être douloureux et angoissant pour la personne qui l'éprouve.

Au fur et à mesure que la personne vieillit son état d'isolement s'accroît. Aujourd'hui, plus de 25% de la population européenne est âgée de 60 ans ou plus et ce chiffre va considérablement augmenter dans les années à venir. D'ici 20 ans, le tiers environ de la population européenne sera âgé de plus de 60 ans, soit 100 millions d'habitants [1]. Une directrice de foyer/logement pour les personnes âgées au centre ville de Grenoble, lors d'un interview, nous dit: « On a eu ici une petite histoire : une jeune retraité de 73 ans était très demandeuse de présence, elle sollicitait surtout le soir, le soir était rongée d'angoisse et du coup à chaque fois on était obligé de la faire hospitaliser en psychiatrie parce que son angoisse se manifestait parfois par de comportements inadaptés, et de l'agressivité ou elle allait discuter tard le soir avec des personnes âgées qui avaient besoin de se reposer, alors ça coûte très cher à la collectivité, on lui donne beaucoup de médicament le soir alors que souvent c'était un besoin de communiquer, de pouvoir parler, de pouvoir désangoisser et l'hôpital a refusé, ils vont la mettre dans une structure très lourde où elle sera enfermée. Parfois la communication à distance et la possibilité de voir (parce qu'elle téléphonait aux infirmières de services de psychiatrie mais elle ne les voit pas) peut aider, elle avait besoin de voir quelqu'un, de sentir que quelqu'un est proche d'elle ». Cette petite histoire illustre clairement le problème de solitude chez certaines personnes âgées. D'où l'importance des moyens apportés par les nouvelles

¹ Boursière du CNRS libanais

technologies et en particulier par l'émergence des nouvelles technologies mobiles de communication. Le but du projet présenté par ce papier est la réalisation d'un outil de travail coopératif associant la communication vidéo et la protection de la vie privée pour faciliter le contact informel et permanent entre les différents acteurs du système. Il s'agit d'un outil de communication personne à personne mettant en œuvre une série de technologies avancées dans le domaine de la communication : le médiaspace. En favorisant les échanges informels, cet outil peut renforcer le sentiment d'appartenance à une famille, qu'elle soit privée ou sociale. Il intègre également les capacités de télésurveillance médicale, d'aide à domicile et de facilitation de la mobilité des personnes dépendantes.

PROBLEMATIQUE

Dans la littérature plusieurs habitats intelligents ont été construits à des fins divers Fig.1 (rendre des services, contrôler, aider au travail collectif etc). Certains étaient des maquettes académiques (exemples : le « Georgia Tech Aware Home », le « MIT House-n », « l'UC Boulder Adaptive House »), d'autres sont commerciaux (exemples : le « General Electric Smart Home » et le « Microsoft Easy Living »). Plusieurs outils de télésurveillance et de communication ont été développés, tels que la montre Serviligne ou Vivago, VideoProbe au sein du projet interliving et beaucoup d'autres mais jusqu'à maintenant nous ne trouvons pas un moyen réel de communication pour des personnes âgées, qui sont dans la plupart fragiles (déficits sensoriels, auditifs...)

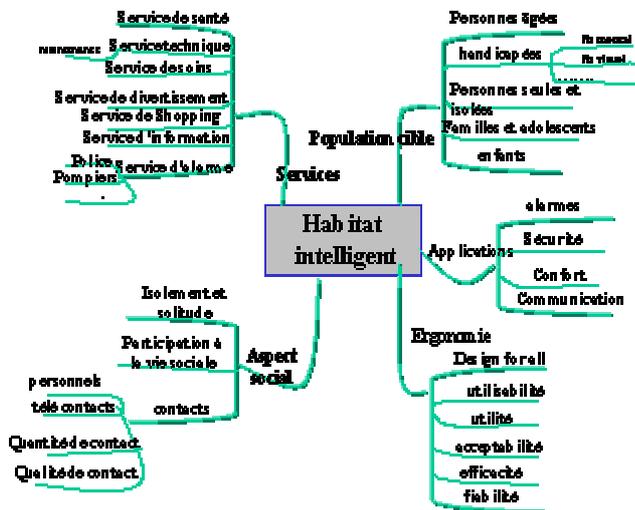


Figure.1 : l'habitat intelligent dans la littérature

Ce que nous proposons c'est le concept de 'médiaspace conscient' qui ajoute à celui de l'HIS (Habitat Intelligent pour la Santé) un certain nombre de facultés offertes par le domaine de l'informatique omniprésente (*Ubiquitous Computing*) et celui des outils de TCAO (travail

coopératif assisté par ordinateur) pour participer à la coordination des soins et faciliter la coopération et la communication entre les différents acteurs mobiles du système Fig.2. Il vise également beaucoup les aspects relationnels, l'amélioration et le respect de la vie privée de la personne confinée à son domicile.

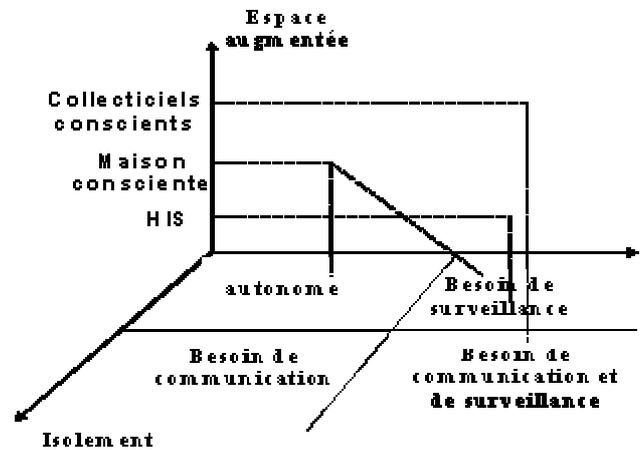


Figure 2 : Espace problème

L'HABITAT INTELLIGENT POUR LA SANTE DE GRENOBLE

Afin de pouvoir réaliser notre système et l'évaluer, nous nous appuyons sur le projet AILISA. Notre travail prolonge une première étape constitué par le projet AILISA (RNTS 2004-2006, resp. N. Noury, Labo. TIMC-IMAG) : « Appartements adaptés à une longévité effective ». Cette première étape a pour but de mettre en place, dans des environnements contrôlés, des plateformes pour l'évaluation médicale, technique et éthique, de technologies pour le maintien à domicile de certaines personnes âgées dépendantes. Il se propose de créer et de pérenniser des lieux de validation qui permettront d'accumuler l'expérience et d'augmenter la connaissance en toute sécurité. Il adopte une attitude nouvelle en prenant en compte dès le début la dimension éthique de la prise en charge de la santé par des moyens technologiques. Partenaires : Hôpital Charles Foix (Paris), Hôpitaux La Grave et Casselardit (Toulouse), CCAS de Grenoble, LRP-Laboratoire de Robotique de Paris, LIP6-Laboratoire d'Informatique de Paris 6, LEM-Laboratoire d'Ethique Médicale, Laboratoire de Physiologie du Mouvement (U483 INSERM), Hôpital Paul Brousse (Villejuif).

L'équipe AFIRM du laboratoire TIMC-IMAG a développé un projet intitulé Système d'Information et de Communication de l'Habitat Intelligent pour la Santé (SIC-HIC). L'objectif est de surveiller le patient, chez lui, à distance, en déclenchant des alarmes dans les centres d'urgence adéquats en cas de malaises, de chutes ou de situations anormales, ceci au moyen de capteurs installés à son domicile.

Un local a été entièrement équipé à la Faculté de Médecine de Grenoble pour en faire un véritable

« Habitat Intelligent pour la Santé » pilote à des fins d'expérimentation et de simulation.

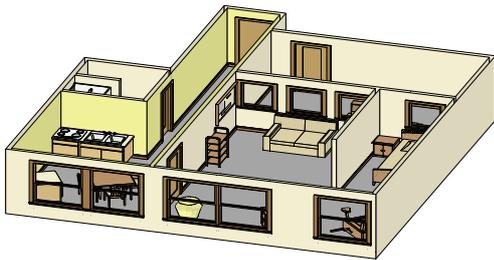


Figure 3 : plan de l'HIS de Grenoble

Cette réalisation constitue un prototype d'appartement d'environ 50 m² Fig. 3, comprenant les zones d'habitat classiques (chambre, séjour, cuisine, toilettes, douche et couloir) couvertes par des détecteurs de mouvement infrarouges qui détectent les déplacements de la personne. D'autres capteurs permettent de connaître les paramètres vitaux du résident, comme son poids, sa tension, sa saturation en oxygène dans le sang, ainsi que sa position. Une zone technique attenante à l'appartement a été ajoutée afin de recevoir le système informatique d'expérimentation du projet.

Ce premier travail n'inclut pas la dimension « communication », objet de notre travail : Un support technique ouvrant de nouveaux modes de communication et de socialisation.

Approche par Scénarios

Concevoir un système médiaspace pour des personnes âgées fragiles est une tâche complexe, étant donnée la baisse des capacités cognitives et l'ignorance de la technologie par une bonne partie des personnes âgées ou handicapées. Le système doit être d'une compréhension et d'un usage très simple et aisé (concept de *design for all*). Il faut absolument minimiser l'intervention des personnes âgées dans l'utilisation du système. Pour bien préciser les besoins des utilisateurs et comprendre leur environnement, nous avons, dans une première étape, défini, à partir d'études déjà conduites dans ce domaine, les différents acteurs concernés: personnes fragiles, membres de la famille, voisins et amis, professionnels des soins médicaux et sociaux, responsables de la télésurveillance.

Les scénarios ont été proposés par Caroll comme des descriptions détaillées du contexte d'utilisation dans un langage bien compréhensible par tout le monde afin de prendre des décisions de conception [3].

En outre, en s'adressant à des personnes âgées, c'est une procédure plus délicate. Avec les personnes âgées, il faut surtout écouter et observer. Il est très difficile de leur proposer des activités telles que dessiner ou imaginer des schémas. La plupart refusent et rare sont celles qui acceptent. Il faut savoir comment s'approcher d'elles et de gagner le peu de leur confiance avant de débiter quoique ce soit. Pour réaliser notre étude, nous nous sommes adressés à un foyer-logement pour personnes

âgées du centre ville de Grenoble. Un groupe de 8 personnes âgées a ainsi été constitué dont 3 initialement inclus dans le projet AILISA.

Afin de réaliser nos scénarios, nous avons préparé deux questionnaires : l'un pour les professionnels, l'autre pour les personnes âgées. Ces questionnaires, évalués par la directrice du foyer, constituaient une base de discussion avec les différentes personnes. Toutes les interviews étaient assez longues, alors que le questionnaire n'a besoin que de 50 minutes approximativement.

La plupart des questions se déroulent autour des problèmes d'isolement, de communication et les raisons pour lesquelles la personne a quitté son domicile. Ces rencontres étaient assez riches et montrent que la plupart des personnes âgées que l'on a rencontrées ont déménagé pour raison de solitude.

Quatre personnes âgées sur 8 ont dit : « je ne vis pas seule, ils sont tous là », en nous montrant par la main les souvenirs qu'elles ont et les tableaux de photos de leurs enfants, petits-enfants et arrière-petits-enfants Fig 4.



Figure 4 : les personnes âgées vivent les souvenirs

7 sur 8 personnes âgées étaient motivées par l'idée de notre système, dont deux sont incluses dans le projet AILISA. Ce sont elles qui ont nommé 'Juliette' notre actrice principale dans les scénarios, en hommage à une personne récemment décédée.

L'utilisation de l'approche par scénarios a engagé la participation des utilisateurs finaux, ce qui a plu aux personnes âgées : « le fait de participer, ça nous réprecise les choses, et ça nous fait réfléchir, c'est pas négligeable » nous a déclaré l'une d'elles.

La conception est évaluée au fur et à mesure de l'évolution du travail et l'étude doit durer le long de la réalisation pour éviter d'avoir des surprises à la dernière minute ; surtout qu'avec les personnes âgées, on découvre des choses que parfois elles ne peuvent pas exprimer aisément.

En étudiant la vie des personnes âgées, nous déduisons que l'intégration de la technologie doit être faite d'une façon invisible. D'où l'idée du système. Ça sera un tableau portable, léger, mobile et anti-choc, avec une interface multimodale pour répondre aux déficiences sensorielles des personnes âgées. Pour exemple, lorsque la personne âgée approche sa main de

la zone consacrée à un acteur du système sur l'interface tactile, elle entendra son nom directement. En cas d'absence ou en cas de détection par les capteurs d'une situation non souhaitable, les acteurs seront remplacés par des objets à l'écran qui les représentent aux yeux de la personne âgée...

Mise en œuvre

La multimodalité des interfaces personnes-système (oraux, visuels, tactiles) offre une certaine variété de mode de communication et permet de répondre aux contraintes kinesthésiques et motrices de différents handicaps qui peuvent exister chez les personnes âgées/dépendantes.

Le système que nous envisageons est basé sur une tablette informatique, une caméra, et un téléphone portable au sein de l'HIS. L'HIS dispose déjà d'un ensemble de capteurs électroniques, indépendants pour la plupart, dont les informations sont centralisées au niveau d'une carte particulière.

L'application existante est facilement portable sur différents systèmes d'exploitation. Pour des raisons de programmation orientée objet exigée par la nature du projet, de portabilité de l'application sous différents systèmes d'exploitation, et de gestion de coût, le langage Java s'impose de lui-même.

La connaissance de l'activité de la personne est simulée par la technique du Magicien d'Oz.

Les différents objets informatiques sont connectés entre eux par une liaison sans fil ayant une portée de l'ordre d'une vingtaine de mètres. Cette portée limitée permet à la fois de ne pas avoir à porter en permanence le dispositif sur soi et de le laisser par exemple dans son sac, mais également de ne pas partager ses données avec tout le quartier.

La tablette est l'objet qui simule le tableau interactif dont se servent les utilisateurs pour interagir avec le système. Elle leur permet notamment de choisir la personne avec qui ils souhaitent communiquer.

Ethique

La gestion de la protection de vie privée ne consiste pas à fabriquer des règles et de les imposer. Elle consiste plutôt en la gestion continue des limites entre les diverses sphères d'action et les degrés de divulgation d'informations dans ces sphères. Il s'agit d'une recherche des points d'équilibre entre des buts contradictoires[4]. Posséder à tout instant des informations sur la personne dépendante introduit un dilemme entre d'une part la prise de conscience rapide d'éventuels problèmes de la part de personnes distantes, et d'autre part les risques de rupture de protection de la vie privée.

Il est important que les utilisateurs soient maîtres de leur espace privée fin d'accepter la présence éventuelle de caméras et de capteurs pour la télésurveillance médicale. « La télé réalité c'est la mort du relationnel, la famille n'aurait plus besoin d'aller les voir. » , « je suis pour

mais en même temps je suis contre parce que finalement ils peuvent me remplacer par un robot, moi j'estime que ma fonction c'est pas seulement répondre purement au besoin technique mais il y a aussi répondre à un besoin de contact, j'ai pas choisi ce métier pour être coupé du contact physique, de contact émotionnel. Sur le principe c'est fantastique », ce sont quelques réflexions du personnel du foyer qui ouvrent la porte sur de nombreuses questions qu'il faut traiter avec délicatesse.

