

Redirection d'applications existantes et nouvelles interactions pour des usages collaboratifs co-localisés sur une table interactive

Guillaume Besacier^{1,2}
guillaume.besacier@limsi.fr

Frédéric Vernier¹

frederic.vernier@limsi.fr

Olivier Chapuis^{2,3}

chapuis@lri.fr

Nicolas Roussel^{2,3}

roussel@lri.fr

¹ LIMSI-CNRS
BP 133

91403 Orsay Cedex, France

² LRI – Univ. Paris-Sud & CNRS
Bâtiment 490, Université Paris-Sud
91405 Orsay Cedex, France

³ INRIA
91405 Orsay Cedex, France

RESUME

Cet article décrit un module logiciel et un ensemble de techniques d'interaction permettant l'utilisation à plusieurs d'applications existantes non modifiées sur une table interactive. Nous décrivons tout d'abord le module logiciel que nous avons développé pour permettre l'interaction avec des applications X Window sur une table, à travers le système Metisse et la boîte à outils DiamondSpin. Nous présentons ensuite plusieurs problèmes spécifiques à la collaboration co-localisée sur une table interactive et proposons enfin plusieurs techniques d'interaction pouvant être spécifiquement mises en oeuvre pour permettre l'utilisation collaborative de ces applications initialement mono-utilisateur et mono-pointeur.

MOTS CLES : Table interactive, X Window, boîte à outils.

ABSTRACT

This paper describes a software component and several interaction techniques enabling existing non-modified desktop applications to be used by several users on a tabletop system. We describe our software implementation, design to redirect X Window applications to tabletop using the Metisse server and the DiamondSpin toolkit. We then present some co-located collaborative work problems, and suggest several interaction techniques that can be used to allow collaborative use of these single-user single-pointer applications.

CATEGORIES AND SUBJECT DESCRIPTORS: H.5.2 [User Interfaces]: Graphical user interfaces (GUI), Interaction styles, Windowing systems.

GENERAL TERMS: Design

KEYWORDS: Tabletop, X Window, toolkit.

INTRODUCTION

Les nouvelles technologies et la collaboration distante occupent une place de plus en plus importante dans le monde du travail et de la communication. Pourtant, les réunions en face-à-face restent extrêmement présentes, notamment lorsqu'il s'agit de créer ou manipuler une masse importante de documents. Les tables, les classeurs, le papier et autres accessoires de réunion ont, dans ce contexte, une place de choix. Des alternatives numériques existent sans rencontrer le succès auquel on pourrait s'attendre. Ainsi, les tables interactives, ou tabletop (par référence au desktop et au laptop), sont des ordinateurs se présentant sous la forme d'une surface horizontale, tenant lieu de périphérique d'entrée et de sortie, partagée par plusieurs utilisateurs. Le surcoût de prix ne peut pas complètement expliquer cet absence de succès et de nombreux travaux de recherche en IHM font le pari que le verrou à lever se situe entre des systèmes mal adaptés à ce nouvel usage et des utilisateurs qui ne peuvent pas se permettre de remettre complètement en question un des outils principaux de leur vie professionnelle.

Les interfaces tabletop font l'objet de nombreuses recherches, avec, par exemple, la conférence *IEEE International Workshop on Horizontal Interactive Human-Computer Systems*, et un numéro spécial de *Computer Graphics and Applications* sur les systèmes tabletop [9].

Différentes approches ont été explorées, tant pour les systèmes d'entrées (vision [3], tactile, à deux états via un contact électrique [6, 12], ou à plus de deux états [7], gants haptiques [1],...) que pour les sorties (du simple projecteur au HMD individuel [1], en passant par des affichages sensibles à la position autour de la table [10]). Chacune de ces techniques apporte des possibilités d'interactions ou des informations supplémentaires susceptibles d'améliorer la prise en compte de la situation de réunion (identification de l'utilisateur, possibilité de dé-

finir des documents privés, manipulation directe ou à l'aide d'outils, points de contacts multiples, utilisation temps réel, etc.).

