Guide pratique des tablettes USB Graphire de Wacom

Version française du guide pratique Wacom Graphire USB Mini-HOWTO

Arnaud Claden

<tiamat CHEZ club TIRET internet POINT fr>

Joseph Mokadim

<dagdadiamaith CHEZ free POINT fr>

Adaptation française

Jean-Philippe Guérard

<fevrier CHEZ tigreraye POINT org>

Relecture de la version française

Jean-Philippe Guérard

<fevrier CHEZ tigreraye POINT org>

Préparation de la publication de la v.f.

Version : 2000-06-01.fr.1.1

26 janvier 2007

Historique des versions		
Version 2000-06-01.fr.1.1	2007-01-26	JM, JPG
Inclusion de quelques corrections du traducteur.		
Version 2000-06-01.fr.1.0	2006-03-20	JM, JPG
Première traduction française. Mise à jour des liens de la version originale. Suppression du lien vers xink qui semble ne plus exister.		
Version 2000-06-01	2000-06-01	AC

Résumé

Ce document explique comment installer une tablette USB Graphire de Wacom afin de l'utiliser sous Linux (console et X). Il va de la configuration du noyau à celle de l'application. Ces informations devraient être également utilisables pour une tablette USB Intuos.

Table des matières

Droits d'utilisation (copyright)

Introduction

Ce document est un guide d'installation d'une tablette Wacom Graphire USB sous Linux. C'est une installation assez complexe, mais ce guide vous guidera pas à pas tout au long du processus.

Cet ouvrage a été principalement réalisé à partir du guide pratique des tablettes Wacom de Stephan Runkel, mais il apporte des éléments relatifs au paramétrage USB absents du document original. Pour plus d'informations sur les tablettes Wacom en général, reportez-vous au guide pratique de Stephan Runkel.

J'ai écrit ce document car j'ai eu beaucoup de mal à installer ma tablette Graphire flambant neuve sous Linux. Elle fonctionne à présent parfaitement bien avec mon portable Toshiba 4030CDT.

Nouvelles versions de ce document

Vous trouverez la plus récente version française de ce document à l'adresse : http://www.traduc.org/docs/howto/lecture/Wacom-USB-mini-HOWTO.html.

La plus récente version originale de ce document est disponible à l'adresse : http://tiamat.club.fr/computer/howto/Wacom_USB-Mini_HOWTO.html.

Commentaires et corrections

N'hésitez pas à faire parvenir vos commentaires, corrections et ajouts à l'adresse : <`tiamat CHEZ club TIRET internet POINT fr`>

N'hésitez pas à faire parvenir vos commentaires et suggestions concernant l'adaptation française de ce document au projet Traduc.org à l'adresse : <`commentaires CHEZ traduc POINT org`>.

Remerciements

Je souhaite remercier toutes les personnes qui travaillent sur la partie USB du noyau et spécialement Vojtech Pavlik pour son travail sur le module wacom.

Bien sûr, je remercie Stephan Runkel pour son formidable guide pratique sur les tablettes Wacom, qui m'a beaucoup aidé dans le processus de paramétrage.

Des remerciements spéciaux vont aussi à Frédéric Lepied pour son travail sur le module xf86Wacom.

Limitation de responsabilité

À ma connaissance, les informations de ce document sont correctes. Mais rien ne garantit que je n'aie pas fait d'erreur. Je vous invite donc à être prudent lorsque vous suivez les indications de ce guide pratique, et ceci plus spécialement si elles ne vous semblent pas logiques. Rien de ce que j'indique ne devrait avoir d'effet préjudiciable à votre ordinateur mais, juste au cas où, je décline toute responsabilité quant aux dommages causés par l'utilisation des informations contenues dans ce guide. Ce document n'est pas un guide pratique du noyau, je supposerai donc connue la façon de télécharger, d'installer, de paramétrer et de compiler un nouveau noyau.

Matériel et logiciels nécessaires

Matériel

Le plus important bien sûr, est de posséder un ordinateur avec un port USB et une tablette USB Graphire de Wacom.