CONCLUSION

En France, 2,6 millions de seniors âgés de plus de 50 ans utilisent Internet, soit 16,2% participent à des chats, 26,9% se servent de la messagerie instantanée, 42,1% pratiquent de téléchargement et 79,2% consultent les sites marchands[5]

Ces jeunes seniors seront les seniors du futur, et ... Après notre rencontre avec ces personnes âgées, nous avons cédé à l'idée d'un système bien très étendu pour réaliser un système très simple, compréhensible par les personnes âgées. Ce qui nous a encouragé le plus, c'est que la direction du foyer/logement pour les personnes âgées de Grenoble, a décidé d'installer l'année prochaine, une salle d'informatique pour les seniors. De même, une entreprise spécialisée dans l'informatique, depuis une vingtaine d'années ; basée à Rennes, a décidé de se tourner vers un nouveau marché en pleine expansion : la formation des personnes âgées à Internet[6]. Tout cela montre que l'âge n'est pas un obstacle pour l'apprentissage des nouvelles technologies.

Au fur et à mesure de notre étude, notre idée de conception a été changée selon les besoins de la personne. Une bonne observation de la vie de la population cible de notre étude nous a guidé dans la conception de notre système dont la mise en place devrait débiter prochainement.

REFERENCES

1. Camarinha-Matos, L.M. and Afsarmanesh, H. Virtual Communities and Elderly Support *Proceedings of MIV'01 in "Advances in Automation, Multimedia and Video Systems, and Modern Computer Science"*, WSES, ISBN 960-8052-44-0, pp. 279-284, Sept 2001.
2. Carroll, JM (2000) Making use. Scenario-based design of computer interactions. MIT Press, Cambridge, Massachusetts.
3. Communiqué France Télécom, Paris mai 2004, http://francetelecom.com/fr/espaces/journalistes/communiqués/CP_Old/cp040524-2.html
4. Delisle I. Réflexion sur la solitude. Virage, Volume 1 Numéro 3, printemps 1996.
5. Mynatt, E. D., J. Rowan, A. Jacobs et S. Craighill. *Digital Family Portraits: Supporting Peace of Mind for Extended Family Members*. Proceedings of CHI 2001: ACM Conference on Human Factors in Computing Systems: 333-240
6. http://www.senioractu.com/index.php?action=article&id_article=61678&id_rubrique=7850

Variations pour une nouvelle méthode de saisie de données : Glyph

Gurvan Uguen

VALORIA – Université de Bretagne-Sud
Campus de Tohannic
56000 Vannes, France
gurvan.uguen1@etud.univ-ubs.fr

Franck Poirier

VALORIA – Université de Bretagne-Sud
Campus de Tohannic
56000 Vannes, France
franck.poirier@univ-ubs.fr

RÉSUMÉ

Cet article présente différentes propositions d'interfaces permettant la mise en œuvre d'une nouvelle méthode de saisie de données : Glyph. Cette méthode s'appuie sur un jeu restreint de sept primitives permettant une décomposition des lettres et symboles à saisir en séquences d'au maximum trois primitives.

MOTS CLÉS : saisie de données, clavier physique, clavier virtuel, interface tangible.

ABSTRACT

This paper briefly presents various proposals for interfaces allowing the deployment of a new data entry method: Glyph. This method is based on a restricted set of seven primitives to decompose letters and symbols in sequences with a maximum length of three primitives.

CATEGORIES AND SUBJECT DESCRIPTORS : H.5.2 [Information Interfaces and Presentation]: User Interfaces - *Input devices and strategies, Interaction styles, Prototyping*; D.2.2 [Software Engineering]: Design Tools and Techniques – User Interfaces.

GENERAL TERMS : Design, Human Factors.

KEYWORDS : data input, hard keyboard, soft keyboard, tangible interface.

INTRODUCTION

De nombreuses méthodes de saisie de données ont été mises en œuvre ces dernières années. Les principaux domaines ayant motivé ces développements sont la mobilité et le handicap. Pour un état de l'art en français concernant les dispositifs nomades on pourra consulter [4]. Nous allons ici décrire brièvement une nouvelle mé-

thode de saisie de données avant de s'intéresser à des possibilités d'adaptation des principes de la méthode à différents types de dispositifs et interfaces.

PRÉSENTATION DE LA MÉTHODE

Nous verrons que selon les aspects considérés, dans sa version clavier virtuel dynamique Glyph peut être comparée à diverses méthodes : T-Cube [6] pour la forme, Symbol Creator [2] pour la composition de séquences de primitives ou Unistrokes [1] pour le tracé de ces séquences. Pour permettre une saisie de données complète, cette méthode s'appuie sur un jeu de sept primitives.

Le jeu de primitives pour la saisie de caractères

Afin de décomposer les caractères, Glyph s'appuie sur l'ensemble de six primitives donné dans le tableau 1.

Primitive	Symbole
Incurvée droite (ID)	
Incurvée gauche (IG)	
Incurvée haut (IH)	
Incurvée bas (IB)	
Horizontale (H)	
Verticale (V)	

Tableau 1 : Le jeu de primitives de saisie de caractères

La décomposition des caractères a été effectuée en cherchant à maximiser l'analogie avec les tracés réels des lettres et en fixant la longueur maximale des séquences à trois primitives. Comme on le voit sur la figure 1, la recherche d'analogie a été faite au niveau de la forme tout d'abord, et au niveau temporel ensuite.

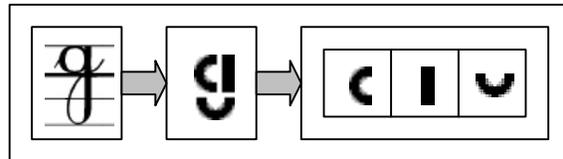


Figure 1 : Obtention de la séquence pour la lettre « g »

On obtient ainsi un ensemble de séquences de primitives définissant les caractères de l'alphabet par défaut, un al-

phabet latin minuscule augmenté, de 43 caractères. A ces séquences s'ajoutent une douzaine de séquences redondantes comme dans l'exemple de la figure 2.

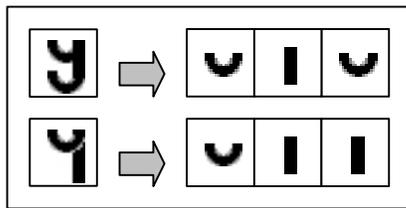


Figure 2 : La séquence de la lettre « y » et une variante

La primitive de segmentation

Afin de permettre une saisie de données autres que des caractères, une septième primitive est ajoutée, que nous appellerons « primitive de segmentation » (Segm). Outre la saisie du caractère d'espace, elle sert à définir des séquences donnant accès à six modes permettant :

- de saisir des caractères accentués ;
- de ponctuer le texte ;
- de choisir un des alphabets supplémentaires, dont un d'abréviations et la capitalisation ;
- d'effacer un caractère, un mot ou une sélection, et d'annuler ou refaire les dernières séquences ;
- d'accéder au presse-papier du système ;
- de naviguer dans le texte et d'effectuer une sélection de texte (ou d'accéder à un mode numérique bloquant pour la saisie de longues suites de chiffres, selon le type d'interface).

DÉCLINAISONS DE L'INTERFACE

Les principes initiaux de la méthode peuvent être envisagés sur différents dispositifs et selon différentes approches. Trois types d'interfaces vont être abordés : les claviers physiques ou virtuels et les interfaces tangibles.

Saisie sur clavier physique

Différents types de claviers physiques sont possibles. On peut chercher soit à adapter la méthode à des dispositifs déjà existants soit à proposer des dispositifs spécifiquement conçus pour une utilisation de Glyph.

Nous avons vu que la méthode s'appuie sur l'utilisation d'un jeu de sept primitives. Il est cependant possible d'adapter la méthode à des dispositifs à moins de sept touches. On peut envisager des versions à cinq, quatre ou trois touches : avec une touche de segmentation, une de défilement et de une à trois touches ambiguës (par exemple {IG/ID}, {IH/IB} et {H/V} pour une version à cinq touches). On peut aussi supprimer la touche de défilement et utiliser un temporisateur pour la validation selon le principe du multitap. On peut notamment ainsi obtenir une version à deux touches. Enfin une version à une unique touche (ou une molette) aura une touche de défilement entre les sept primitives et un temporisateur de va-

lidation. Notons que l'utilisation d'un temporisateur dans ces différentes versions peut être envisagée de deux façons. Soit le défilement s'effectue par n pressions et la validation se fait par temporisation (principe du multitap avec temporisateur), soit la validation s'effectue par un clic après avoir attendu le défilement de n unités de temps (comme Sybille [5]).

L'application de la méthode aux claviers standards à douze touches des téléphones portables permet d'envisager différentes répartitions des primitives sur le clavier dont celles de la figure 3. Les images de droite et du centre sont une même répartition permettant à des claviers à touches hexagonales comme celui du Philips 355 (à droite) de conserver la disposition des primitives du clavier virtuel dynamique présenté par la suite.



Figure 3 : Deux variantes pour téléphones portables (les images du centre et de droite sont une même variante)

On pourra utiliser une des touches « * » ou « # » (par exemple une pression longue sur la touche) ou une touche supplémentaire spécifique (les téléphones portables actuels disposant généralement de plus de douze touches), ceci afin de basculer facilement entre la saisie numérique standard et la saisie avec Glyph.

Il est possible d'envisager des claviers spécifiques. La figure 4 montre un clavier pour une saisie à deux mains, les six touches associées aux primitives de saisie de caractère venant se positionner sous l'index, le majeur et l'annulaire de chaque main. Afin d'optimiser la saisie de la primitive la plus fréquente, la primitive de segmentation est associée à deux touches sous les pouces.

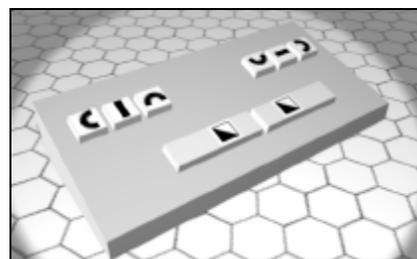


Figure 4 : Dispositif spécifique pour une saisie à deux mains

Pour une saisie à une main on pourra utiliser un clavier physique reprenant la forme du clavier virtuel dynamique (figure 7) ou, comme dans la figure 5, utiliser un dispositif sur le même principe que Twiddler [3] mais avec moins de touches. Les six primitives de saisie de caractères sont positionnées sur une face de manière à pouvoir

être saisies par trois doigts. La primitive de segmentation vient se placer sous le pouce (selon la forme du dispositif, sur le haut ou sur une autre face). Comme on ne peut envisager de saisie par accord, certaines séquences comportant plusieurs fois la même primitive, la saisie se fait en séquence. Toutes les séquences sont de trois primitives, celles normalement plus courtes étant complétées par des primitives de segmentation. Ainsi la séquence de la lettre « m », initialement (IH, IH), devient (IH, IH, Segm).



Figure 5 : Dispositif spécifique pour une saisie à une main

Saisie sur clavier virtuel

On peut bien évidemment implémenter un clavier virtuel correspondant à un des claviers physiques venant d'être décrits. Cependant il est aussi possible de chercher à adapter la méthode aux spécificités de son utilisation sur des dispositifs à écran tactile de type PDA ou Tablet PC. Il s'agit donc ici de tirer partie des avantages apportés par la dynamique de l'affichage sur de tels dispositifs tout en tenant compte de leurs contraintes.

Le premier type de clavier ayant été envisagé était le clavier statique de la figure 6. Ce clavier pouvait être utilisé pour une saisie par tracé continu ou par tracé semi-continu (continu sur une séquence).

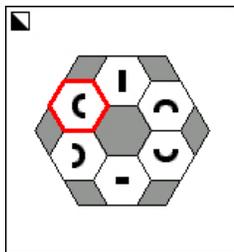


Figure 6 : Interface pour une saisie continue

Toutes les séquences doivent être de trois primitives ou complétées par une primitive de segmentation. Les six primitives de saisie de caractères sont disposées dans des hexagones. La primitive de segmentation est positionnée à l'extérieur de l'hexagone englobant. Les zones grisées, dont l'hexagone central, sont des zones permettant la saisie consécutive de la même primitive. Ceci permet d'accéder aux sept primitives depuis toute autre primitive et garantit ainsi la saisie continue.

Dans le cas d'une utilisation sur un dispositif mobile de type PDA, comme tout clavier virtuel statique il occupe une zone importante. De plus l'utilisation de la zone rectangulaire englobante comme une zone de détection nécessite un espace important afin d'éviter les erreurs de saisie dues aux déplacements parfois non linéaires dans cette zone (pour aller saisir la primitive suivante).

Privilégiant une saisie semi-continue sur une saisie continue, la primitive de segmentation a été repositionnée au centre afin de réduire les dimensions du clavier, la zone rectangulaire englobante n'étant plus utilisée comme zone de saisie mais comme zone de feedback.



Figure 7 : le clavier dynamique utilisé

Le clavier, présenté dans la figure 7, est dit dynamique pour deux raisons. Tout d'abord, le clavier propose différents affichages d'informations en cours de saisie d'une séquence : les primitives déjà saisies sont affichées au niveau du curseur d'insertion (figure 8), les primitives ne correspondant à aucune séquence définie dans l'alphabet en cours d'utilisation ne sont pas affichées sur le clavier et les primitives correspondant à la fin d'une séquence sont remplacées par le caractère correspondant. De la même manière un guidage est utilisé dans les modes en utilisant des icônes des fonctions.



Figure 8 : Primitives déjà saisies pour la séquence en cours

La deuxième justification de l'aspect dynamique du clavier est que durant la saisie d'une séquence, à chaque primitive saisie le clavier est recentré au point de contact du stylet sur l'écran. Le clavier n'occupe donc plus une zone spécifique fixe à l'écran, il s'affiche au dessus d'une partie de l'écran et peut être repositionné ou redimensionné en un petit rectangle si l'utilisateur souhaite voir l'ensemble du texte à l'écran. La saisie d'une séquence d'une seule primitive s'effectue en une simple pression sur une des sept zones. Une séquence de deux ou trois primitives correspond au tracé continu d'un ou deux segments partants du centre jusqu'à une des six primitives de saisie de caractère. Les segments peuvent ne comporter qu'une longueur minimale sans longueur maximale en ajoutant un temporisateur entre les saisies des deuxième et troisième primitives.

Utilisation d'une interface tangible

Le dernier type d'interfaces envisagé est l'interface tangible. En s'inspirant de EdgeWrite [7], pour obtenir une saisie continue ou semi-continue on peut simplement disposer un cache solide au dessus du clavier de la figure 5, avec une zone circulaire évidée légèrement plus grande que le grand hexagone ; ceci permet des déplacements plus rapides dans la zone extérieure du clavier.

On peut chercher à adapter la méthode en faisant tracer les primitives par l'utilisateur, sans se soucier du sens du tracé. On retrouve une plus grande analogie avec l'écriture des caractères. La forme de l'interface tangible peut simplement être une zone circulaire évidée mais il est possible d'ajouter des zones intérieures fixes, toujours afin de limiter les erreurs de saisie (les zones fixes solides sont en noir sur la figure 9, les zones neutres en gris, les zones de détection en blanc). La primitive de segmentation correspond à un tracé du centre vers l'une des quatre zones périphériques (exemple en rouge sur la figure 9).