Les systèmes de tables interactives introduisent de nouveaux défis dans la conception d'interactions et d'applications. Nous pouvons entre autre citer :

- plusieurs utilisateurs partagent la même surface et peuvent interagir ensemble ou en parallèle ;
- ils sont placés tout autour de la table, et ont donc des points de vues différents sur les documents ;
- la quantité de documents numériques disposés sur la table peut être très importante, et inclure des documents personnels à chaque utilisateur.

Nous relevons plusieurs difficultés auxquelles font faces les études sur les tables interactives. Outre l'absence du dispositif parfait (table trop petites, ombres dues aux projecteurs, latence des systèmes de vision, etc.), nous notons que la plupart des travaux de recherche testent de nouvelles techniques d'interaction sur des documents statiques (par exemple : trier des photographies [8], organiser des objets dans l'espace [17]) ou se servent d'icônes abstraites pour figurer des documents [1]. L'utilisation de documents statiques permet sans doute de tester un grand nombre de techniques d'interaction mais ne préfigure certainement pas l'utilisation réelle de ce type de système.

La boîte à outils DiamondSpin [15] permet de construire des interfaces pour les tables interactives, en se basant sur les composants d'interface utilisateur classiques (tel que les fenêtres, les boutons, ou les barres de menus) augmentés d'interactions spécifiquement conçues pour les tables interactives (comme la possibilité de tourner les fenêtres, d'attacher un clavier virtuel à une zone de texte, ou de déplacer un menu surgissant).

Les interactions classiques sur les tables ont, dans nos essais, souvent suscité un intérêt des utilisateurs pour la possibilité d'avoir d'abord leurs documents standard avant de réfléchir à la création de nouvelles formes de supports au travail de réunion. Or il n'est pas raisonnable d'envisager la réécriture d'une suite bureautique complète, d'un navigateur web, ou des applications métiers spécifiques à un groupe donné d'utilisateur.

Le but de cet article est de montrer comment rediriger l'interface d'une application existante non modifiée vers le système de table interactive. Nous présenterons ensuite des interactions destinées à la collaboration co-localisée sur une table, et la façon dont nous pouvons les mettre en œuvre plus facilement et les tester de manière plus approfondie, grâce à notre technique.

IMPLÉMENTATION D'UN CLIENT METISSE

Metisse est basé sur X Window [14], le système couramment utilisé pour gérer l'interaction homme-machine par l'écran, la souris et le clavier sous Linux et les diverses variantes d'UNIX. Au lieu de faire le lien entre les applications et l'écran, le clavier, et la souris, le serveur Metisse, développé pour le système de fenêtrage du même nom [5], stocke les fenêtres à afficher sous forme d'images numériques, et simule l'écran et le clavier. Le serveur Metisse est un serveur X modifié pour dessiner les fenêtres hors écran, rendre leurs images accessibles à un compositeur et rediriger les événements souris/clavier envoyés par ce compositeur vers les applications (voir figure 1). Les applications, qui n'ont pas besoin d'être modifiées ou recompilées, ne savent pas qu'elles ont été redirigées.

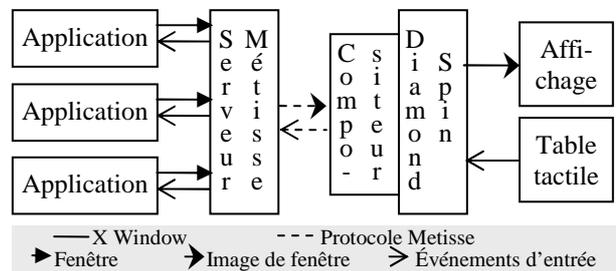


Figure 1 : Architecture de la redirection d'applications.