Noyau

Lors de cette installation, j'ai utilisé le plus récent noyau en développement : linux-2.3.99-pre6. Il est assez stable et possède toutes les capacités nécessaires. Vous pouvez également utiliser un noyau 2.2 en utilisant les correctifs appropriés[1 [p 11]] : http://web.archive.org/web/20010124002200/www.suse.cz/development/usb-backport/.

Programmes en mode console

Pour la console Linux, le seul programme dont vous ayez besoin est gpm, disponible sur http://linux.schottelius.org/gpm/.

XWindow

Pour utiliser toutes les capacités de votre tablette, il vous faudra utiliser XInput. Plus spécifiquement vous aurez besoin du module xf86Wacom.so (qui est fourni avec les versions les plus récentes de XFree86, et ceci depuis la version 3.3.5). Une autre solution (que vous devrez peut-être utiliser) est d'utiliser le module xf86WacomUSB.so, qui est une adaptation au port USB, en cours de développement, du xf86Wacom.so original. J'en parlerai plus tard dans ce document.

Autres

Les applications compatibles XInput sont :

1. The Gimp - un puissant programme de manipulation d'images dans le style de Photoshop. La version instable 1.1.x en cours de développement inclut en standard la compatibilité XInput. Vous pouvez obtenir Gimp sur http://www.gimp.org.

2. gsumi - un programme simple de dessin en noir et blanc qui permet de dessiner et d'effacer en fonction de la pression et de l'inclinaison. Vous pouvez l'obtenir via sa page web : http://www.gtk.org/~otaylor/gsumi/.

3. RasMol - un programme de visualisation moléculaire qui permet d'utiliser des cadrans via XInput. Reportez-vous à http://www.umass.edu/microbio/rasmol/distrib/rasman.htm.

4. xinput (de Frédéric Lepied) - un utilitaire très utile pour configurer et tester les périphériques XInput. xinput est disponible sur : ftp://ftp.x.org/contrib/utilities/xinput-1.2.tar.gz.

Noyau et matériel

Paramétrage du noyau

Comme je l'ai dit précédemment, les options que je vais donner viennent du noyau 2.3.99pre6. Si tout n'est pas exactement identique, je pense que vous arriverez bien à deviner quoi choisir ;)

Rappel

`[ ]` représente un choix entre « oui » et « non » (y/n), et <> représente les options « oui », « non » et « module » (y/n/m) du noyau.

En premier lieu, vous devez autoriser le code expérimental :

```
Code Maturity level options --->
[*] Prompt for development and/or incomplete code/drivers
```

Si vous ne le faites pas, vous n'aurez pas accès à certains pilotes.

Ensuite, allez à la section USB.

```
USB support --->
<*> Support for USB
[*] USB verbose debug messages
--- Miscellaneous USB options
[*] Preliminary USB device filesystem
```

Avec cette option, vous activerez les nouvelles fonctionnalités de /dev/usb/* (que, bien sûr, vous devrez créer : à voir plus loin).

```
--- USB controllers
<*> UHCI (Intel PIIX4, VIA, ...) support
< > OHCI (Compaq, iMacs, OPTi, SiS, ALi, ...) support
```

Vous devez choisir un et un seul d'entre eux. Donc, si l'un ne fonctionne pas, choisissez l'autre. Si vous utilisez un système x86, il est probable que vous preniez le pilote UHCI.

```
--- USB Devices
< > USB Printer support
< > USB Scanner support
< > USB Audio support
< > USB Modem (CDC ACM) support
```

Si vous possédez un de ces matériels, activez ces options à votre guise.

```
<M> USB Serial Converter support
```

Ces options doivent être activées pour que la tablette Wacom fonctionne.

```
[*] USB Generic Serial Driver
[ ] USB Handspring Visor Driver
[ ] USB ConnectTech WhiteHEAT Serial Driver (EXPERIMENTAL)
[ ] USB FTDI Single Port Serial Driver (EXPERIMENTAL)
[ ] USB Keyspan PDA Single Port Serial Driver (EXPERIMENTAL)
[ ] USB ZyXEL omni.net LCD Plus Driver (EXPERIMENTAL)
[*] USB Serial Converter verbose debug
```

