

Multi Écrans pour TX HOWTO

Solution basée sur le projet TX

Benoît NICQ <benoit.nicq@odys-solutions.net>

pour Odys

\$Date: 2003/07/08 07:50:03 \$

Ce document détaille l'installation d'un Terminal X11 **TX** à plusieurs écrans. L'ensemble des écrans du **TX** forme un seul bureau.

1 Introduction

Il est tout à fait possible de faire fonctionner un terminal X11 avec plusieurs écrans pour ne disposer que d'un seul bureau plus grand. La seule limite étant le nombre de cartes qu'il est possible de mettre dans un PC, et la principale difficulté est de trouver aujourd'hui des cartes vidéo **PCI** acceptant de fonctionner ensemble.

Le PC utilisé pour l'exemple de ce document est un Pentium 200 Mhz, 128Mo de RAM avec une carte video ATI intégrée à la carte mère et deux cartes Matrox Millénium PCI. XFree86 en mode `jj` multi-écrans `ii` est gourmand en mémoire, 64 Mo de RAM est un minimum. Le bureau fait 3456x864 pixels (3 x 1152x864).



Figure 1: **TX** multi-écrans

2 Installer un nouveau système de fichiers pour le TX

Pour commencer l'installation, il est nécessaire d'installer un nouveau système de fichier pour le **TX** `jj` multi-écrans `ii` sans pour autant détruire le ou les système(s) des terminaux existants. Si vous n'avez jamais installer de **TX** reportez-vous aux documentations à l'adresse *Documentation TX* <<http://tx.odys-solutions.net/doc.html>> .

Pour ne pas écraser les systèmes de fichiers des **TX** existants il faut modifier le fichier `/etc/tx-master.conf` et changer le chemin du système de fichiers :

remplacer

```
tx_base="/var/lib/diskless/tx"
```

par

```
tx_base="/var/lib/diskless/multi"
```

Ensuite exécuter la commande *mk-tx-root*. Pour terminer l'installation voir la documentation complète à l'adresse *Documentation TX* <<http://tx.odys-solutions.net/doc.html>> .

3 Configurer X11

Pour le mode `;;` multi écrans `;;` il est nécessaire de configurer le **TX** pour fonctionner avec la version Xfree86 **4.x.x.** En effet les versions antérieures à la version 4.0 ne supportent pas ce mode.

Il faut avant tout définir quelles sont les cartes reconnues par le système et sur quels bus elles sont connectées. Une fois que le TX a démarré (inutile pour le moment de se préoccuper de la configuration de X) appuyer sur les touches **Ctrl+Alt+F4** de façon à passer sur la console du terminal. Appuyer sur la touche **Entrée** pour l'activer et exécuter la commande suivante (Attention le clavier est configuré en QWERTY!) :

```
# /usr/X11R6/bin/XFree86 -scanpci
```

Voici ce que ça donne pour l'exemple:

```
Probing for PCI devices (Bus:Device:Function)

(0:13:0) Matrox Millenium II using a Matrox MGA 1064SG
(0:14:0) Matrox unknown card (0x051a) using a Matrox MGA 1064SG
(1:0:0) ATI card using a ATI Mach64 GB
```

Une fois les informations sur les cartes vidéo récupérées il faut modifier le script */var/lib/diskless/multi/etc/tx/XF86Config-4.sh* qui génère le fichier de configuration de **XFree86**

Le script *XF86Config-4.sh* remplace des variables par les valeurs correspondantes envoyées par le serveur **DHCP**. C'est un remplacement de variables par substitution. Pour indiquer une valeur en `;;` dure `;;` il suffit de remplacer dans le script la variable par la valeur souhaitée. Ce script n'est pas fait à l'origine pour gérer plusieurs écrans. Il faut envisager l'installation d'un système de fichiers par **TX** à moins que les PCs aient la même configuration matérielle.

Le plus simple est de télécharger le script modifié qui sert pour l'exemple à l'adresse *XF86Config-4.sh* </doc/tx_multi_echans/XF86Config-4.sh> et de le comparer au script original.

Pour plus d'information sur le fonctionnement en mode `;;` multi-écrans `;;` sous GNU/Linux voir la page *Multi-Head ou Utiliser plusieurs écrans et/ou cartes graphiques* <<http://lea-linux.org/xwindow/multihead.php3>>

Tout d'abord il faut supprimer la *Section "Device"* existante et créer une nouvelle *Section "Device"* pour chaque carte suivant le modèle décrit dans l'exemple.