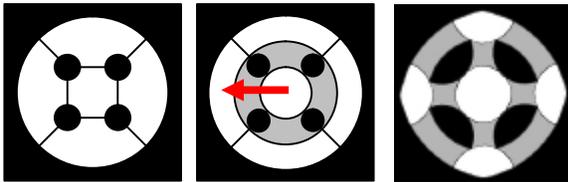


Figure 9 : Exemples d'interfaces tangibles

Pour une saisie semi-continue, entre les tracés de deux primitives l'utilisateur peut avoir à tracer des arcs correspondant à des demi-primitives incurvées afin de rejoindre la position pour tracer la primitive suivante. Ceci ne garantit pas la saisie semi-continue car quelques rares couples de tracés de séquences sont à désambigüiser. L'utilisateur peut aussi éviter ces ambiguïtés en traçant tout arc de liaison dans une direction différente de celle de la primitive venant d'être saisie.

La dernière interface de la figure 9 est une variante dans laquelle les courbes ont été travaillées (arcs de cercles et butées) afin de maximiser le guidage du tracé et proposer une solution plus adaptée pour des personnes handicapées ; il s'agit ici plus d'une interface physique pour une utilisation sur un dispositif de saisie spécifique de plus grande taille que la zone d'écriture d'un PDA.

PERSPECTIVES

Un problème des méthodes utilisant la prédiction de mot est le nombre limité de mots proposables sans décourager l'utilisateur de parcourir la liste et sans diminuer sa vitesse de saisie. Le clavier virtuel dynamique autorise l'affichage de propositions près des primitives corres-

pondant à la première primitive du caractère suivant ; ceci permet d'augmenter le nombre total de propositions tout en limitant la liste à consulter par l'utilisateur.

Un problème des claviers virtuels est la grande difficulté d'effectuer une saisie en aveugle, en raison de la taille limitée des touches (notons qu'un tracé dans une zone de détection d'une méthode de reconnaissance d'écriture nécessite aussi de tenir compte du tracé global pour ne pas sortir de la zone). Il s'agira donc de chercher à dépasser cette contrainte de dimensions.

CONCLUSION

Glyph est une méthode d'édition de données complète proposant une logique simple d'analogie pour la saisie. Cette simplicité pour trouver ou retrouver une séquence et la capacité à s'adapter sur différents types de dispositifs font de Glyph une méthode de saisie de données adaptée non pas seulement à un type d'utilisateur visant un niveau expert sur un unique type de dispositif spécifique mais aussi à des utilisateurs occasionnels ou réguliers d'un ensemble de dispositifs mobiles.

BIBLIOGRAPHIE

1. Goldberd D., Richardson C., *Touch typing with a stylus*. Proceedings INTERCHI'93, 80-87, 1993.
2. Luoma M., *Symbol Creator: Usability Evaluation of the Novel Pen-Based Text Input Technique*. New Interaction Techniques 2003, université de Tampere, 2003.
3. Lyons K., Starner T., Plaisted D., Fusia J., Lyons A., Drew A., Looney E., *Twiddler typing: One-handed chording text entry for mobile phones*. Proc. of the SIGCHI conference on Human Factors in Computing Systems. ACM Press, 328-335, 2004.
4. Poirier F., Schadle I., *Etat de l'art des méthodes de saisie de données sur dispositifs nomades. Une typologie des approches*. IHM'04, Namur, 2004.
5. Schadle I., Antoine J.Y., Le Pévédic B., Poirier F., *SibyLettre : Prédiction de lettre pour la communication assistée*. RIHM, vol. 4 - n° 1, 2003.
6. Venolia D., Neiberg F., *T-Cube: a fast, selfdisclosing pen-based alphabet*. Proc. on Human Factors in Computing System: celebrating interdependence, ACM Press, 265-270, 1994.
7. Wobbrock J. O., Myers B. A., Kembel J. A., *EdgeWrite : A Stylus-Based Text Entry Method Designed for High Accuracy and Stability of Motion*, ACM Symposium on User Interface Software and Technology, UIST'03, 2003.

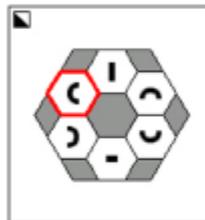
Variations pour une nouvelle méthode de saisie de données - Glyph -



Gurvan Uguen, Franck Poirier,
VALORIA - Université de Bretagne-Sud
prenom.nom@univ-ubs.fr



Clavier virtuel statique



Permet une saisie continue ou semi-continue (continue sur une séquence)

Clavier virtuel dynamique

Saisie semi-continue
Clavier affiché au dessus le document et se déplaçant durant la saisie
Affichage dynamique d'informations sur le clavier pour guider l'utilisateur durant une saisie



Adaptation aux claviers physiques existants ex : téléphone portable



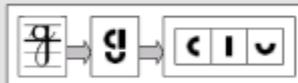
Possibilité d'utiliser une huitième touche pour basculer entre une saisie standard de chiffre et la saisie avec Glyph

Glyph 1/2 Les primitives de saisie de caractères

Décomposition des caractères en séquences d'au maximum trois primitives grâce à un jeu de six primitives

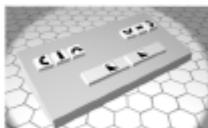
Primitive	Symbole
Incurvée droite	⤴
Incurvée gauche	⤵
Incurvée haut	⤶
Incurvée bas	⤷
Horizontale	—
Verticale	

Analogies de forme et temporelle



Claviers physiques spécifiques

Saisie à deux mains



Saisie à une main



Interfaces tangibles

• Utilisation du clavier statique

• Tracé des primitives par l'utilisateur



guidage physiques en noir, zones neutres en gris, zones de détection en blanc



Glyph 2/2 La primitive de segmentation

Une septième primitive définit six modes pour :

- accentuer,
- ponctuer,
- choisir un autre alphabet, saisir des abréviations, capitaliser,
- effacer un caractère, un mot ou une sélection, annuler ou refaire les dernières séquences,
- accéder au presse-papier,
- naviguer dans le document ou sélectionner du texte

• Guidage maximal du tracé



Architectures Logicielles Appliquées à l'Interaction Personne-Système

Jérôme Lard

THALES Recherche et Technologie
Domaine de Corbeville
91404, Orsay, France
jerome.lard@thalesgroup.com

Célestin Sedogbo

THALES Recherche et Technologie
Domaine de Corbeville
91404, Orsay, France
celestin.sedogbo@thalesgroup.com

RESUME

Alors que les moyens d'interaction deviennent de plus en plus complexes et sophistiqués, l'interaction en tant que telle demande l'intégration de modalités hétérogènes telles que la voix, le geste ainsi que de nouvelles représentations graphiques. En parallèle, le nombre de terminaux va grandissant (PC, PDA, tablet PC, etc.). Une des solutions qui permet de fournir à l'utilisateur un accès plus large à des systèmes existants et qui peut augmenter le confort d'utilisation des interfaces hommes-machines, est de concevoir des systèmes interactifs intelligents qui s'adaptent dynamiquement aux différents contextes d'utilisation. Nous illustrerons ici le concept de HIC (*Human-system Interaction Container*) qui introduit un changement important dans les modes de conception des systèmes interactifs.

MOTS CLES : interaction peronne-système, architecture d'interaction, modèles de conception logicielle, intergiciel.

ABSTRACT

As today's means of interaction become more and more various and sophisticated, interaction demands the integration of heterogeneous modalities, such as voice, gesture, graphics and animation, as well as appliances, such as traditional laptop and desktop workstations, mobile phones, Personal Digital Assistants (PDA), tablet PC, etc. One solution for providing users with wider access to existing systems and for enhancing user-friendliness of existing interfaces is to design intelligent interaction systems that dynamically adapt to the interaction environment and react appropriately in various contexts of use, without implying any modification to the core application. We will illustrate here the concept of Human-computer Interaction Container (HIC) which introduces an important shift in the field of Human-Computer Interaction (HCI), moving from an application-centric to a user-centric perspective, through the adoption of a service-oriented view of application and user interface capabilities.

CATEGORIES AND SUBJECT DESCRIPTORS: H.5.2 [User Interfaces]: Miscellaneous; D.2.2 [Design Tools

and Techniques]: Modules and interfaces, Object-oriented design methods, User interfaces; H.1.2 [User/Machine Systems]: Human Information Processing; I.3.6 [Methodology and Techniques]: Device Independence, Interaction Techniques, Languages.

KEYWORDS : human computer interaction, interaction architecture, software design pattern, middleware.

INTRODUCTION

Dans la plupart des cas, les interfaces logicielles existantes ne sont qu'un prisme qui permet de consulter les données d'une application et d'agir sur celles-ci au travers de commandes prédéfinies (un logiciel de traitement de texte propose une vue formatée d'un espace de données sans aide particulière pour l'utilisateur). Or, nous devons garder présent à l'esprit qu'un système d'information est un système d'aide et d'amélioration des performances.

Un cycle sans fin

Ainsi, nous constatons que les applications sont de plus en plus puissantes et les interfaces qui les pilotent sont de plus en plus complexes. En parallèle, l'homme ne peut pas démultiplier ses capacités mentales de la même manière que le nombre de fonctions disponibles s'accroît dans une application. La fuite en avant qui consiste à fournir plus de commandes au sein d'une interface pour maîtriser la complexité d'une application n'est pas une solution viable à moyen terme. D'autre part, le fait de proposer une interface graphique générique, en espérant qu'elle conviendra à un nombre croissant d'utilisateurs ne semble pas non plus être une réponse appropriée. En effet, chacun d'entre nous possède des buts bien différents et donc des usages individuels lorsqu'il s'agit d'utiliser un système informatique. Le nombre d'utilisateurs grandissant, il est probable que les modes d'utilisation vont aller en se diversifiant. Cela contredit les concepts présentés actuellement qui cherchent à simplifier l'interface homme-machine en lui donnant un cadre interactif assez pauvre (navigation Web), pour tous types d'application Ceci n'est possible que si on réduit les processus interactifs à leur minimum.

Vers des interfaces plus intelligentes

Nous souhaitons explorer une autre voie dans la conception des systèmes interactifs qui permette au contraire de mieux servir les utilisateurs dans leurs tâches en leur offrant des interfaces permettant de manipuler des concepts de plus haut niveau. Il faut pour cela que l'interface faisant le lien entre l'homme et la machine devienne plus "intelligente" et propose un vrai dialogue à l'opérateur. L'interface doit représenter de manière appropriée les concepts manipulés par un utilisateur qui n'ont sans doute aucune relation avec le modèle de l'application sous jacente.

L'homme dans la boucle système

Il est peu probable que dans des systèmes à risque élevé (tels que ceux utilisés pour le contrôle aérien) l'homme disparaisse de la boucle de traitement des informations. Il reste un acteur important des processus de décision surtout lorsque celle-ci est critique. Tenir compte de l'homme dans la boucle système est un pré requis essentiel de nombreux systèmes d'informations professionnels. Ce type de problématique est rarement prise en compte lors de la conception d'interfaces. Il devient nécessaire de proposer un traitement de l'information qui intègre l'homme comme une composante du système au moment de sa conception. Mais cela ne résout pas complètement les problèmes soulevés ici qui sont plus de complexité dans les applications et dans les interfaces et plus d'informations à traiter. Il est important que le système prenne également des initiatives face au comportement de l'utilisateur et face aux éventuels changements de contexte.. Pour cela, nous devons modifier les modèles classiques qui définissent la séparation entre les logiques de présentation et les logiques d'application. Une nouvelle infrastructure logicielle doit pouvoir intégrer ces notions.

INTERACTION HOMME-MACHINE

Afin de pouvoir définir des services d'interaction génériques, nous avons besoin d'isoler précisément les spécificités du domaine de l'interaction. Malheureusement, la définition du domaine *HCI* est très large. Nous empruntons la définition du comité ACM ici : L'Interaction Homme-Machine est "*une discipline intéressée par le design et l'implémentation de systèmes informatiques interactifs pour l'usage humain, et par l'étude des phénomènes majeurs qui les entourent*".

Partant de cette définition, nous sommes capable de

revenir aux concepts fondamentaux de l'interaction homme-machine. Nous sommes habitués à interagir avec notre environnement par manipulation directe et physique. Lorsque l'environnement devient plus complexe, nous tentons de trouver des solutions pour surmonter nos limites physiques et naturelles. La perception de la complexité d'un tel environnement et le type de contrôle que nous voulons appliquer peut être soutenu par un système informatique. Dans des situations de plus en plus nombreuses, l'ordinateur personnel devient un intermédiaire de choix dans nos relations à l'environnement. L'utilisation d'un système informatique permet même de considérer de nouvelles opportunités d'interaction et de créer son propre champ d'actions possibles: dans un monde virtuel par exemple. L'ordinateur devient aujourd'hui un outil plébiscité lorsque nous voulons communiquer nos idées ou participer à un processus créatif.

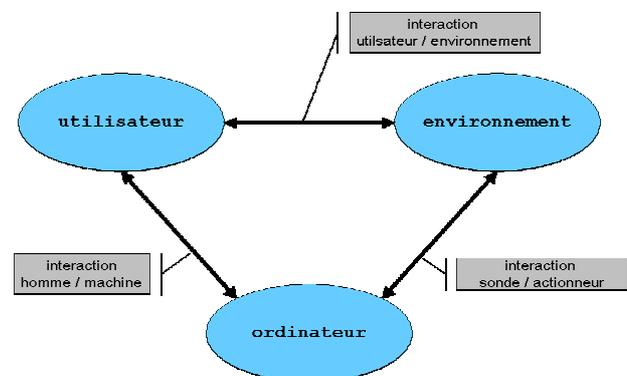


Figure 1: Principe d'action/perception

Nous retrouvons dans le schéma de la figure 1 l'action/perception directe dans ce qui constitue le lien entre utilisateur et environnement. Nous voyons que si nous introduisons un système informatique dans la communication entre ces deux entités, le couple action/perception se sépare en deux nouveaux chemins: l'interaction homme/machine et l'interaction libellée ici sonde/actionneur, signifiant que l'ordinateur perçoit l'environnement par l'intermédiaire de sondes et qu'il réagit grâce à des dispositifs mécaniques, électriques, etc., que nous avons nommés actionneurs. Le cadre de notre étude cherche donc à expliciter les principes fondamentaux qui définissent la communication entre l'utilisateur et l'ordinateur. Nous voulons ainsi en extraire les composantes les plus génériques de manière à les implémenter sous forme de services intégrés à une plateforme d'interaction. Cette plateforme couplée à un nouveau patron de conception logicielle permettra à tout concepteur d'interfaces de concevoir un système interactif, s'adaptant aux différents contextes d'utilisation et à diverses applications.

LE CONCEPT DE CONTENEUR D'INTERACTION

Ce que nous appelons conteneur d'interaction personne-système a pour but d'encapsuler tout composant logiciel dédié à la gestion ou au contrôle de l'interaction personne-système au sein d'un conteneur. Il est conçu de manière à faciliter la séparation logique entre application, interaction et présentation et manipule tous les processus interactifs permettant à une application et à ses nombreuses interfaces utilisateurs de communiquer entre elles. Ce type de découpage offre une interaction indépendante des applications d'une part et des interfaces d'autre part. Ce conteneur logiciel gère les liens entre ces composants, on parle alors de logique interne d'interaction. Celle-ci peut-être comparée pour les systèmes interactifs à la logique applicative et à la logique de présentation. D'autre part ce conteneur est sensible à la modification des contextes des utilisateurs ou du système. Dans le cas d'un changement de contexte de l'utilisateur, le conteneur d'interaction doit être capable d'en tenir compte et de répercuter cet événement au niveau de l'application cible. Dans le cas d'un changement de contexte du système, le conteneur doit être en mesure d'en informer l'utilisateur si cela est nécessaire. Si ces propriétés sont respectées, nous aurons alors réalisé un système dans lequel l'homme et la machine entretiennent un vrai dialogue. Ce conteneur agit telle une passerelle entre les services applicatifs et les services de présentation. Il implémente des services dits d'interaction qui se veulent génériques. Ils n'ont besoin que de modèles pour être configurés en accord avec un utilisateur, une tâche complexe ou encore une logique métier à respecter. Ces services d'interaction incluent par exemple le traitement du dialogue homme-machine, de la tâche et de l'activité des utilisateurs, de la gestion de la multimodalité et de la génération d'une présentation multimédia appropriée. Notre approche cherche à générer ou à ajuster de manière dynamique une présentation qui correspond le mieux aux attentes d'un utilisateur dans un contexte particulier et pour une tâche bien définie. Cette dynamique doit suivre les changements de contexte ainsi que le déroulement de la tâche d'un utilisateur. Pour autant, ce travail n'a pas pour cible la conception automatique d'interfaces homme-machine mais il s'agit plutôt d'étudier les composantes génériques du dialogue homme-machine. Ce dialogue est défini comme une gestion de la communication inter-processus entre des applications de différentes natures.