Cette dernière option permet d'avoir tous les messages (ce qui peut être très utile au début). Vous pourrez la désactiver lorsque tout fonctionnera correctement.

```
< > USB Kodak DC-2xx Camera support
< > USB Mustek MDC800 Digital Camera support (EXPERIMENTAL)
< > USB Mass Storage support (EXPERIMENTAL)
< > USS720 parport driver
< > DABUSB driver
< > PLUSB Prolific USB-Network driver (EXPERIMENTAL)
< > USB ADMtek Pegasus-based device support (EXPERIMENTAL)
< > USB Diamond Rio500 support (EXPERIMENTAL)
--- USB HID
```

Le HID est l'interface entre vous et vos périphériques. Vous DEVEZ l'activer.

```
<M> USB Human Interface Device (HID) support
< > USB HIDBP Keyboard support
< > USB HIDBP Mouse support
<M> Wacom Intuos/Graphire tablet support
```

Nous y sommes, le support Wacom devrait être activé (à moins que vous ne lisiez ceci que pour le plaisir ;))

```
< > Logitech WingMan Force joystick support
< > Keyboard support
<M> Mouse support
```

À activer pour utiliser la souris Wacom Intuos/Graphire.

```
(1024) Horizontal screen resolution
(768) Vertical screen resolution
<M > Joystick support
<M> Event interface support
```

Compilation du noyau

Maintenant, le plus facile :

Lorsque vous sortez du paramétrage, suivez la procédure classique :

```
make dep bzImage modules modules_install
```

Tapez ceci tel quel et l'ordinateur effectuera toutes les étapes les unes après les autres.

Ensuite, copiez le fichier bzImage dans /boot, copiez System.map dans /boot, modifiez votre lilo.conf pour pointer vers votre nouveau noyau et tapez `/sbin/lilo` à l'invite.

Note

Bien sûr, tout ceci (configuration et compilation du noyau, ajustement de lilo, doit être fait sous le compte root).

Avant de redémarrer

FStab

Si vous voulez éviter d'avoir un message d'erreur au prochain redémarrage, vous devrez faire ce qui suit :

Modifiez le fichier /etc/fstab et ajoutez les lignes suivantes :

```
none    /proc/bus/usb    usbdevfs    defaults    0    0
```

La ligne ci-dessus pour activer le système de fichier usb.

```
none    /var/shm         shm         defaults    0    0
```

La ligne ci-dessus à cause d'un changement dans le noyau de la gestion de la mémoire partagée.

Périphériques USB

Ensuite, si vous voulez utiliser l'USB :

```
mkdir /dev/input
mknod /dev/input/mouse0 c 13 32
```

Les commandes ci-dessus permettent d'activer l'utilisation de la souris.

```
mknod /dev/input/event0 c 13 64
mknod /dev/input/event1 c 13 65
mknod /dev/input/event2 c 13 66
mknod /dev/input/event3 c 13 67
```

Et ces commandes pour prendre en compte le stylet : déplacement, pression, boutons et gomme.

Paramétrage des modules

Éditez le fichier /etc/modules.conf et ajoutez la ligne suivante :

```
alias char-major-13 usbcore
```

Ce que j'ai fait ensuite

Ensuite j'ai ajouté les commandes suivantes dans le fichier /etc/rc.d/boot.local (rappelez-vous, j'utilise une distribution SuSE, mais vous devriez trouver l'équivalent pour RedHat dans /etc/rc.d/init.d/boot.local ou /etc/rc.d/init.d/rc.local).

```
/sbin/modprobe wacom
/sbin/modprobe mousedev
/sbin/modprobe evdev
```

Ceci, afin d'être sûr que les modules se chargeront au moment du démarrage.

Et enfin

Redémarrage ;)

Regardez les messages de démarrage :

```
dmesg | less
```

et vérifiez qu'il n'y ait aucune erreur relative à l'USB. Cela se pourrait si vous avez choisi UHCI et que votre système est utilise l'OHCI.

