

Support du multitouch dans Linux

Benjamin Tissoires, Stéphane Chatty

▶ **To cite this version:**

Benjamin Tissoires, Stéphane Chatty. Support du multitouch dans Linux. FITG 2010, Forum sur l'Interaction Tactile et Gestuelle, Jun 2010, Lille, France. pp xxx. hal-01022258

HAL Id: hal-01022258
https://enac.hal.science/hal-01022258

Submitted on 23 Jul 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Support du multitouch dans Linux

Benjamin Tissoires[1,2]

[1]DTI R&D
7 avenue Edouard Belin
31055 Toulouse
tissoire @ cena.fr

Stéphane Chatty[2]

[2]ENAC
7 avenue Edouard Belin
31055 Toulouse
stephane.chatty @ enac.fr

RESUME

Cet exposé présente nos derniers travaux pour intégrer le multitouch nativement dans le système Linux. Nous travaillons actuellement sur deux niveaux, le noyau et le serveur X. Les objectifs sont pour nous d'intégrer les travaux de recherche issus de la communauté dans ce système d'exploitation.

MOTS CLES : Multitouch, Linux.

INTRODUCTION

Programmer des interactions tactiles et gestuelles requiert d'accéder aux ressources d'entrée des différents dispositifs d'acquisition. Sous le système d'exploitation Linux, le support du multitouch n'en est encore qu'à son balbutiement, et il n'existe pas de moyen standardisé et facile d'utilisation pour accéder à ces périphériques. Notre laboratoire s'investit donc dans le support du multitouch sous Linux pour permettre à tout un chacun de bénéficier des dernières avancées dans ce domaine, et servira notamment aux chercheurs en Interaction Homme Machine. Par ailleurs, ceci nous permettra de résoudre les problèmes rencontrés sous Windows ou Mac. S'il est vrai que ces systèmes d'exploitation supportent dans leur dernière version le multitouch, ces implémentations ne peuvent gérer qu'une seule application en plein écran à la fois, problème que nous sommes en mesure de résoudre grâce à Linux. Notre exposé s'articule autour de trois axes: tout d'abord, nous présenterons la partie noyau de ce support, puis nous aborderons la partie serveur graphique (Xorg) et enfin, nous verrons quelles perspectives sont envisageables.

LE NOYAU

Le sous-système d'input de Linux repose sur un principe simple : des séquences (attribut, valeur) séparées par des messages de synchronisation. Ce système ne prévoit pas que le même nom d'attribut ('X' par exemple) soit utilisé plusieurs fois entre deux messages de synchronisation. Il a donc fallu modifier le protocole, ce qui a été fait dès le noyau 2.6.30. Ce besoin de modification est symptomatique de problèmes que rencontrent la plupart des systèmes de description des entrées, y compris le standard HID : leur pouvoir d'expression est trop faible pour les entrées multiples. Il est alors impossible pour un système de fournir un driver unique et cohérent pour tous les périphériques multitouch existants. S'il est vrai que Microsoft Windows semble fournir un tel driver, les applications doivent régler manuellement les spécificités des différents périphériques d'entrée. Les utilisateurs souhaitent cependant que toutes leurs applications soient compatibles sans configuration spéciale, ce qui est actuellement impossible à réaliser avec un système comme Windows. En particulier, pendant une longue phase de transition la plupart des applications n'utiliseront qu'un point de contact, et il faut donc fournir une émulation d'écran tactile simple. Pour s'assurer qu'en cas de contact multiple c'est toujours le premier doigt posé qui est pris en compte dans l'émulation, l'algorithme est différent pour chaque matériel. Nous développons donc des drivers Linux pour les périphériques multitouch afin de tenir compte de chacune de leurs spécificités, et de standardiser leur comportement pour les couches supérieures.

LE SERVEUR X

Une fois ces drivers noyau écrits, le périphérique est normalisé pour une utilisation plus générique. Dans cette partie de notre exposé, nous détaillerons comment nous permettons aux couches applicatives de plus haut niveau de travailler avec des périphériques multitouch. Ces travaux se basent sur Xinput 2 et MPX[4] (Multi Pointer X) qui permettent d'avoir plusieurs couples pointeurs/claviers et de les configurer par l'utilisateur. Notre travail consiste donc à séparer les différents touchers reçus au travers du périphérique vers un ensemble de périphériques virtuels.

PERSPECTIVES

Depuis la version 2.6.32 du noyau Linux, nos drivers sont intégrés directement dans la branche stable. Nous en sommes actuellement à 8 drivers. Ces 8 périphériques sont donc reconnus dès leur branchement. Actuellement, nous discutons avec différents responsables de ces systèmes logiciels, c'est à dire avec les développeurs noyau, Xorg, GTK, et Qt ainsi que les responsables de distributions comme Ubuntu, pour intégrer nos travaux dans les couches supérieures. En particulier, l'endroit où ajouter la reconnaissance de gestes fait l'objet de débats, qui seront abordés par notre autre exposé. De cette manière, nous sommes capables de maîtriser un peu plus la chaîne de l'interaction de ce système d'exploitation

pour nos applications futures. Ainsi, nous sommes en mesure de proposer et d'intégrer effectivement des concepts issus de la recherche au sein de Linux tels que ceux présentés par Stéphane Chatty à UIST 94 [1], ou encore Icon[3] ou plus récemment Istar[2].

BIBLIOGRAPHIE

1. Chatty, S. Extending a graphical toolkit for two-handed interaction. In *UIST '94: Proceedings of the 7th annual ACM symposium on User interface software and technology*, pages 195–204, New York, NY, USA, 1994. ACM.

2. Chatty, S. Supporting multidisciplinary software composition for interactive applications. In *Software Composition*, pages 173–189, 2008.

3. Dragicevic, P., and Fekete, J.-D. Support for input adaptability in the icon toolkit. In *ICMI '04: Proceedings of the 6th international conference on Multimodal interfaces*, pages 212–219. ACM, 2004.

4. Hutterer, P. *Groupware Support in the Windowing System*. PhD thesis, School of Computer and Information Science, University of South Australia, 2008.

BIOGRAPHIES

Benjamin Tissoires est actuellement en troisième année de thèse à l'Université de Toulouse, au sein des laboratoires de la DTI R&D et de l'ENAC, sous la direction de Stéphane Conversy. Il développe des drivers multitouch sous Xorg depuis octobre 2009.

Stéphane Chatty dirige actuellement le LII (Laboratoire d'Informatique Interactive) de l'ENAC. Ses recherches portent sur la modélisation des logiciels interactifs et de leur architecture. Son travail sur les drivers est en partie financé par le projet ShareIT soutenu par Aérospace Vallée, qui porte sur les écrans multitouch dans les cockpits d'avions.