ARCHITECTURE D'INTERACTION

Nous avons considéré que la plupart des architectures logicielles implémentées à l'heure actuelle sont basées sur un paradigme en trois couches: base de données, application et présentation. Apporter plus d'intelligence à un système c'est introduire une nouvelle couche qui va se charger de la médiation entre l'utilisateur et les objets informatiques qu'il manipule. La représentation fonctionnelle de ce concept est illustré ci dessous:

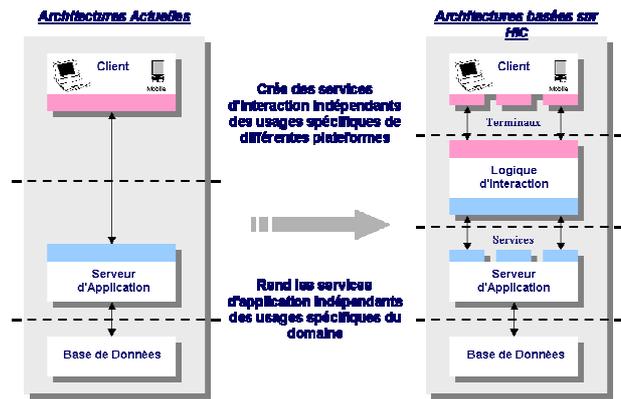


Figure 2: Architecture Fonctionnelle HIC

Nous avons choisi d'implémenter ce paradigme sous la forme d'un middleware spécifique à un domaine particulier : celui de l'interaction personne-système. Nous reviendrons plus en détail sur les spécificités de ce middleware par la suite. Nous pouvons dire que la gestion de l'interaction fournie par cette plate-forme s'applique à la *compréhension mutuelle entre des applications de natures différentes*. En conséquence notre choix est de placer cette interaction au centre des échanges entre le système et les utilisateurs. C'est la solution que nous proposons afin de répondre à la complexité croissante des systèmes et des logiciels. Un système informatique est facile à manipuler lorsqu'il offre un nombre limité de fonctionnalités. Si ces fonctionnalités sont dédiées à des tâches de haut niveau, un tel système est très puissant pour accomplir ce pour quoi il a été conçu. D'un autre côté, les systèmes qui essaient d'être le plus générique possible tendent à proposer trop de fonctions, le plus souvent d'assez bas niveau, au sens où elles sont très proches de celles du système lui-même et pas assez proches de la tâche de l'opérateur. L'architecture présentée à la figure 2 illustre l'introduction d'une nouvelle couche logicielle. Cette nouvelle couche qui contient la logique d'interaction est basée sur des modèles. Les différents modèles que nous pensons utiliser sont les suivants : modèle de tâche, profil utilisateur, modèle de l'organisation, règles métiers ou du domaine d'application.

NOTION DE MIDDLEWARE D'INTERACTION

Un middleware est selon toute vraisemblance un logiciel qui consiste en un ensemble de services permettant à plusieurs processus s'exécutant sur une ou plusieurs machines de communiquer entre eux au sein d'un réseau. Une autre définition prétend qu'un middleware est un bout de code logiciel qui agit comme une couche de conversion ou de translation entre deux processus. Cette dernière définition est plus proche de ce que nous souhaitons implémenter. En effet, en regardant la séparation fonctionnelle de la figure 2, nous nous sommes aperçu que ce que nous voulons insérer entre la logique applicative et la logique de présentation est bel et

bien une couche de conversion bien que particulière. Ce nouveau niveau doit effectuer une conversion entre la syntaxe et la sémantique d'une application et la transformer en une présentation graphique représentative de l'état du système d'une part et d'autre part est chargé d'interpréter les intentions de l'utilisateur pour l'application. Le middleware développé comporte trois niveaux : services standards, services techniques et services d'interaction. Les services standards sont ceux que l'on retrouve dans tous les middleware classiques. Les services techniques viennent supporter l'intégration de services de plus haut niveau que sont les services d'interaction. La figure 3 représente les services d'interaction de la plateforme.

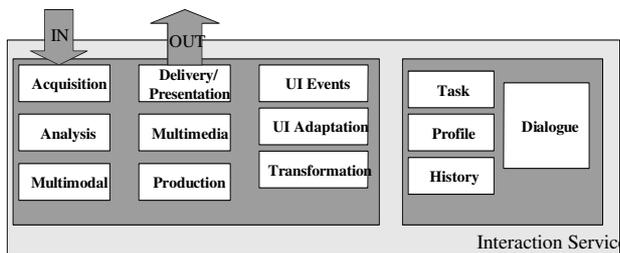


Figure 3: Couche de services d'interaction

CONCLUSION

La plateforme telle qu'elle a été présentée dans cet article a été implémentée et testée dans le contexte particulier d'une application professionnelle pour la gestion du trafic aérien. Il a été possible d'enrichir l'interaction existante avec l'application en ajoutant des fonctionnalités nouvelles sans pour autant modifier l'application de départ. Non seulement l'interaction s'est révélée plus riche mais cette expérience concrète nous a permis d'étudier la conception de différents modèles utiles à la prise en compte des contextes d'utilisation. Ainsi, nous avons défini trois modèles d'utilisateurs, un modèle de l'organisation, des modèles pour les plateformes matérielles et des règles représentatives du domaine d'activité. La plateforme d'interaction une fois implémentée se compose de trois parties distinctes : des composants d'interaction qui représentent les traitements interactifs qui ne sont pas généralisables quelque soit le contexte, le conteneur d'interaction qui assemble ces composants entre eux de manière dynamique et le middleware d'interaction qui propose des services d'interaction génériques disponibles pour toutes les applications du système. Cette architecture complète est représentée en figure 4.

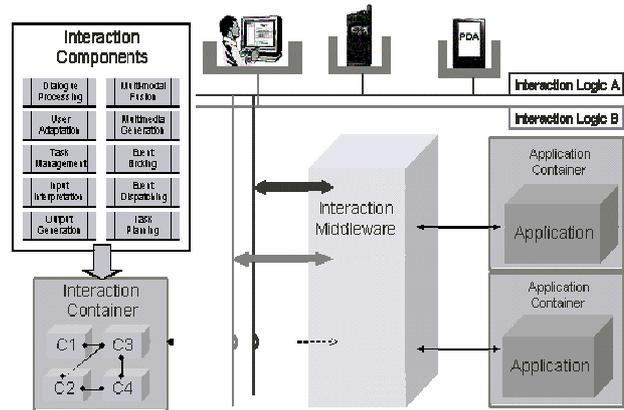


Figure 4: Architecture Fonctionnelle HIC

BIBLIOGRAPHIE

1. Brochers J., A pattern approach to interaction design.
2. Carroll J., HCI Models, Theories and Frameworks: Towards a multidisciplinary Science.
3. Geihs K., Middleware Challenges Ahead. IEEE Computer Society, 2001
4. Gulliksen J., Goransson B., Boivie I., Blomkvist S., Persson J. and Cajander A., Key Principles for User-Centred Systems Design. *Behaviour & Information Technology*, nov-déc 2003, pages 397-409
5. Kirste T., Rapp S., Architecture for Multimodal Assistance Systems. *International Status Conference, HCI*, October 2001
6. Kuniavsky M., Observing the user experience: a practitioner's guide to user research. *Morgan Kaufmann Series*.
7. Sendin M., Lorés J., Montero F. and López V., Towards a Framework to Develop Plastic User Interfaces. *Mobile HCI*, September 2003
8. Schmidt D. C., Buschmann F., Patterns, Frameworks, and Middleware: Their Synergistic Relationships. 2003
9. Thévenin D., Thèse de Doctorat de l'Université Joseph Fournier de Grenoble, "Adaptation en Interaction Homme-Machine : le cas de la plasticité", Grenoble 2001
10. Wang Z., Garlan D., Task Driven Computing. May 2000

Architectures Logicielles Appliquées à l'Interaction Personne-Système

THALES

Jérôme Lard



THALES Recherche et Technologie France
 Domaine de Corbeville
 91404, Orsay, France

LRI Laboratoire de Recherche en Informatique
 Bâtiment 490, Université Paris Sud
 91405, Orsay, France

jerome.lard@thalesgroup.com

Interfaces Intelligentes



Figure 1: l'homme et le besoin informationnel

L'interface homme-machine n'est qu'un prisme permettant à l'homme de mieux appréhender son environnement. Comment peut-on l'aider à être plus efficace dans ce processus?

Homme dans la boucle système



Figure 2: l'homme et le système ne faisant qu'un

L'homme fait partie intégrante des systèmes développés dans de nombreuses industries. Comment peut-on en tenir compte à un stade d'étude avancé?

Conteneur d'interaction



Figure 3: l'homme face à la complexité du système

L'homme dialogue avec le système informatique. Comment modéliser son activité face à celui-ci et intégrer ce modèle dans un "conteneur" informatique?

Architecture d'interaction

Nous avons considéré que la plupart des architectures logicielles implémentées à l'heure actuelle sont basées sur un paradigme en trois couches: base de données, application et présentation. Apporter plus d'intelligence à un système c'est introduire une nouvelle couche qui va se charger de la médiation entre l'utilisateur et les objets informatiques qu'il manipule. La représentation fonctionnelle de ce concept est illustrée ci dessous:

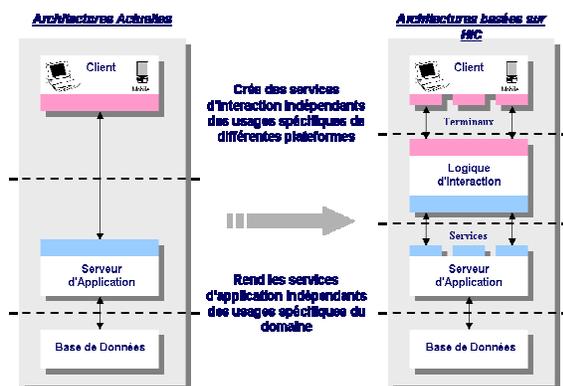


Figure 4: Architecture Fonctionnelle HIC

Nous avons choisi d'implémenter ce paradigme sous la forme d'un middleware spécifique à un domaine particulier: celui de l'interaction personne-système. Cette architecture complète est représentée en figure 4. Le conteneur d'interaction personne système encapsule la logique d'interaction pour un utilisateur et gère les appels et les interconnexions avec des composants d'interaction élémentaires.

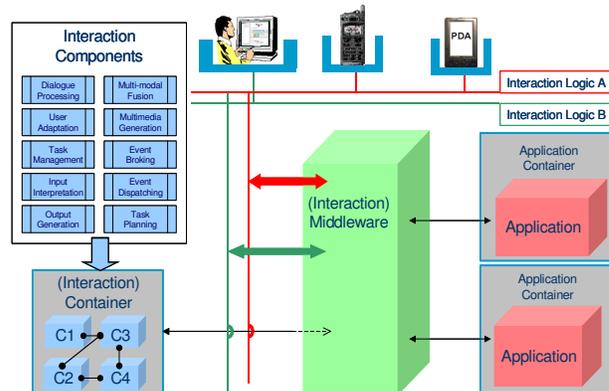


Figure 5: Architecture HIC au Runtime

Le projet SAI : Scénarios d'Animations Interactives

Nicolas Pagès, Monique Noirhomme
Institut d'Informatique – FUNDP
Rue Grandgagnage, 21
5000 Namur – Belgique
npa@info.fundp.ac.be

RESUME

Ce projet porte sur la mise en œuvre d'un système interactif prenant la forme d'un laboratoire virtuel. Il s'adresse à deux types d'utilisateurs distincts : d'une part l'auteur ou professeur, et d'autre part l'étudiant. L'objectif est de permettre à un utilisateur non informaticien la création d'expériences scientifiques, et leur mise à disposition pour les étudiants. Dans cet article, nous présentons les raisons qui ont motivé la mise sur pied de ce projet, nous détaillons le système et son implémentation, et nous présentons les deux expériences qui sont en cours de développement à l'heure actuelle.

MOTS CLES : Interactivité, laboratoire virtuel, expérience.

ABSTRACT

This project is about the development of an interactive system taking the shape of a virtual laboratory. It has two types of distinct users: first the author or professor, and second the student. The objective is to allow a non computer scientist user to create scientific experiences, and to put them at the students' disposal. In this article, we present the reasons that motivated this project, we detail the system and its implementation, and we present the two experiences that are currently being developed.

KEYWORDS: Interactivity, virtual laboratory, experience.

INTRODUCTION

La plupart des domaines scientifiques requièrent un apprentissage qui se veut à la fois théorique et pratique. Cela signifie concrètement qu'il s'agit pour ceux qui se destinent à devenir expert dans l'un de ces domaines, qu'ils devront non seulement maîtriser tous les fondements théoriques de leur science, mais qu'ils devront également mettre en

pratique et vérifier ces théories. Cela se fait généralement dans le cadre d'un laboratoire.

Cependant, l'existence de certains problèmes constitue un obstacle à un apprentissage de qualité un tel cadre. Parmi ceux-ci, on peut citer le nombre d'étudiants souvent trop élevé par rapport aux capacités des laboratoires, le coût élevé de ces laboratoires, en particulier de tout l'équipement nécessaire pour fournir un travail de qualité, etc. On peut encore évoquer le problème de la distance pour certains étudiants, par exemple ceux qui ont un horaire décalé, et qui doivent parfois parcourir de longues distances afin de se rendre aux séances de laboratoire. Enfin, on évoquera le manque de rentabilité de ces laboratoires, en termes de temps passé par l'étudiant à l'intérieur, et comparé à l'investissement nécessaire.

L'objectif du projet SAI est de développer un laboratoire virtuel qui permette de reproduire des expériences réelles et de les rendre accessibles au plus grand nombre via Internet. Cela suppose, d'une part la prise en compte de toutes les barrières imposées par un tel contexte, et d'autre part l'émergence de possibilités qui n'existent pas dans un laboratoire réel. Parmi les limitations, nous citerons la principale, à savoir l'absence de contact physique direct, avec tout ce que cela comporte comme inconvénients, et notamment l'absence de manipulation réelle des outils du laboratoire, ce qui implique donc une absence de cet apprentissage. On peut également souligner que l'objectif d'un tel laboratoire reste de simuler des expériences, et qu'il est bien entendu impossible de manière parfaite de reproduire les conditions réelles dans un environnement virtuel.