Voici les messages qui apparaissent sur mon système :

```
.....
.....
usb.c: registered new driver usbdevfs
usb.c: registered new driver hub
usb-uhci.c: $Revision: 1.9 $ time 17:54:28 May 8 2000
usb-uhci.c: High bandwidth mode enabled
usb-uhci.c: Intel USB controller: setting latency timer to 0
usb-uhci.c: USB UHCI at I/O 0xffe0, IRQ 11
usb-uhci.c: Detected 2 ports
usb.c: new USB bus registered, assigned bus number 1
usb.c: USB new device connect, assigned device number 1
usb.c: kmalloc IF c3fcd2e0, numif 1
usb.c: new device strings: Mfr=0, Product=2, SerialNumber=1
usb.c: USB device number 1 default language ID 0x0
Product: USB UHCI Root Hub
```

```
SerialNumber: ffe0
hub.c: USB hub found
hub.c: 2 ports detected
hub.c: ganged power switching
hub.c: standalone hub
hub.c: global over-current protection
hub.c: power on to power good time: 2ms
hub.c: hub controller current requirement: 0mA
hub.c: port 1 is removable
hub.c: port 2 is removable
hub.c: local power source is good
hub.c: no over-current condition exists
hub.c: enabling power on all ports
usb.c: hub driver claimed interface c3fcd2e0
.....
.....
hub.c: port 1 connection change
usb.c: USB new device connect, assigned device number 2
usb.c: kmalloc IF c3fcd620, numif 1
usb.c: skipped 1 class/vendor specific interface descriptors
usb.c: new device strings: Mfr=1, Product=2, SerialNumber=0
usb.c: USB device number 2 default language ID 0x409
Manufacturer: WACOM
Product: ET-0405-UV1.1-1
usb.c: unhandled interfaces on device
usb.c: This device is not recognized by any installed USB driver.
Length                          = 18
DescriptorType                  = 01
USB version                     = 1.00
Vendor:Product                  = 056a:0010
MaxPacketSize0                  = 0
NumConfigurations               = 1
Device Version                  = 1.11
Device Class:SubClass:Protocol  = 00:00:00
Per-interface classes
Configuration:
.....
.....
```

Lorsque le noyau dit « *This device is not recognized by any installed USB driver (ce périphérique n'est reconnu par aucun pilote USB installé)* », c'est parce que, au moment du démarrage, le module wacom n'est pas encore chargé. Mais comme vous pouvez le voir, toutes les interfaces et bus USB sont reconnus. Aussi, si votre système ne fonctionne pas et n'affiche aucun message de ce genre, vérifiez si vous n'avez pas oublié une option USB (l'un des éléments que j'ai indiqué comme devant figurer dans le noyau et que vous avez choisi comme module : si c'est le cas, dans le fichier `/etc/rc.d/init.d/boot.local`, ajoutez une commande modprobe *avant* les 3 lignes que j'ai indiquées, par exemple :

```
/sbin/modprobe usb-uhci
```

Voici les messages correspondant au chargement du module wacom :

```
.....
.....
usb.c: registered new driver wacom
input0: Wacom Graphire on usb2
usb.c: wacom driver claimed interface c3fcd620
mouse0: PS/2 mouse device for input0
mice: PS/2 mouse device common for all mice
event0: Event device for input0
.....
.....
```

Vérifiez tout et réessayez jusqu'à ce que vous obteniez tous ces messages (ou quelque chose qui corresponde).

La console Linux

Maintenant que votre USB est activé, il est temps de tester tout cela avec gpm :

Si gpm était activé au moment du démarrage, vous allez devoir le tuer :

```
gpm -k
```

Ensuite, essayez votre tablette Wacom :

```
gpm -tIMPS/2 -m /dev/input/mouse0
```

Déplacez le stylet, ou votre souris et TADA ... le curseur se déplace sur l'écran.

XWindow

Avoir 2 souris sur le même système

Avez-vous jamais rêvé d'avoir 2 souris sur votre système ? Si la réponse est oui, faites ce qui suit.

Éditez le fichier `/etc/XF86Config`.