Affichage

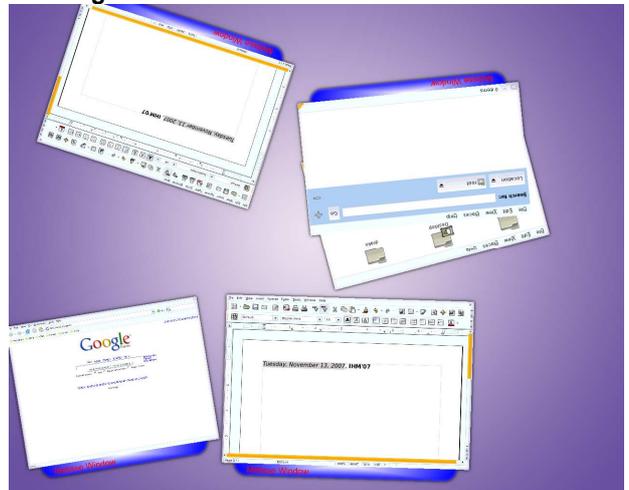


Figure 2 : Plusieurs applications existantes (Firefox, Nautilus et OpenOffice) ouvertes par deux utilisateurs. Le document OpenOffice est partagé.

Le serveur X de Metisse rend accessible les images des fenêtres par une connexion réseau. Nous avons donc implémenté un module client pour DiamondSpin [15], une boîte à outils open-source pour table interactive. Du point de vue du serveur Metisse, notre module est un compositeur qui va récupérer les images des fenêtres et générer des événements clavier et souris. Le protocole de

communication entre le serveur Metisse et le compositeur est dérivé du protocole Remote Frame Buffer (RFB) [13] : les mises à jour des fenêtres (et non pas seulement du frame buffer) sont envoyées sous la forme d'images rectangulaires correspondants à la plus petite zone englobant la modification. Ces images sont récupérées et stockées par DiamondSpin dans des textures OpenGL, qui les positionne ensuite de manière appropriée par rapport aux utilisateurs (orientées dans le sens de lecture de l'utilisateur qui manipule la fenêtre,...) et les affiche sur la table.

Les décorations des fenêtres, tel que la barre de titre ou le bouton de fermeture, peuvent être inclus ou non par le serveur dans ces images. Nous avons choisis de ne pas les inclure. En effet, les décorations conçues pour les environnements de bureau ne nous semblent pas adaptées aux tables interactives. Notre boîte à outils fournit des décorations adaptées à cette situation. Les barres de titre des fenêtres sont plus grosses, pour une manipulation avec le doigt plus facile, et situées en bas des fenêtres, pour éviter que la main de l'utilisateur ne cache le contenu quand il manipule la fenêtre par sa barre de titre (voir figure 2).

Les événements clavier et souris

Dans un environnement classique, l'élément essentiel de la gestion du clavier et de la souris est le focus. L'ensemble du système, incluant toutes les applications ouvertes, est mono utilisateur. A un instant donné, un seul élément de l'interface reçoit les événements du clavier. D'autre part, certains éléments du système sont modaux, soit au niveau d'une application (par exemple, une boîte de dialogue modale), soit au niveau de l'ensemble du système (par exemple, un menu surgissant).

Sur une table interactive, en revanche, plusieurs utilisateurs interagissent en même temps avec le système. Une gestion spécifique du focus est nécessaire pour rendre ceci transparent aux applications redirigées. Pour l'instant, il nous faut soit utiliser un serveur Metisse par application, soit vérifier que deux utilisateurs ne touchent pas la table en même temps.

Dans DiamondSpin, nous utilisons des claviers virtuels (projetés sur la table), chacun associé à une fenêtre donnée. Il n'y a donc pas un focus clavier, ni un focus clavier par utilisateur, mais un focus clavier par fenêtre. Le mécanisme existant de l'application pour déterminer le composant (zone de texte, bouton, etc.) actif peut donc être conservé.