En ce qui concerne les possibilités offertes par un tel système, on se contentera de noter qu'elles sont nombreuses, et de relever celles qui à nos yeux sont les plus significatives :

- Intégration de l'expérimentation avec l'apprentissage, sans avoir à attendre les séances de laboratoire pour pouvoir faire le lien entre théorie et pratique
- Accélération des expérimentations par le biais de la simulation, permettant de faire beaucoup plus d'études paramétriques ou d'études de cas.
- Accélération des expériences et suppression des tâches répétitives, permettant de passer plus de temps à des tâches liées à l'analyse et l'interprétation des résultats.
- Accès illimité à l'expérimentation en tout temps, sauf dans le cas d'expérimentation réelle à distance.

Comme souligné dans [LVEST], un nombre important d'institutions travaillent à la mise en place de laboratoires virtuels sur le web. Il est tout d'abord à noter que les articles traitant des laboratoires virtuels ne sont pas nombreux alors que le nombre de sites de ce type sur le web augmente chaque jour. On semble donc avoir adopté dans ce domaine une approche expérimentale très ouverte, caractéristique du web, donnant lieu à une multitude de réalisations des plus diverses. La visite de nombreux sites « laboratoire virtuel » sur Internet révèle ainsi une très grande disparité tant dans les buts que dans les approches choisies par leurs concepteurs et aussi dans la qualité des résultats obtenus. Dans un but d'analyse, une typologie des laboratoires virtuels mis en place à ce jour sur le web a été établie [LVEST], se basant sur les buts visés par chacun d'eux. Si on laisse de côté un grand nombre de sites qui offrent sous ce nom des visites en image de laboratoires existants, on peut regrouper les autres sites en cinq grandes catégories, selon leur buts, explicites ou implicites :

- les sites avec **expérimentation à distance** par la télécommande d'instruments réels, avec ou sans rétroaction visuelle par canal vidéo,
- les sites basés sur la **simulation**, que ce soit en mode local, distant (programme de simulation tournant sur le serveur) ou encore au travers de fureteurs par le biais d'applets (Java),
- les sites orientés vers les **procédures** de laboratoire, dans lesquels l'apprenant est généralement amené à faire une série d'opérations dans un ordre bien déterminé afin de mener à bien son expérience, souvent supporté par des animations,
- les sites offrant du **matériel de référence** en complément au laboratoire, à utiliser tant avant qu'après les séances de laboratoire (préparation des expériences, familiarisation avec le montage et les instruments, etc., puis traitement des données, analyse, interprétation, rapport, etc.),
- les sites visant la mise en commun de ressources à des fins de recherche et favorisant le **travail collaboratif**. Ces sites sont souvent construits autour d'équipements coûteux tels que microscopes électroniques, télescopes, logiciels complexes, etc.

Les quatre premières catégories ont un but didactique alors que la dernière est généralement orientée vers la recherche. Le projet SAI s'inscrit à la fois dans la deuxième catégorie et dans la troisième, c'est-à-dire qu'il est basé sur des simulations, mais qu'il est nécessaire pour l'apprenant de réaliser certaines opérations afin de pouvoir réaliser la simulation. La pertinence de l'ordre de ces opérations est déterminée par l'auteur au cours de la conception de l'expérience.

LE PROJET SAI

Ce projet a pour objectif d'offrir à un auteur de contenu, non informaticien, un outil très simple qui lui permette la création et la simulation d'expériences scientifiques, de manière interactive. Il est conçu pour s'adresser à deux types d'utilisateurs distincts : d'une part le professeur ou auteur, et d'autre part l'étudiant ou apprenant. Pour chacun d'eux, aucune expertise particulière n'est requise en matière d'informatique. Deux interfaces différentes sont conçues dans le cadre de ce projet, une pour chaque type d'utilisateur. Chacune de ces deux interfaces offre à l'utilisateur diverses possibilités et options, qui varient donc selon qu'il s'agit d'un professeur ou d'un étudiant.

Un certain nombre d'objets sont mis à la disposition de l'auteur. C'est autour de ces objets que le projet s'articule, ils en constituent la base. L'auteur se voit donc offrir la possibilité de choisir parmi le panel d'objets mis à sa disposition ceux qui l'intéressent. Il a ainsi la possibilité de les « ajouter » dans l'expérience en cours de création. Chaque objet a bien entendu un certain comportement qui lui est

propre, et différents objets peuvent interagir entre eux. L'auteur peut alors paramétrer ces différents objets à sa guise, et peut définir les différentes actions qui seront accessibles ou pas à l'apprenant. Il lui donc possible de créer de toute pièce une expérience uniquement à partir des différents objets, en insérant ceux qu'il juge nécessaires pour la réalisation de cette expérience. Néanmoins, il aura également la possibilité de choisir parmi un certain nombre de scénarios prédéfinis. Une fois l'expérience créée, elle peut être stockée dans une base de données. L'auteur peut ensuite apporter les modifications qu'il souhaite à une expérience.

Une expérience qui a été sauvegardée dans la base de donnée devient accessible pour un étudiant. Ce dernier peut uniquement la consulter, et n'a aucune possibilité de la modifier en aucune façon. Il n'a à sa disposition que les possibilités de paramétrage que l'auteur de l'expérience lui a accordées. Il pourra donc de cette manière modifier les paramètres auxquels il a accès des objets se trouvant sur la scène. Ensuite il lui sera loisible de vérifier les résultats de l'expérience compte tenu de ces paramètres.

Dans un premier temps, le projet s'appuiera sur deux exemples particuliers en biochimie et en physique : la chromatographie échangeuse d'ions et les lois de Newton

Ces choix sont motivés par les raisons suivantes :

La chromatographie : il s'agit d'une expérience qui est à l'heure actuelle dispensée en deuxième année d'études de médecine à l'Université de Namur. Malheureusement, cette expérience est amenée à disparaître du cursus à court terme. Il s'agit d'une expérience très intéressante et importante dans le domaine de la biochimie. Par conséquent, il semble intéressant de pouvoir la maintenir accessible aux étudiants malgré tout.

Les lois de Newton : il existe déjà de nombreuses expériences sur le web dans le domaine de la physique. Néanmoins, la grande majorité de celles-ci concerne le domaine de l'électricité, et très peu, pour ne pas dire aucune, ne concernent le domaine couvert dans ce projet. Par conséquent il nous semblé intéressant de développer cette expérience de manière virtuelle, en particulier parce que les lois de Newton sont relativement abstraites, et difficiles à cerner pour un étudiant. Le fait de pouvoir visualiser les effets de forces directement devrait

leur permettre d'appréhender cette théorie plus facilement.

LA CHROMATOGRAPHIE ECHANGEUSE D'IONS

La chromatographie est une méthode analytique qui est largement utilisée pour la séparation, l'identification et le dosage des constituants chimiques dans des mélanges complexes. Il n'existe aucune autre méthode qui soit aussi puissante et d'application aussi générale. La chromatographie recouvre une grande diversité de méthodes et de techniques. Cependant, toutes les méthodes ont en commun l'utilisation simultanée d'une phase stationnaire et d'une phase mobile. La phase stationnaire est une phase qui reste en place, soit dans une colonne, soit sur une surface plane. La phase mobile se déplace sur ou à travers la phase stationnaire, en entraînant le mélange avec elle. On distingue deux grands types de méthodes chromatographiques : la chromatographie planaire, et la chromatographie sur colonne. La chromatographie échangeuse d'ions est une méthode particulière de chromatographie sur colonne, où des résines échangeuses d'ions servent de phase stationnaire. Il s'agit donc d'une méthode de séparation où des ions de même signe sont séparés par élution sur une colonne remplie d'une résine finement divisée.

Dans le cadre de cette expérience, différents objets interviennent, parmi lesquels on peut citer la colonne et la pompe.

Le plus important de ces deux objets est la colonne, car elle peut être paramétrée de manière complexe. On peut tout d'abord modifier ses dimensions, c'est-à-dire sa longueur et son diamètre. On peut également modifier le diamètre des particules qui remplissent la colonne (actuellement, les niveaux de granulométrie atteints sont de l'ordre de 5 à 10 μm de diamètre particulaire). Ces 2 paramètres sont très importants et ont une grande influence sur l'efficacité de la colonne. De plus, pour être efficace, la chromatographie requiert un équilibre judicieux entre les forces intermoléculaires qu'impliquent les 3 acteurs du processus de séparation : l'analyte, la phase mobile, et la phase stationnaire. Etant donné que dans le cadre d'une expérience à objectif didactique, la composition de l'analyte est généralement donnée, il reste à choisir les phases mobile et stationnaire. On soulignera ici l'importance pour l'apprenant d'avoir la possibilité

d'effectuer de multiples essais en un temps relativement court. En effet, dans le cadre d'un laboratoire, la réalisation complète d'une chromatographie échangeuse d'ions a durée une d'à peu près 120 minutes. Alors que dans un contexte virtuel, les résultats peuvent être obtenus immédiatement.

Le résultat fourni suite à la réalisation de l'expérience est un graphique dénommé chromatogramme, qui présente en ordonnée la concentration en soluté, et en abscisse le temps d'élution ou le volume d'élution. Généralement on obtient une série de pics symétriques, et les positions des pics sur l'axe du temps permettent d'identifier les constituants de l'échantillon tandis que les aires sous les pics mesurent leur quantité. Il est donc important d'obtenir un chromatogramme où les pics ne se chevauchent pas et ne sont pas trop larges. Il est donc nécessaire de maîtriser les paramètres qui soit augmentent la vitesse de séparation des pics, soit diminuent leur vitesse d'élargissement. Par exemple, l'un des paramètres qui influence la largeur des pics est le diamètre des particules de support. On peut également citer le temps de rétention (temps qui s'écoule entre l'injection de l'échantillon et l'apparition d'un pic de soluté sur le détecteur d'une colonne chromatographique). Le temps de rétention peut être influencé par le choix de la nature de la phase stationnaire.

LES LOIS DE NEWTON

Rappelons des lois de Newton :

- **Première loi** (ou principe d'inertie) : lorsqu'un solide est isolé ou pseudo isolé (résultante des forces nulle), son centre d'inertie G est soit au repos (G est initialement immobile), soit animé d'un mouvement rectiligne uniforme.
- **Deuxième loi** (ou théorème du centre d'inertie) : la somme des forces extérieures appliquées à un solide est égale au produit de la masse du solide par l'accélération a_G de son centre d'inertie G
- **Troisième loi** (ou principe de l'action et de la réaction) : lorsqu'un corps A exerce sur un corps B une action mécanique représentée par une force $F(A/B)$, le corps B exerce sur A une action mécanique représentée par une force $F(B/A)$. Ces deux forces ont même direction, sens contraire et même norme

- **Loi de la gravitation universelle** : deux corps quelconques s'attirent en raison directe de leur masse et en raison inverse du carré de la distance de leurs centres de gravité.

Ici, les différents objets impliqués sont à l'heure actuelle :

- une masse
- une corde
- une surface
- une traction sur la corde (symbolisée par une personne)
- une poulie
- une pression sur un objet (symbolisée par une main)

Ces différents objets sont bien entendus paramétrables, et peuvent faire l'objet de nombreuses interactions. On citera parmi les différents paramétrages possibles les suivants :

- L'objet masse a comme caractéristiques une masse particulière, un matériau particulier, et enfin une certaine gravité.
- La surface, quant à elle, a également comme paramètre le matériau qui la compose, qui a une influence directe sur le frottement de cette surface. Elle a également un certain angle.
- La corde : on distingue deux types de corde, à savoir celle qui a une masse nulle, et celle qui a une masse non nulle.

Voilà pour les principaux paramétrages possibles concernant ces différents objets. On peut maintenant citer certaines des interactions possibles entre ces différents objets :

- L'une des interactions principales est la possibilité pour la masse de glisser sur la surface, plus ou moins vite selon, sa masse, la gravité, et leur composition respective qui implique un frottement plus ou moins élevé.
- On citera également les différentes interactions impliquant des forces et se produisant entre la corde et la masse, sur la poulie, entre la personne (représentant la traction) et la masse, etc.

Scénarios d'Animations Interactives

Nicolas Pagès, Monique Noirhomme
Institut d'Informatique – FUNDP
Rue Grandgagnage, 21
5000 Namur – Belgique
npa@info.fundp.ac.be

Motivation de la recherche

Constat :

La plupart des domaines scientifiques requièrent un apprentissage pratique qui se fait généralement dans le cadre d'un laboratoire.

Problèmes :

- Nombre important d'étudiants
- Coût élevé des laboratoires
- Problème de la distance
- Manque de rentabilité

Solution :

Développer un laboratoire virtuel qui permette de reproduire des expériences réelles et de les rendre accessibles au plus grand nombre via Internet.

Avantages d'un laboratoire virtuel

Avantages

Intégration de l'expérimentation avec l'apprentissage, sans avoir à attendre les séances de laboratoire pour pouvoir faire le lien entre théorie et pratique

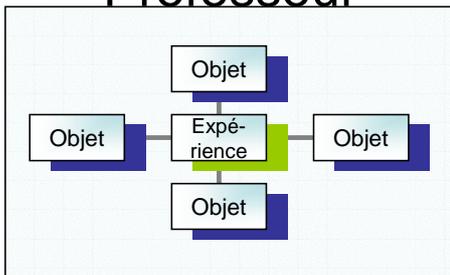
Accélération des expérimentations par le biais de la simulation, permettant de faire beaucoup plus d'études paramétriques ou d'études de cas

Accès illimité à l'expérimentation en tout temps, sauf dans le cas d'expérimentation réelle à distance.

Accélération des expériences et suppression des tâches répétitives, permettant de passer plus de temps à l'analyse et l'interprétation des résultats

Approche orientée « Objets »

Professeur



Etudiant



Expériences en cours de développement

Chromatographie échangeuse d'ions

Méthode de séparation où des ions de même signe sont séparés par élution sur une colonne remplie d'une résine finement divisée.

Lois de Newton

Première loi : Principe d'inertie
Deuxième loi : Théorème du centre d'inertie
Troisième loi : Principe de l'Action et de la réaction
Loi de la gravitation universelle

Présentation de l'Harmonie Musicale par une Notation Spatiale et une Souris à Retour de Force

Bertrand Tornil

TOBIA-IRIT
118, route de Narbonne
31062 Toulouse Cedex 4
tornil@irit.fr

Nadine Baptiste-Jessel

TOBIA-IRIT
118, route de Narbonne
31062 Toulouse Cedex 4
baptiste@irit.fr

RESUME

Cet article décrit une méthode qui facilite l'accès des utilisateurs non-voyants à l'information musicale harmonique, à l'aide d'une représentation spatiale de la musique et d'une souris à retour de force. Nous proposons tout d'abord un contexte d'utilisation de la souris à retour de force : la localisation relative. Cette dernière permet la reconstruction progressive de l'image mentale que l'on se fait d'un document en prenant conscience des positions de ses éléments. Nous présentons ensuite le contexte technique qui sous-tend notre prototype, ainsi que les diverses alternatives possibles. Nous décrivons alors le prototype de notre application musicale et nous terminons en présentant les axes de recherche que nous envisageons pour le futur : autres domaines d'application (géographie, géométrie) et contexte d'utilisation « enseignant-élève ».

MOTS CLES : Accessibilité, retour de force, SVG, notation hexagonale de l'harmonie.