La section modules doit ressembler à ceci :

```
Section "Module"
Load "xf86Wacom.so"
EndSection
```

Vérifiez la section XInput :

```
Section "XInput"
 SubSection "Mouse"

  DeviceName "TabletteWacom" ❶
  Protocol "IMPS/2" ❷
  Port "/dev/input/mouse0"
  Buttons 5
  ZAxisMapping 4 5
  AlwaysCore  ❸

 EndSubSection
EndSection
```

❶ Ou le nom de votre choix.

❷ Je ne suis pas arrivé à faire reconnaître la molette par le protocole imwheel, si vous y arrivez, envoyez-moi un courrier électronique.

❸ `AlwaysCore` veut dire que la souris Wacom est toujours considérée comme pointeur principal. Autrement dit, que déplacer la souris Wacom ou la souris normale revient au même.

Et voilà, la souris Wacom (ou le stylet) est maintenant considérée comme un second pointeur, mais le premier fonctionne toujours.

La réalité

De retour à la section XInput, après les paramètres pour la souris, nous trouvons les réglages intimes de la tablette Wacom :

```
SubSection "WacomStylus"

  Port "/dev/input/event0" ❶
  DeviceName "WacomStylus" ❷
  Mode Absolute ❸
  #Suppress 6
  DebugLevel 10 ❹
  #TiltMode
  #HistorySize 200
  #AlwaysCore
  #KeepShape

  #TopX 20412                                    ❺
  #TopY 14730                                    ❺
  #BottomX 0                                     ❺
  #BottomY 0                                     ❺

EndSubSection

SubSection "WacomEraser"

  DeviceName "/dev/input/event0"
  DeviceName "WacomEraser"
  Mode Absolute

EndSubSection

SubSection "WacomCursor"

  Port "/dev/input/event0"
  DeviceName "GraphireMouse"
  Mode Absolute

EndSubSection
```

❶ Le périphérique d'événements USB.

❷ Ou le nom de votre choix.

❸ C'est-à-dire que la surface de la tablette est équivalente à celle de l'écran.

❹ Une fois que tout fonctionne, mettre cette variable à zéro.

❺ Ces 4 lignes permettent un ajustement fin des coordonnées pour correspondre à l'écran.

Le nouveau module

Le module xf86Wacom.so n'est pas spécifiquement écrit pour les tablettes USB Wacom et il arrive qu'il ne fonctionne pas correctement avec ces tablettes. Un nouveau pilote a été écrit par Matsumura Namihiko <`po TIRET jp CHEZ geocities POINT co POINT jp`> et nommé xf86WacomUSB (son code source est disponible à la même adresse que ce petit guide). Cependant, afin de pouvoir le compiler, vous aurez besoin du code source du noyau et du code source complet de XFree86 (j'ai réussi à le compiler avec XFree86 version 3.3.5, mais cela devrait marcher avec la version 3.3.6).

Ce pilote ne prend pas en charge toutes les possibilités de la tablette et se plante si vous lui passez le paramètre `AlwaysCore`.

Pour corriger cela, je reste en contact avec Frédéric Lepied afin qu'il modifie le module original afin qu'il prenne en compte toutes les possibilités de la tablette.

Nouveau

Sur le site de Frédéric Lepied vous trouverez un lien vers le nouveau pilote USB qui fonctionne parfaitement bien, mais exclusivement avec la possibilité de faire fonctionner votre tablette en souris : c'est-à-dire soit vous utilisez votre tablette avec Le Gimp pour dessiner, soit vous l'utilisez comme une souris XFree, mais pas les deux en même temps. Je vais me renseigner plus avant et mettre ce document à jour en conséquence.

Pour aller plus loin

Pour plus d'information sur les tablettes Wacom en général, essayez le formidable guide pratique de Stephan Runkel <`runkel CHEZ runkeledv POINT de`> : http://www.traduc.org/docs/howto/lecture/Wacom-Tablet-HOWTO.html

En ce qui concerne le module xf86Wacom.so, vérifiez chez Frédéric Lepied <`lepied CHEZ xfree86 POINT org`> : http://www.teaser.fr/~flepied/.

xf86WacomUSB.so, disponible sur ftp.aros.org/pub (lien mort) .

[1 [p 3]] Il s'agit d'adaptations au noyau 2.2 des pilotes USB destinés au noyau 2.4.