APPLICATIONS AUX TABLES INTERACTIVES : LA COLLABORATION CO-LOCALISÉE

Les tables interactives sont utilisées pour des applications collaboratives entre les utilisateurs co-localisés. Cette forme de collaboration diffère des travaux habituels sur les collecticiels car une large partie des problè-

mes classiques peuvent être résolus par des protocoles sociaux mis en place par les utilisateurs. Le partage de la même surface physique de travail permet aux participants d'avoir connaissance des actions des autres utilisateurs, mais également de leurs intentions. Par exemple, il est rare que deux participants essaient de déplacer la même fenêtre au même moment car ils voient les autres participants tendre le bras vers la fenêtre avant qu'il ne commence à la déplacer.

En revanche, la proximité avec les documents de l'ensemble des participants peut induire un comportement de « do it yourself ». Si un utilisateur détecte une faute mineur (d'orthographe, par exemple), dans le document d'une autre personne, il est peut être plus simple pour lui de saisir le document, corriger la faute, et remettre le document à sa place que de prévenir le propriétaire du document.

Télépointeur

Le télépointage permet à un utilisateur de mettre en évidence une zone d'un document partagé sur toutes les copies du document, à partir d'une action sur sa copie du document. La copie de documents est décrite dans [4].

Dans le cas d'un document papier imprimé en plusieurs exemplaires et, par exemple, distribué aux participants d'une réunion, l'utilisateur devrait d'abord repérer la zone à mettre en évidence sur sa copie, puis la repérer sur la copie de l'autre participant (qu'il voit à l'envers), et enfin se pencher sur la table de réunion pour pointer la zone.

Notre système lui permet de se restreindre à la première étape, à savoir repérer la zone sur sa copie numérique du document. Il tire parti de la synchronisation entre toutes les vues du document, puisqu'il ne s'agit que d'une image numérique dupliquée et non de vues indépendantes. Nous exprimons donc simplement cette zone en termes de pixels, et nous l'affichons en surimpression de la texture de la fenêtre, par exemple sous la forme d'une zone plus claire (métaphore du spot de lumière).

Annotation collaborative

L'annotation à main levée (répandue sur les tables interactives) de documents d'applications non modifiées se heurte au problème de l'opacité des processus de l'application pour le système de table interactive. En effet, l'annotation ne peut se faire qu'au travers d'une vue, par l'intermédiaire des fenêtres redirigées. La vue peut changer (par exemple elle peut être équipée d'une barre de défilement), et il faut alors reconstruire le lien visuel entre les annotations et le contenu des documents.

Nous proposons d'instrumenter les composants qui servent couramment à effectuer de tels changements sur la vue, en utilisant les technologies d'accessibilité. Ces

technologies permettent à des applications tierces (habituellement des applications proposant un moyen alternatif d'utiliser un ordinateur aux personnes souffrants de handicaps visuels) d'énumérer et de contrôler la hiérarchie des composants d'interface utilisateur. Elles sont mises en œuvre dans Metisse [16].

CONCLUSION

Dans cet article, nous avons présenté une technique pour intégrer des applications existantes dans un système de table interactive. L'utilisation d'applications existantes nous permet d'être au plus près des usages des utilisateurs, et de leur proposer de nouveaux modes d'interactions adaptées aux tâches qu'ils doivent effectuer.

Nous avons implémenté cette technique sous la forme d'un module client pour le serveur Metisse, que nous avons intégré à la boîte à outils open-source Diamond-Spin. Ce module est écrit en Java et l'affichage est effectué avec OpenGL. Il est constitué d'environ 2000 lignes de code, et le rafraîchissement s'opère à 20 images par secondes sur un ordinateur PC standard.

Enfin, nous avons proposé deux interactions, permettant à plusieurs utilisateurs co-localisés autour d'une table interactive d'utiliser une application existante de manière collaborative : le télépointage et l'annotation collaborative du document au travers de sa vue.