ABSTRACT

This paper describes a method which enables blind users to access to harmonic information thank to a spatial representation and a force feedback mouse. First of all, we propose a general context of using our force feedback mouse: the relative localisation. The relative localisation enables the progressive rebuilding of the mental image of a graphical document by knowing the positions of its elements. Next, we present the technical context of our prototype and the possible alternatives. Then, we describe the prototype of our application and we finish by the perspective of this project : some others applicative context (geography, geometry) and "teacher-student" architecture.

CATEGORIES AND SUBJECT DESCRIPTORS: H.5.2 [User Interfaces]: Haptic I/O

GENERAL TERMS: Design

KEYWORDS : Accessibility, force feedback, SVG, harmonic hexagonal notation.

INTRODUCTION

Dans une situation classique, un utilisateur non-voyant utilise un clavier pour interagir avec un ordinateur. La machine lui répond via une synthèse vocale et/ou un afficheur braille. Cette méthode est totalement adaptée à l'accès à des documents textuels. Cependant, lors d'une navigation sur Internet, l'utilisateur non-voyant aura des problèmes avec les documents graphiques. Le World Wide Web Consortium a proposé des guides de conception [16] pour les auteurs de sites WEB, afin que ces derniers soient plus accessibles. Par exemple, une description textuelle doit être proposée à la place d'une image. Le problème persiste cependant car d'une part ces descriptions ne sont pas toujours présentes, et d'autre part parce que les descriptions textuelles peuvent vite s'avérer longues et fastidieuses à consulter.

L'utilisation de périphériques à retour de force a ainsi été étudiée dans le cadre de l'accessibilité. En effet, ces périphériques autorisent une interaction plus directe basée sur les capacités sensorielles.

Nous présentons ici une méthode pour accéder à l'information musicale (l'harmonie) utilisant une souris à retour de force. Nous allons tout d'abord présenter l'utilisation que nous faisons du retour de force pour l'accessibilité. Nous proposons un contexte spécifique : la localisation relative. Nous présenterons ensuite l'architecture client-serveur que nous avons retenue pour nos prototypes ainsi que les dépendances technologiques de notre système et les pistes possibles pour le futur. Dans la partie suivante, nous décrirons notre prototype d'application permettant l'accessibilité à l'information harmonique présentée sous une forme spatiale. Enfin nous terminerons en présentant les axes de recherche que nous envisageons pour le futur : ouverture à d'autres domaines comme la géographie et la géométrie et contexte d'utilisation « enseignant-élève ».

LA LOCALISATION RELATIVE

Le système tactilo-kinesthésique ou haptique[11] est la synthèse des mouvements d'exploration du système moteur et des perceptions du système tactile. Le sens haptique est donc à la fois effecteur et récepteur. Il consiste en :

- Le sens cutané : c'est le sens du toucher. Il permet la perception de la température, de la pression ou de la douleur. Les récepteurs sensoriels sont situés dans la peau.
- Le sens kinesthésique : il permet de ressentir la position et les mouvements du corps. Il nous permet par exemple de connaître le poids, la forme et la position d'un objet que nous sommes en train de manipuler. Il est relayé par des récepteurs sensoriels situés dans les muscles, les tendons et les articulations.

La manipulation d'un périphérique de pointage à retour de force, comme une souris, est basée sur la perception kinesthésique du bras. Le sens cutané n'est pas stimulé. Ainsi, dans notre approche, il ne s'agit pas de ressentir une texture, mais de sentir les emplacements que notre main a repérés au cours de la manipulation du périphérique.

Grâce à la mémoire sensorielle associée à notre perception kinesthésique, nous pouvons nous représenter mentalement les positions des objets les uns par rapport aux autres (Figure 1). C'est ce que nous appelons la « localisation relative ».

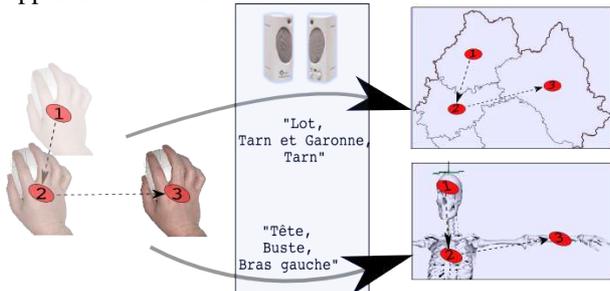


Figure 1 : La localisation relative

Couplée à un retour sonore sur chaque point aimanté, cette approche permettra la reconstruction d'une image mentale d'un objet à partir des éléments de cet objet.

ASPECTS TECHNIQUES

La Souris à Retour de Force

Nous utilisons la souris à retour de force « Wingman Force Feedback Mouse » créée par Immersion Corporation [7] et commercialisée par Logitech (Figure 2).



Figure 2 : La souris à retour de force

La surface effective de travail est très petite : 1.9cm x 2.5cm et les forces peuvent atteindre 1N. La Wingman a été originalement créée pour le jeu vidéo, mais son utilisation a également été étudiée pour l'accessibilité [2][5][12][13].

La souris à retour de force Wingman Force Feedback Mouse ne possède des pilotes que sous Windows. Le développement d'un pilote sous Linux est envisageable en s'appuyant sur le pilote iForce qui permet l'utilisation des joysticks à retour de force du même constructeur.

Enfin, Immersion Corporation propose un plugin pour navigateur Internet [6] donnant la possibilité de contrôler la souris via une programmation javascript/Ecmascript contenue dans une page HTML. Ce plugin existe pour Internet Explorer et Netscape 4. Une version pour les navigateurs basés sur le moteur Gecko (Netscape Navigator 7.x, Mozilla 1.x et ses dérivés) existe aussi mais malheureusement pas pour les versions récentes de ces navigateurs.

Le Format SVG

Nous basons le mode visuel sur le format SVG (Scalable Vector Graphic [14]), qui est le format XML pour les images vectorielles. Un fichier SVG est codé sous forme de fichier texte. Son contenu peut ainsi être indexé par les moteurs de recherche. Un fichier SVG peut de plus être généré dynamiquement côté serveur via l'utilisation des classes perl:SVG et côté client grâce à une programmation javascript.

D'un autre côté le SVG supporte le DOM (Document Object Model) et peut réagir aux événements utilisateurs (onMouseOver par exemple lorsque le pointeur de la souris passe au dessus d'une forme SVG).

Les formes décrites dans un fichier SVG sont définies explicitement par une liste de coordonnées. Ceci constitue un réel avantage comparé aux images bitmaps. Grâce à ses possibilités, le SVG a déjà été utilisé dans le cadre de l'accessibilité [2][5].

Pour l'affichage dans Internet Explorer nous utilisons le plugin Adobe SVG Viewer [1]. Ce plugin est actuellement le plus avancé en terme de respect des spécifications SVG et du support du DOM. Nous suivons également l'initiative SVG du Mozilla Project [9] qui permettra à terme de se passer de l'utilisation d'un plugin externe mais qui pour l'instant ne supporte pas toutes les spécifications SVG.

Le Retour sonore

Pour le moment, nous utilisons des fichiers sons au format mp3 préalablement enregistrés sur le serveur. La lecture d'un son est lancée grâce aux extensions SMIL du plugin Adobe SVG Viewer. Il y a cependant une limitation importante : on ne peut lire que 20 sons différents dans un fichier SVG. Les spécifications de la

future version 1.2 du SVG sont maintenant connues [15], et la balise <audio> y fait son apparition. Ainsi, quand le SVG 1.2 sera normalisé, il sera possible de lire des fichiers mp3, ogg/Vorbis ou Wav, sans limitation de nombre.

La classe perl:MIDI pourra également être utilisée pour générer dynamiquement le retour sonore. La lecture de fichiers midi dans un navigateur nécessite cependant l'utilisation d'un lecteur multimédia de préférence intégré au navigateur.

Architecture

Nous avons opté pour un contexte d'applications WEB. Nous nous appuyons donc sur un schéma client-serveur classique (Figure 3).

Nous utilisons Apache comme serveur WEB. Nous disposons également sur le serveur des fichiers SVG et des fichiers sons. Comme vu plus haut, une programmation côté serveur nous autorise également la génération d'autres fichiers SVG et sons.

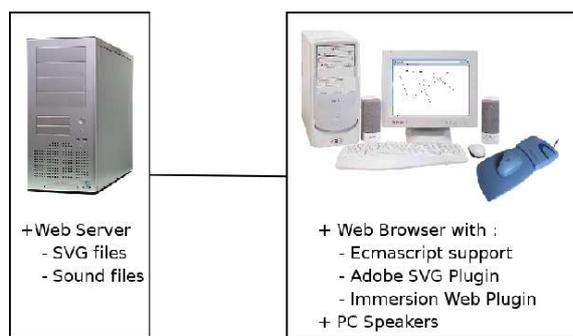


Figure 3 : Architecture client-serveur de notre application

Côté client, c'est le navigateur Internet Explorer de Microsoft qui est utilisé. C'est à l'heure actuelle le seul choix possible comme vu au dessus : support de l'Immersion Web Plugin. C'est enfin une programmation Javascript qui contrôle la souris.

APPLICATION : ACCESSIBILITÉ À L'HARMONIE

La notation hexagonale

La notation hexagonale a été proposée par le musicologue Jean-Marc Chouvel [3]. Il s'agit d'une représentation instantanée de la musique comme on peut avoir sur un manche de guitare. Les notes sont disposées sur un damier hexagonal.

Par rapport à la représentation originale de Jean-Marc Chouvel, nous avons effectué une rotation de 60 degrés afin de conserver la corrélation intuitive entre hauteur de la note et direction à l'écran : les mouvements vers le haut feront toujours monter le son dans la gamme (Figure 4).

Les principales caractéristiques de cette représentation sont explicitées dans [8] :

- L'axe vertical est composé de quintes majeures.
- L'axe diagonal-droit est composé de tierces majeures.
- L'axe diagonal-gauche est composé de tierces mineures.

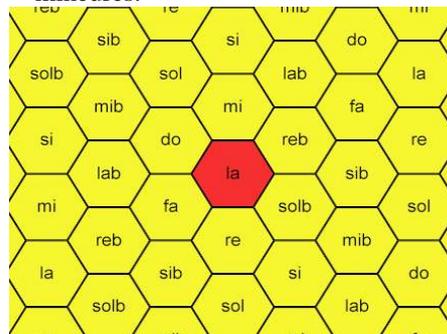


Figure 4 : La représentation hexagonale

La propriété la plus remarquable reste la suivante : Un accord (plusieurs notes) est formé par plusieurs hexagones. Si deux formes sont identiques, on a le même accord.

Enfin, les notes les plus rapides à atteindre (au sens de la loi de Fitts [4]) en partant d'un point sont les notes « proches » harmoniquement (les tierces mineures et majeures et la quinte) : ce sont les notes voisines d'un hexagone.

Chaque hexagone est un interacteur indépendant défini à partir de sa description dans le fichier SVG, par sa note (jouée lorsque le pointeur passer au dessus) et son retour de force (schématisé sur la figure 5).

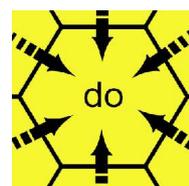


Figure 5 : Effet haptique pour une note

Dans notre prototype, nous nous servons de la représentation hexagonale pour présenter les accords harmoniques les plus simples à l'utilisateur (Figure 6). Chaque accord a donc une forme spécifique, quelque soit la tonalité.

A l'utilisation, un utilisateur non-voyant peut explorer librement le damier d'hexagones. Ensuite, les touches du clavier permettent de passer d'un accord à l'autre. Dans cette configuration, l'utilisateur n'explore plus que les notes de cet accord afin de déduire sa forme. Par exemple un triangle isocèle pointant vers la droite est un accord de quinte majeure.

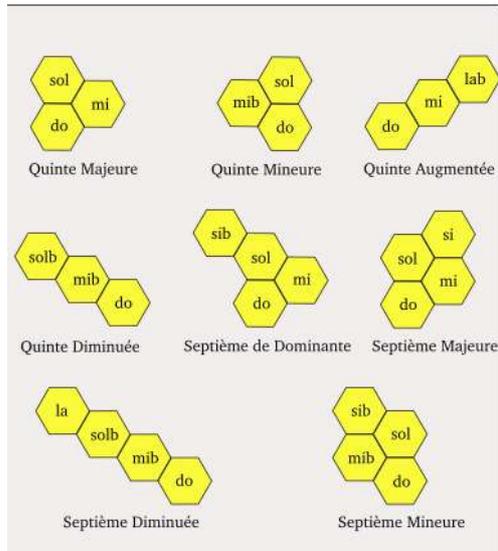


Figure 6 : Les formes des accords implémentés

PERSPECTIVES ET CONCLUSION

Nous prévoyons d'utiliser cette approche à d'autres domaines comme l'apprentissage de la géographie ou de la géométrie. Nous mettons actuellement en place un protocole de test pour évaluer notre approche dans ces différents contextes applicatifs et auprès de différents publics : personnes non-voyantes et voyantes, enfants et autistes.

Dans un premier temps, cependant, nous allons poursuivre le développement de notre prototype dédié à l'apprentissage de l'harmonie, en proposant notamment la possibilité d'enregistrer sa propre musique; la création d'un accord pourra se faire par exemple via une reconnaissance des gestes réalisés à la souris.

Enfin, nous développons actuellement une architecture orientée « enseignant-élève ». Elle devra permettre à l'enseignant de concevoir un interacteur visuel (SVG), audio(mp3) et haptique en un minimum de temps. Enfin, le paramétrage du comportement de cet interacteur pourra se faire en réponse directe aux remarques des utilisateurs lors des séances de test.