Exemple:

```
#Section "Device"
#       Identifier      "Generic Video Card"
```

```

#
# Driver          $X_DRIVER
#
# Option          "UseFBDev"          "true"
#EndSection

Section "Device"
    Identifier     "Video_Card 1"
    Driver         "mga"
    BusID         "PCI:0:13:0"
EndSection

Section "Device"
    Identifier     "Video_Card 3"
    Driver         "mga"
    BusID         "PCI:0:14:0"
EndSection

Section "Device"
    Identifier     "Video_Card 2"
    Driver         "ati"
    BusID         "PCI:1:0:0"
EndSection

```

Pour notre exemple, les trois écrans sont identiques. Si les écrans ont des caractéristiques différentes il faut créer autant de *Section "Monitor"* que de modèles d'écran et y inscrire leurs caractéristiques.

```

Section "Monitor"
    Identifier     "Generic Monitor"

    $X_HORIZONTAL_SYNC
    $X_VERTICAL_REFRESH
    $X_MODELINE

    Option        "DPMS"
EndSection

```

Il est nécessaire de créer autant de *Section "Screen"* que de cartes vidéo utilisées. Les cartes video peuvent utiliser des résolutions différentes en fonction de leur capacité en mémoire et l'écran qui y est connecté. Par-contre il est primordial d'utiliser la même profondeur de couleur (`DefaultColorDepth`) pour toutes les cartes.

```

Section "Screen"
    Identifier     "Screen 1"
    Device         "Video_Card 1"
    Monitor        "Generic Monitor"
    $X_COLOR_DEPTH
    SubSection "Display"
        Depth      1
        $X_RESOLUTION
    EndSubSection
    SubSection "Display"
        Depth      4
        $X_RESOLUTION

```

```
EndSubSection
SubSection "Display"
    Depth        8
    $X_RESOLUTION
EndSubSection
SubSection "Display"
    Depth        15
    $X_RESOLUTION
EndSubSection
SubSection "Display"
    Depth        16
    $X_RESOLUTION
EndSubSection
SubSection "Display"
    Depth        24
    $X_RESOLUTION
EndSubSection
EndSection

Section "Screen"
    Identifier    "Screen 2"
    Device        "Video_Card 2"
    Monitor       "Generic Monitor"
    $X_COLOR_DEPTH
    SubSection "Display"
        Depth        1
        $X_RESOLUTION
    EndSubSection
    SubSection "Display"
        Depth        4
        $X_RESOLUTION
    EndSubSection
    SubSection "Display"
        Depth        8
        $X_RESOLUTION
    EndSubSection
    SubSection "Display"
        Depth        15
        $X_RESOLUTION
    EndSubSection
    SubSection "Display"
        Depth        16
        $X_RESOLUTION
    EndSubSection
    SubSection "Display"
        Depth        24
        $X_RESOLUTION
    EndSubSection
EndSection

Section "Screen"
    Identifier    "Screen 3"
    Device        "Video_Card 3"
    Monitor       "Generic Monitor"
    $X_COLOR_DEPTH
```

```

SubSection "Display"
    Depth      1
    $X_RESOLUTION
EndSubSection
SubSection "Display"
    Depth      4
    $X_RESOLUTION
EndSubSection
SubSection "Display"
    Depth      8
    $X_RESOLUTION
EndSubSection
SubSection "Display"
    Depth      15
    $X_RESOLUTION
EndSubSection
SubSection "Display"
    Depth      16
    $X_RESOLUTION
EndSubSection
SubSection "Display"
    Depth      24
    $X_RESOLUTION
EndSubSection
EndSection

```

Il faut décrire la disposition des écrans. Les options principales sont :

- **RightOf** écran à droite
- **LeftOf** écran à gauche
- **Below** écran au-dessous
- **Above** écran au-dessus

Pour plus d'information : *man XF86Config-4*

```

Section "ServerLayout"
    Identifier      "Default Layout"

    Screen          "Screen 1"
    Screen          "Screen 2" RightOf "Screen 1"
    Screen          "Screen 3" RightOf "Screen 2"

    InputDevice    "Generic Keyboard"
    InputDevice    "Configured Mouse"
    # InputDevice  "Generic Mouse"
EndSection

```

Il ne reste plus qu'à modifier le fichier */var/lib/diskless/multi/etc/tx/rcS* pour que l'option Xinerama soit prise en compte. Cette option est nécessaire pour avoir un bureau qui utilise l'ensemble des écrans.

Il faut remplacer:

```
X_OPT="-nolisten local -nolock "
```

par

```
X_OPT="-nolisten local -nolock +xinerama"
```

Et voilà c'est fini!