Dans un avenir proche, l'évolution des technologies pour les ordinateurs de bureaux devrait nous permettre d'aller plus loin dans la redirection d'applications vers une table interactive. Ainsi Microsoft développe Multipoint [11], un système permettant de connecter plusieurs souris à un ordinateur PC, chaque souris correspondant à un pointeur individuel. Cette technologie incorporera un système de gestion du focus adapté à un environnement multi pointeurs que nous pourrions réutiliser. Le support de plusieurs curseurs système fera également parti du serveur X de Metisse grâce au projet MPX (the Multi-Pointer X Server, <http://wearables.unisa.edu.au/mpx/>).

REMERCIEMENTS

Nous tenons à remercier Michel Beaudouin-Lafon et Gaëtan Rey pour leur soutien et leurs commentaires. Le projet DigiTable est financé par le RNTL. Nous remercions la région Île-de-France pour son soutien financier.

BIBLIOGRAPHIE

1. Agarawala, A., Balakrishnan, R. Keepin' it real: pushing the desktop metaphor with physics, piles and the pen. *Proc. CHI 2006*, pp. 1283-1292.
2. Benko, H., Ishak, E., Feiner, S. Cross-Dimensional Gestural Interaction Techniques for Hybrid Immersive Environments. *Proc. IEEE Virtual Reality (VR'05)*, pp. 209-216, 327.

3. Bérard, F. The Magic Table: Computer-Vision Based Augmentation of a Whiteboard for Creative Meetings. *IEEE International Conference in Computer Vision, Workshop on Projector-Camera Systems (PROCAMS'03)*, Nice, France.
4. Besacier, G., Rey, G., Najm, M., Buisine, S., Vernier, F. Paper Metaphor for Tabletop interaction design. To appear in *Proc. of HCI International 2007*, 22-27 July 2007, Beijing, P.R. China.
5. Chapuis, O., Roussel, N. Metisse is not a 3D desktop! *Proc. UIST'05*, pp. 13-22.
6. Dietz, P., Leigh, D. DiamondTouch: A Multi-User Touch Technology. *Proc. UIST 2001*, pp. 219-226.
7. Forlines, C., Shen, C., Buxton, B. Glimpse: a novel input model for multi-level devices. *Ext. Abstracts CHI 2005*, pp. 1375-1378.
8. Hinrichs, U., Carpendale, S., Scott, S. D., Pattison, E. Interface Currents: Supporting Fluent Collaboration on Tabletop Displays. *Proc. of Smart Graphics 2005*, pp. 185-197.
9. *IEEE Computer Graphics and Application*. Interacting with Digital Tabletops. Volume 26, number 5, september/october 2006.
10. Kakehi, Y., Iida, M., Naemura, T., Shirai, Y., Matsushita, M., Ohguro, T. Lumisight Table: Interactive View-Dependent Tabletop Display Surrounded by Multiple Users. *IEEE Computer Graphics and Applications*, 25(1):48-53, January 2005.
11. Pawar, U., Toyama, K., Pal, J. Multiple Mice for Computers in Education in Developing Countries. *International Conference on Information Technologies and Development*, May 2006.
12. Rekimoto, J. SmartSkin: An Infrastructure for Free-hand Manipulation on Interactive Surfaces. *Proc. CHI 2002*, pp. 113-120.
13. Richardson, T., Wood, K. R. *The RFB Protocol Version 3.3*. AT&T Laboratories Cambridge, January 1998.
14. Scheifler, R. W., Gettys, J. The X window system. *ACM Transactions on Graphics (TOG)*, Volume 5, Issue 2 (April 1986), pp. 79-109.
15. Shen, C., Vernier, F., Forlines, C., Ringel, M. DiamondSpin: An Extensible Toolkit for Around-the-Table Interaction. *Proc. CHI 2004*, pp. 167-174.
16. Stuerzlinger, W., Chapuis, O., Phillips, D., Roussel, N. User interface façades: towards fully adaptable user interfaces. *Proc. UIST 2006*, pp. 309-318.
17. Wu, M., Balakrishnan, R. Multi-Finger and Whole Hand Gestural Interaction Techniques for Multi-User Tabletop Displays. *Proc. UIST03*, pp. 193-202.