BIBLIOGRAPHIE

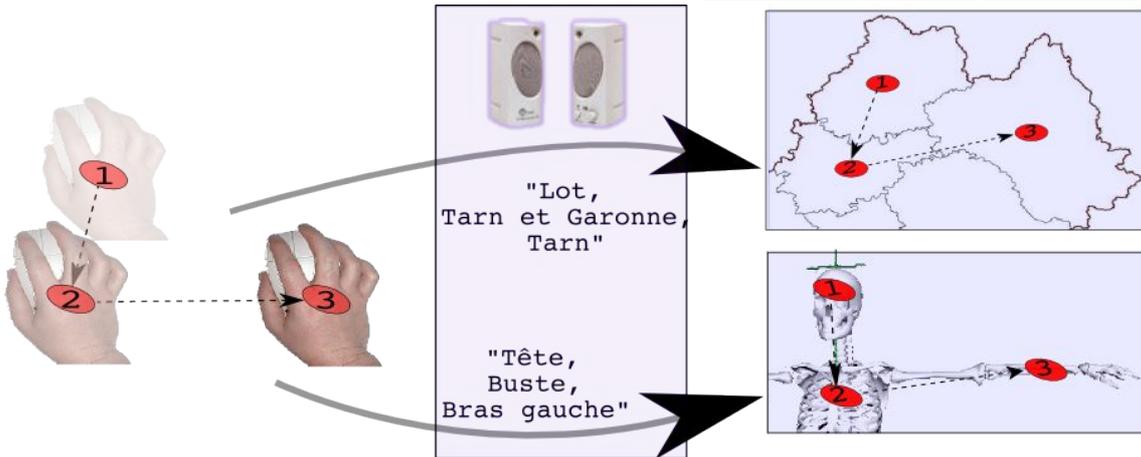
1. Adobe. *Adobe SVG Viewer*. Disponible à l'adresse <http://www.adobe.com/svg/viewer/install/main.html>, 2004.
2. Campin, B., Brunet, L.L., McCurdy, W.A., Siekierska, E.M. *Svg mapping for people with visual impairment*. In: *Proceedings of SVG Open 2003*, 2003.
3. Chouvel, J.M., Assayag, G., Cholleton, J.P. *La représentation hexagonale toroïdale, application à l'analyse harmonique de la musique d'Hermeto Pascoal*. Séminaire MaMux, 2002.
4. Fitts, P.M. *The information capacity of the human motor system in controlling the amplitude of movement*. In *Journal of Experimental Psychology* vol. 47, 1954, pp. 381-391.
5. Gardner, J., Bulatov, V. *Smart figures, svg, and accessible web graphics*. In *Proceedings of Technology And Persons With Disabilities Conference 2001*, Los Angeles, 2001.
6. Immersion : *Immersion Web Plugin*. Disponible à l'adresse <http://www.immersion.com/plugins/>, 2002.
7. Immersion : *Touchsense Devices*. Disponible à l'adresse <http://www.immersion.com/developer/technology/devices/index.php>, 2003.
8. Mathieu, B. *Outils informatiques d'analyse musicale*. Disponible à l'adresse <http://www.ircam.fr/equipes/repmus/Rapports/mathieu2002/outils-analyse-BM-2002.pdf>, 2002.
9. Mozilla SVG Project. Disponible à l'adresse <http://www.mozilla.org/projects/svg/>, 2004.
10. Offen, D., Thomlinson, B. *Good vibrations: Using a tactile mouse to convey page layout information to visually impaired computer users*. In: *Proceedings of CSUN'S Sixteenth Annual International Conference : "Technology and Persons with Disabilities"*, Los Angeles, 2001.
11. Revesz, G. *Psychology and art of the blind*. New York: Longmans, 1950.
12. Sile O'Modhrain, M., Guillepsie, R. *The moose: A haptic user interface for blind persons*. In: *Proceedings of the Third WWW6 Conference*, Santa Clara, California, 1996.
13. Wies, E., Gardner, J., O'Modhrain, M., Hasser, C., Bulatov, V. *Web-based touch display for accessible science education*. In *Haptic Human-Computer Interaction LNCS 2052*, 2001, pp. 52-60.
14. World Wide Web Consortium (W3C). *Scalable Vector Graphics (SVG) 1.1 Specifications*. Disponible à l'adresse <http://www.w3.org/TR/SVG/>, 2001.
15. World Wide Web Consortium (W3C). *Scalable Vector Graphics (SVG) 1.2. Working Draft Specifications*. Disponible à l'adresse <http://www.w3.org/TR/SVG12/>, 2004.
16. World Wide Web Consortium (W3C). *Web Accessibility Initiative (WAI)*. Disponible à l'adresse <http://www.w3.org/WAI/>, 2004.

Présentation de l'Harmonie Musicale par une Représentation Spatiale et une Souris à Retour de Force

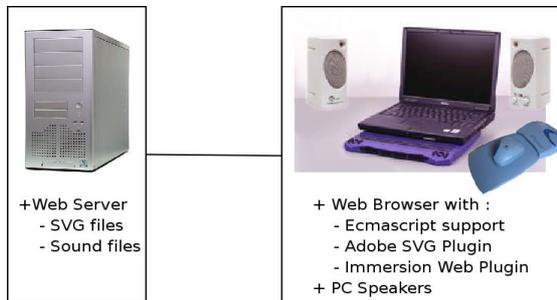
Localisation Relative

Perception kinesthésique
+ Retour sonore

Reconstruction d'une image
Mentale d'un objet à partir
de ses éléments



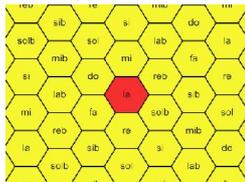
Architecture matérielle : client-serveur



Harmonie Haptique

Notation hexagonale

- Axe verticale : quintes justes
- Axe diagonale-droit : tierces majeures
- Axe diagonale-gauche : tierces mineures

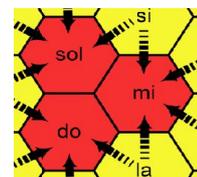


Retour multi-modal :

- Effet magnétique au centre d'un hexagone
- La note est jouée

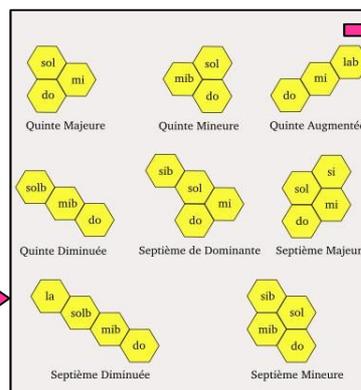


Reconnaissance des accords



Possibilité future d'éditer des partitions par reconnaissance des gestes effectués à la souris.

Propriété remarquable de la notation hexagonale :
Une même configuration représente un même accord indépendamment de la tonalité



Conduite de projet informatisé dans les PME/PMI : Les apports d'une démarche ergonomique

Virginie Valiani

Laboratoire ECCHAT-équipe *ConTActS*
Université de Picardie Jules Verne
Faculté de Philosophie et de Sciences Humaines
80 025, Amiens, Cedex 1, France
virginie.valiani@wanadoo.fr

Gérard Valléry

Laboratoire ECCHAT-équipe *ConTActS*
Université de Picardie Jules Verne
Faculté de Philosophie et de Sciences Humaines
80 025, Amiens, Cedex 1, France
gerard.vallery@libertysurf.fr

RESUME

Cette communication présente des éléments d'une recherche en cours. Cette dernière s'inscrit dans le programme Hommes, Technologies et Systèmes Complexes, axe mobilisateur de recherche financé par le Conseil Régional de Picardie. Cette étude participe aux réflexions d'un groupe interdisciplinaire : « Tolérants », réunissant des entreprises, institutions et universités, autour d'une thématique de développement technologique et industriel. La place donnée à l'ergonomie renvoie à la dimension humaine du projet. Elle ambitionne, par une approche comparative « PME/PMI – groupe », de signifier les déterminants et les effets (individuel et organisationnel), de l'intégration de la Maîtrise Statistique des Procédés dans une organisation. Tout en développant une réflexion sur la position de l'ergonomie dans une conduite de projet : des phases de négociation, aux étapes de conception, jusqu'à la mise en exploitation réelle du produit et son évolution.

MOTS CLES : ergonomie de conception, conduite de projet, utilisation et appropriation, organisation du travail, milieux industriels.

ABSTRACT

This communication presents elements of a research in progress. The latter falls under the program Complex Men, Technologies and Complex Systems, mobilizing axis of research financed by the Regional Council of Picardy. This study takes part in the reflexions of an interdisciplinary group: "Tolerants", joining together companies, institutions and universities, around a set of themes of technological and industrial development. The place given to ergonomics returns to the human dimension of the project. Her ambition, by a comparative approach "PME/PMI - group", to mean the determinants

and the effects (individual and organisational), to the integration of the Statistical Processes Control in an organization. While developing a reflexion on the position of ergonomics in a project control: to the phases of negotiation, at the stages of design, to the real introduction of the product and its evolution.

CATEGORIES AND SUBJECT DESCRIPTORS:

K.6. [Management of Computing and Information Systems]: K.6.1. : Project and People Management; H.1. [Models and Principles]: H.1.2. User/machine Systems: Human Factors.

GENERAL TERMS: Human Factors, Management

KEYWORDS: design ergonomics, project control, use and appropriation, organization of work, industrial circles.

PROBLEMATIQUE

Notre objectif est de montrer les apports d'une démarche ergonomique centrée sur l'analyse de l'activité des opérateurs [Daniellou, Terssac (de) et Schwartz, 2003], [Pomian, Pradère et Gaillard, 1997], dans les conduites de projet informatisé développées dans un premier temps en PME/PMI, et par la suite dans un groupe.

Dans un contexte où une tâche manuelle va devenir informatisée, il s'agit d'accompagner un projet de développement technique en Maîtrise Statistique des Procédés (MSP) tout en développant une problématique de recherche en ergonomie et ingénierie industrielle (associant les aspects technologiques, sociaux et organisationnels). Autrement dit, nous cherchons à approcher les technologies sous l'angle de ses enjeux sociaux et organisationnels (modalités d'appropriation, organisation des temps, évolution des rôles et des fonctions ...), à travers les domaines de conception et de mise en œuvre des technologies, dans une visée d'amélioration de la situation de travail et de contribution à la productivité.

Nos interrogations renvoient aux modalités de coopération dans la conception d'une nouvelle activité, ainsi qu'au rôle de l'ergonome et de son expertise dans les différentes phases constituant le processus d'une

conduite de projet.

Nos hypothèses, pour ce premier travail exploratoire, sont les suivantes :

- *H1* : les motivations et les enjeux d'acquisition du progiciel de MSP par les PME/PMI, portent essentiellement sur l'amélioration des processus technique et économique sans intégrer les dimensions humaine et sociale ;
- *H2* : la participation active des opérateurs et l'analyse fine de leur activité, permettent d'une part, de développer la charte et les fonctionnalités du progiciel, et d'autre part, d'appréhender les impacts socio-organisationnels.

Nous allons maintenant développer le cadre théorique sur lequel nous avons développé notre réflexion.

THEORIES

Ce premier travail cherche avant tout à poser les caractéristiques significatives des objets étudiés, c'est à dire : l'organisation et son processus, la conduite de projet et les apports de l'ergonomie.

L'organisation et son processus : recherche d'optimisation par la qualité

Au cours de ces dernières années, les entreprises ont été confrontées à une concurrence de plus en plus féroce. L'internationalisation de la compétition et la course au développement, ont poussé les entreprises à rechercher des atouts leur permettant de gagner en compétitivité. La recherche de la qualité est alors devenue un enjeu d'importance dans cette quête, ce qui par ailleurs, ne date pas d'aujourd'hui. C'est à travers la MSP et le contrôle que la qualité développe de plus en plus ses méthodes. La MSP est la première manifestation de recherche « scientifique » de la qualité. Cela consiste à suivre, à l'aide d'outils statistiques, le déroulement d'un processus. L'objectif est d'en diminuer la variabilité et de découvrir au plus tôt les dérives afin d'agir avant que la production ne sorte des limites de tolérances acceptables. Hoc (1996, p.7) décrit cette situation de supervision des processus et de contrôle, que nous reprenons sous ses termes : « l'intervention directe de la machine sur l'environnement de travail est encore loin de supplanter entièrement les opérateurs humains dans leur rôle d'acteur, mais ils ne sont plus les seuls à agir. Dans ce contexte, on a coutume de parler de contrôle quand les actions de l'opérateur humain portent assez directement sur l'environnement. Mais, à ces activités de contrôle s'ajoutent des activités de supervision de niveau plus abstrait (surveillance, raisonnement, planification, prise de décision, etc.), visant en particulier à gérer les activités de l'ensemble des acteurs (dont les machines) ». Cette présentation nous sied étant donné qu'elle situe parfaitement le contexte dans lequel notre étude est menée. En somme, nous sommes face à une situation où

l'opérateur travaille déjà en interaction avec un environnement technique, et où son activité de contrôle passe d'une modalité manuelle à une modalité informatisée. Celle-ci va donc être davantage instrumentalisée, et se complexifier. Ce qui implique une transformation des moyens opératoires et cognitifs pour la réaliser.

Malgré une volonté certaine d'amélioration, la plupart des actions entreprises conduisent généralement à la complexification des tâches à réaliser. Cette situation semble renforcer l'écart traditionnellement admis entre la prescription et la réalisation de l'activité. D'autant plus qu'elle provoque un changement pouvant aller jusqu'à la rupture avec l'organisation existante.

Conduire le changement

Le changement désigne la démarche qui accompagne la vie de toute entreprise face à l'instabilité et au développement de son environnement.

Conduire le changement, c'est à la fois anticiper, définir et mettre en place cette démarche. Ce processus commence de la perception d'un problème organisationnel à la définition d'un cadre d'actions qui permet le choix, l'élaboration et la mise en place d'une solution dans des conditions optimales de réussite. Ceci consiste à :

- Résoudre la problématique d'organisation ou de réorganisation au travers une mise en œuvre de démarches, de méthodes et d'outils ;
- Accompagner le projet d'organisation ou de réorganisation par l'utilisation d'une méthodologie de conduite de projet ;
- Prendre en compte la dimension humaine, c'est à dire : les valeurs, la culture, et les résistances au changement, au travers d'une démarche de participation, de communication et de formation/*coaching* dont l'objectif est d'amener à la compréhension et l'acceptation par les individus du changement. Conduire le changement renvoie donc à plusieurs étapes : définir la vision, mobiliser et faire participer, communiquer et informer, catalyser, piloter, concrétiser, gérer les aspects émotionnels, gérer les enjeux de pouvoir. Mais ce n'est pas que cela, c'est aussi prendre en considération les impacts socio-organisationnels, et plus spécifiquement ceux renvoyant aux nouvelles technologies (Valléry, 2001). Ces impacts sont de deux niveaux : individuel (connaissances, compétences, qualifications et décisions) et organisationnel (management, organisation du travail, responsabilités, cultures).

Nous pensons qu'une structuration rigoureuse de la démarche et la participation active des futurs utilisateurs ont des retombées concrètes pour l'action. C'est pourquoi l'ergonomie a toute sa place dans la définition et la conduite d'un projet.

Les apports de l'ergonomie dans les conduites de projet informatisé

L'ergonomie par ses différentes orientations (physique, industrielle, cognitive, conception, correction, logiciel...) va nous permettre d'une part, de développer les tenants et les aboutissants d'une conduite de projet impliquant les acteurs organisationnels ; et d'autre part, de montrer toute la nécessité de concevoir un outil centré sur des caractéristiques physiques et psychologiques des usagers tout en prenant en compte celles de son environnement.

Notre étude interroge le champ de la conception étant donné la problématique dans laquelle nous nous situons. La conception ergonomique centrée sur l'activité du futur utilisateur passe de l'analyse de l'activité de ce dernier à la spécification de solutions de conception. Ce qui permet d'anticiper les évolutions de l'activité liées à l'introduction d'un progiciel de MSP, tout en construisant itérativement la solution la plus adaptée aux futurs utilisateurs et à l'environnement dans lequel ils vont réaliser leur activité. En appliquant les recommandations et spécifications (langage de commande, interface, apprentissage utilisateur, aides ...) ainsi que certaines méthodes développées par l'ergonomie des logiciels, nous cherchons dans cette étude, à mettre en œuvre un aménagement optimisé entre les opérateurs, leur tâche, le logiciel et leur contexte professionnel, afin de prévenir les défaillances du système homme-machine et de garantir un haut niveau de performance, de fiabilité et de qualité (Brangier & Barcenilla, 2003)..

METHODOLOGIE

Notre recherche s'inscrit dans le cadre d'une Action Collective¹, projet pilote associant cinq entreprises picardes aux activités diverses (de la mécanique de précision en passant par les activités de la forge, à la fabrication de filtre à air pour engin moteur) autour d'une logique d'intégration d'un progiciel de MSP.

Cette communication présente l'étude exploratoire (développée dans le cadre d'un travail de DEA) intégrant les phases de présentation du projet (*phase 0 « présentation de l'Action Collective »*), de négociations de l'intégration d'un outil de MSP et d'étude de l'existant (*phase 1 « qualification des entreprises »* et *phase 2 « élaboration du plan de mise en œuvre »*). Autrement dit, les phases d'études en amont du changement ou de l'intégration du progiciel de MSP. La temporalité de ces actions s'est déroulée sur une période de neuf mois (janvier à septembre 2003) en fonction des différentes priorités internes des entreprises. Nous devons préciser que le processus d'une action de ce type offre la possibilité aux entreprises de poursuivre ou non l'étude à l'issue de chacune des phases (*phase 0, phase 1, phase 2*).

¹ Action collective : modalité financière d'accompagnement régionale. Ayant pour but de renforcer le tissu industriel local et d'améliorer ses performances, tout en développant des regroupements existants.

A l'heure actuelle, les entreprises ayant acceptées de poursuivre l'Action Collective sont en cours de « déploiement de la solution » (nommée *phase 3*).

Protocole

Nous avons développé un protocole d'intervention spécifique au contexte dans lequel elle se déroule. Celui-ci prend donc en compte des modalités différentes dans le but d'appréhender les différentes phases de la conduite de projet le plus finement. (voir tableau 1, fin de texte). Une collaboration spécifique s'est établie entre les consultants extérieurs supervisant le projet, la SSII, et les intervenants de l'équipe de recherche *ConTActS*. Nous devons préciser qu'une évaluation ergonomique du logiciel a été réalisée en amont de l'action collective.

La population concernée par ce recueil est constituée de 20 opérateurs (5 par entreprises) et 15 responsables hiérarchiques (Direction, Qualité, Bureau d'études et Méthodes, Maîtrise).

RESULTATS

Les premiers résultats issus des traitements statistiques portés sur l'analyse de contenu thématique des entretiens et de l'analyse des observations selon le schéma des cinq carrés de Christol et Terssac (de) [Pomian, Pradère et Gaillard, 1997], apportent des précisions sur :

- ⇒ Les motivations / enjeux,
- ⇒ La conduite de projet (communication, information, changement)
- ⇒ Les leviers / Freins – Résistances,
- ⇒ Les apports/attentes envisagés,
- ⇒ Les fonctionnalités progiciel,
- ⇒ La formation à envisagée,
- ⇒ Emplacement solution informatique et aménagement du poste,
- ⇒ Coopération entre les différents acteurs du projet.

Ces résultats sont au cœur de cette communication et ils s'y seront développés plus longuement.

DISCUSSION ET CONCLUSION.

Cette étude nous permet de poser ce fait : l'introduction d'un progiciel, dans une situation de travail nécessite une préparation aux modifications qu'il va engendrer. Il semble que les entreprises (PME/PMI) n'envisagent pas de réelle conduite de projet, étant en attente de la proposition qui va être faite. Nous posons l'hypothèse que le contexte spécifique de l'Action Collective peut conduire à cet état de fait.

La prise en compte de modifications de l'activité et des modes opératoires doit cependant se situer en amont du changement. Ce qui n'est pas le cas pour les sociétés de notre étude. Même si la solution informatique n'impose pas une recomposition complète du contenu et de l'organisation du travail, elle va nécessairement engendrer de nouvelles exigences et modalités opératoires dans la pratique de l'activité de travail. C'est pourquoi, il

est important pour les entreprises de se préparer à de telles éventualités. Des groupes de travail, ainsi qu'une communication et une participation des individus concernés par l'utilisation du progiciel, doivent donc se mettre en place pour favoriser ce type de conduite de projet.

Dans un contexte où plus de la moitié des conduites de projet développées par les entreprises n'aboutissent pas, la revue de littérature nous a permis de constater qu'il y a peu, voire pas d'interventions ergonomiques spécifiques pour les PME/PMI en ce qui concerne la conduite de projet informatisé. L'apport de l'ergonomie reste donc à démontrer. Ce qui nous permet de penser qu'il y a des possibilités d'investigations en ce qui concerne ce contexte d'intervention. C'est pourquoi, nous cherchons, à travers un travail de thèse actuel, en continuité avec les actions entreprises avec les PME/PMI engagées dans l'Action Collective, à développer de nouveaux terrains de recherche (groupe) dans le but d'approfondir cette réflexion. Cette dernière permettra de spécifier et d'orienter les différentes interrogations tournant autour : de l'intervention et la position de l'ergonome dans une conduite de projet (d'une part dans les PME/PMI et d'autre part dans les groupes/grandes entreprises) et en particulier dans le contexte d'une Action Collective ; des acteurs ; des relations entre les Bureaux Qualité / Etudes et Méthodes et production en ce qui concerne la Maîtrise Statistique des procédés.

BIBLIOGRAPHIE

1. Brangier, E. and Barcenilla, J. *Concevoir un produit facile à utiliser*. Editions d'organisations, 2003.
2. Cattan, M., Idrissi, N. and Knockeart, P. *Maîtriser les processus de l'entreprise*. Editions d'Organisation, 2001.
3. Daniellou, F., Terssac (de), G and Schwatz, Y. *Des pratiques en réflexion : 10 ans de débats sur l'intervention ergonomique*. Collection Travail et activité humaine, éditions Octarès, 2003.
4. Fernandez-Walch, S. *Management de nouveaux projets : panorama des outils et des pratiques*. AFNOR, 2000.
5. Hoc, J.M. *Supervision et contrôle de processus : la cognition en situation dynamique*. Grenoble : PUG, 1996.
6. Pillet, M. *Appliquer la Maîtrise Statistique des Procédés*. Editions d'organisation, 2002.
7. Valléry, G. *Quels sont les effets des NTIC sur le travail et l'organisation ?* In RH : les apports de la psychologie du travail, sous la direction de Levy-Leboyer, C., Huteau, M., Louche, C. & Rolland, J.P. Editions d'organisations, 2003.

Temporalité	Objectifs	Objets	Moyens
Phase 0 : présentation de l'action collective (1 jour)	Présentation Action Collective	Objectifs poursuivis	- intervenants divers (consultant, SSII, ergonome).
Phase 1 : qualification des entreprises (1 jour)	Analyse de l'opportunité de la demande	- Motivations, - Enjeux.	- entretiens collectifs et semi-directifs à l'aide d'un guide (consultant + ergonome).
PHASE 2 : ELABORATION DU PLAN DE MISE EN OEUVRE			
1er comité de pilotage phase 2 (1/2 journée)	définition des objectifs et précision de la demande	- éléments relatifs à la conduite de projet et aux choix qui ont orientés la prise de décision	- entretiens collectifs et semi-directifs (consultant + SSII + ergonome).
Première journée commune	- identification des services et des personnes, - présentation des intervenants extérieurs et de leurs objectifs.	organisation (services, production...)	- entretiens libres, - observations libres. (ergonome)
Opérateurs (1/2 journée par opérateur)	- analyse de l'activité, - analyse de la conduite de projet dans les entreprises.	- atteindre les représentations des opérateurs, - saisir l'organisation, - appréhender l'activité réelle et l'environnement.	- observations de l'activité, - entretiens semi-directifs et d'explicitation, - traces. (ergonome)
Restitution	Validation des données auprès des parties concernées.	- analyses développées à partir des données recueillies.	- entretiens libres guidés par la lecture des documents cités. (ergonome)
Dernier comité de pilotage phase 2	Présentation des différents rapports (consultant, SSII, ergonome) élaborant le plan de mise en œuvre du projet.	- propositions, - discussions.	- recueil libre des interactions par mode papier / crayon.

Tableau 1 : récapitulatif des modalités d'intervention selon les différentes phases d'analyse et de recueil

Conduite de projet informatisé dans les PME / PMI :
Les apports d'une démarche ergonomique

Virginie VALIANI sous la direction de Gérard VALLERY

Laboratoire ECCHAT – équipe ConTACTS

Université de Picardie Jules Verne

Faculté de Philosophie et Sciences Humaines

Chemin du Thil

80 025 Amiens Cedex 1

virginie.valiani@wanadoo.fr et gerard.vallery@libertysurf.fr

Résumé : Cette communication présente des éléments d'une recherche en cours. Cette dernière s'inscrit dans le programme Hommes, Technologies et Systèmes Complexes, axe mobilisateur de recherche financé par le Conseil Régional de Picardie. Cette étude participe aux réflexions d'un groupe interdisciplinaire : « Tolérants », réunissant des entreprises, institutions et universités, autour d'une thématique de développement technologique et industriel. La place donnée à l'ergonomie renvoie à la dimension humaine du projet. Elle ambitionne, par une approche comparative « PME/PMI – groupe », de signifier les déterminants et les effets (individuel et organisationnel), de l'intégration de la Maîtrise Statistique des Procédés dans une organisation. Tout en développant une réflexion sur la position de l'ergonomie dans une conduite de projet : des phases de négociation, aux étapes de conception, jusqu'à la mise en exploitation réelle du produit et son évolution. Cette communication est le développement d'une étude menée dans les PME/PMI.

Mots Clés : ergonomie de conception, conduite de projet, utilisation et appropriation, organisation du travail, milieux industriels

Problématique : Notre objectif est de montrer les apports d'une démarche ergonomique centrée sur l'analyse de l'activité des opérateurs [Daniellou, Terssac (de) et Schwartz, 2003], [Pomian, Pradère et Gaillard, 1997], dans les conduites de projet informatisé développées dans un premier temps en PME/PMI, et par la suite dans un groupe. Dans un contexte où une tâche manuelle va devenir informatisée, il s'agit d'accompagner un projet de développement technique en Maîtrise Statistique des Procédés (MSP) tout en développant une problématique de recherche en ergonomie et ingénierie industrielle (associant les aspects technologiques, sociaux et organisationnels). Autrement dit, nous cherchons à approcher les technologies sous l'angle de ses enjeux sociaux et organisationnels (modalités d'appropriation, organisation des temps, évolution des rôles et des fonctions ...), à travers les domaines de conception et de mise en œuvre des technologies, dans une visée d'amélioration de la situation de travail et de contribution à la productivité.

Nos interrogations renvoient aux modalités de coopération dans la conception d'une nouvelle activité, ainsi qu'au rôle de l'ergonomie et de son expertise dans les différentes phases constituant le processus d'une conduite de projet.

Nos hypothèses, pour ce premier travail exploratoire, sont les suivantes :

- *H1* : les motivations et les enjeux d'acquisition du progiciel de MSP par les PME/PMI, portent essentiellement sur l'amélioration des processus technique et économique sans intégrer les dimensions humaine et sociale ;

H2 : la participation active des opérateurs et l'analyse fine de leur activité, permettent d'une part, de développer la charte et les fonctionnalités du progiciel, et d'autre part, d'appréhender les impacts socio-organisationnels.

Méthodologie

Temporalité	Objectifs	Objets	Moyens
Phase 0 : présentation de l'action collective (1 jour)	Présentation Action Collective	Objectifs poursuivis	- intervenants divers (consultant, SSII, ergonomie).
Phase 1 : qualification des entreprises (1 jour)	Analyse de l'opportunité de la demande	- Motivations, - Enjeux.	- entretiens collectifs et semi-directifs à l'aide d'un guide (consultant + ergonomie).
Phase 2 : élaboration du plan de mise en œuvre			
1er comité de pilotage phase 2 (1/2 journée)	définition des objectifs et précision de la demande	- éléments relatifs à la conduite de projet et aux choix qui ont orientés la prise de décision	- entretiens collectifs et semi-directifs (consultant + SSII + ergonomie).
Première journée commune	- identification des services et des personnes, - présentation des intervenants extérieurs et de leurs objectifs.	organisation (services, production...)	- entretiens libres, - observations libres.(ergonomie)
Opérateurs (1/2 journée par opérateur)	- analyse de l'activité, - analyse de la conduite de projet dans les entreprises.	- atteindre les représentations des opérateurs, - saisir l'organisation, - appréhender l'activité réelle et l'environnement.	- observations de l'activité, - entretiens semi-directifs et d'explicitation, - traces. (ergonomie)
Restitution	Validation des données auprès des parties concernées.	- analyses développées à partir des données recueillies.	- entretiens libres guidés par la lecture des documents cités. (ergonomie)
Dernier comité de pilotage phase 2	Présentation des différents rapports (consultant, SSII, ergonomie) élaborant le plan de mise en œuvre du projet.	- propositions, - discussions.	- recueil libre des interactions par mode papier / crayon.

Les premiers résultats issus des traitements statistiques portés sur l'analyse de contenu thématique des entretiens et de l'analyse des observations selon le schéma des cinq carrés de Christol et Terssac (de) [Pomian, Pradère et Gaillard, 1997], apportent des précisions sur :

- ⇒ Les motivations / enjeux,
- ⇒ La conduite de projet (communication, information, changement)
- ⇒ Les leviers / Freins – Résistances,
- ⇒ Les apports/attentes envisagés,
- ⇒ Les fonctionnalités progiciel,
- ⇒ La formation à envisagée,
- ⇒ coopération entre les différents acteurs du projet
- ⇒ Emplacement solution informatique et aménagement du poste.

Bibliographie :

- Brangier, E., & Barcenilla, J. (2003). *Concevoir un produit facile à utiliser*. Editions d'Organisations.
- Cattan, M., Idrissi, N., & Knockaert, P. (2001). *Maîtriser les processus de l'entreprise*. Editions d'Organisation.
- Daniellou, F., Terssac (de), G., & Schwartz, Y. (2003). *Des pratiques en réflexion. 10 ans de débats sur l'intervention ergonomique*. Collection "Travail et activité humaine" sous la direction de Christian Martin et Dominique Baradat. Octarès Eds.
- Pillet, M. (2002). *Appliquer la Maîtrise statistique des procédés : MSP/SPC*. Editions d'Organisation.
- Pomian, J.L., Pradère, T., & Gaillard, I. (1997). *Ingénierie et ergonomie*. Cepadués Editions. ANACT.



Participants

Participants

Orateurs

Almeida	Rodrigo	rodrigoalmeida_fr@yahoo.fr
Appert	Caroline	Caroline.Appert@Iri.fr
Balme	Lionel	lionel.balme@imag.fr
Barralon	Nicolas	nicolas.barralon@imag.fr
Basnyat	Sandra	basnyat@irit.fr
Blanch	Renaud	renaud.blanch@Iri.fr
Bouteruche	François	Francois.Bouteruche@irisa.fr
Chevrin	Vincent	v.chevrin@ed.univ-lille1.fr
Costa-Cunha	Pascal	Pascal.Costa-Cunha@Iri.fr
Delotte	Olivier	Olivier.Delotte@ec-lyon.fr
Demeure	Alexandre	alxdmr2@yahoo.fr
Fierston	Jeremy	fierston@essi.fr
Ghorayeb	Abir	abir.ghorayeb@imag.fr
Kamel	Nadjet	kamel@ensma.fr
Lacaze	Xavier	lacaze@irit.fr
Lard	Jerome	jerome.lard@thalesgroup.com
Leal	Angelica	lealangelica@yahoo.fr
Mansoux	Benoit	benoit.mansoux@imag.fr
Merlin	Bruno	merlin@irit.fr
Pages	Nicolas	npa@info.fundp.ac.be
Raynal	Matthieu	raynal@irit.fr
Sanou	Loé	
Tornil	Bertrand	tornil@irit.fr
Uguen	Gurvan	gurvan.uguen1@etud.univ-ubs.fr
Valiani	Virginie	virginie.valiani@wanadoo.fr
Vella	Frederic	frederic.vella4@wanadoo.fr

Intervenants

Coutaz	Joëlle	Joëlle.Coutaz@imag.fr
Mackay	Wendy	wendy.mackay@inria.fr

Organisateurs

Calvary	Gaëlle	Gaëlle.Calvary@imag.fr
Dubois	Emmanuel	Emmanuel.Dubois@irit.fr
Liberati	Véronique	veronique.liberati@laposte.fr

