

sélecteur de session

sur la plupart des distributions GNU/Linux, l'identification, la connexion et le lancement de la session graphiques sont des processus contrôlés par une application de type GDM, KDM ou Slim. ce wiki vous permettra d'installer plusieurs gestionnaires de fenêtres et de choisir lequel choisir à la connexion.

Sommaire

sélecteur de session.....	1
CLI selector.....	1
GUI selector.....	3
session_selector.sh.....	4
dmenu-selector.....	8

CLI selector

`~/.xinitrc` et `~/.(z)profile` : ceux deux fichiers suffisent à obtenir un **sélecteur de session en console**, sans passer par gdm ou slim.

la méthode est simple: créer autant de `~/.xinitrc*` que de sessions désirées, et inclure dans `(z)profile` un sélecteur de fichiers:

à la fin de votre `~/.(z)profile`, rajouter ces quelques lignes (à adapter selon votre configuration):

```
# Si on est dans une console, et qu'aucune
# instance de X n'est détecté, demander si
# startx doit être lancé et dans quel wms.
if [[ -t 0 && $(tty) =~ /dev/tty ]] && ! pgrep -u $USER startx &> /dev/null; then
  echo ""
  echo ""
  echo "`uname -a`"
  echo ""
  echo " 1/dwm(default) - 2/fluxbox - 3/compiz - 4/wmfs - 5/noX"
  read choix
  case $choix in
    1)
      startx
      ;;
    2)
      startx ~/.xinitrc_flux
      ;;
    3)
      startx ~/.xinitrc_compiz
      ;;
    4)
      startx ~/.xinitrc_wmfs
      ;;
    5)
      clear
      echo "if there is a shell ... there is a way"
```

```

        echo ""
        ;;
    *)
        startx
        ;;
esac
fi

```

dans notre exemple, dwm est lancé depuis ~/.xinitrc pour obtenir une session "par défaut". les autres sessions ont leurs propres ~/.xinitrc_<nom_de_la_session>. pour l'exemple, voici mon ~/.xinitrc par défaut:

```

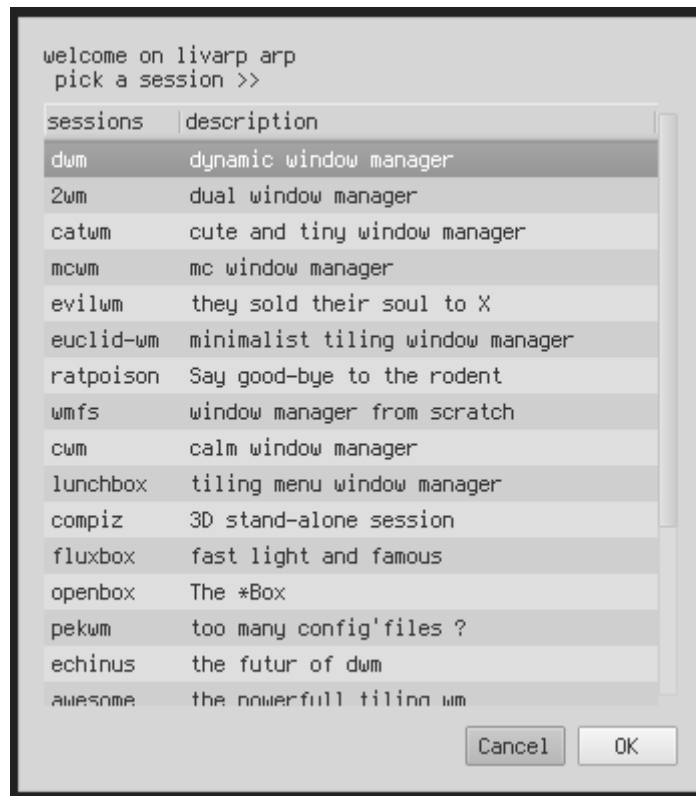
#!/bin/bash
# ~/.xinitrc by arpinux 2013
#####
## D-Bus ##-----
if which dbus-launch >/dev/null && test -z "$DBUS_SESSION_BUS_ADDRESS"; then
    eval "$(dbus-launch --sh-syntax --exit-with-session)"
fi
## clean-up conky ##-----
killall conky &
killall conkyd &
## setup automount -----
devmon --info-on-mount --exec-on-drive 'rox %d' &
## read ~/.Xresources file ##-----
xrdb -merge ~/.Xresources
## enable snap font ##-----
xset fp+ /usr/share/fonts/artwiz-latin1/
xset fp rehash
## launch terminal daemon ##-----
urxvtd -q -f -o
## set cursor ##-----
xsetroot -cursor_name left_ptr
## load ibm-fr keyboard ##-----
setxkbmap fr -variant oss
## set ctrl_alt_bksp as session-killer ##-----
setxkbmap -option terminate:ctrl_alt_bksp
## replace caps_lock by super key ##-----
xmodmap ~/.Xmodmap
## launch screensaver ##-----
xscreensaver-command -exit
xscreensaver -no-splash &
## startup wall ##-----
feh --bg-fill /home/arp/pics/walls/devArp_dark_bg.png
## set statusbar -----
sleep 3s && conkyd -q -c .conkyrc_dwm | while true; read line; do xsetroot -name "$line"; done &
## launch compiled dwm-session #####
#exec ck-launch-session /usr/local/bin/dwm
## launch legacy dwm #####
#exec ck-launch-session /usr/bin/dwm
## launch dwm with log file #####
while true; do exec ck-launch-session /usr/local/bin/dwm 2> ~/.dwm.log; done

```

ainsi, après votre login/mot de passe, votre tty vous proposera les différentes sessions accessibles sur votre système. bien sûr, il faudra éditer ce fichier et créer un ~/.xinitrc* si vous désirez ajouter/enlever une session.

GUI selector

le **session-selector** est un script écrit à l'origine pour [livarp](#): il permet de choisir sa session graphique juste après le login en **tty**. il dépend du paquet *zenity*.



pour que le *session-selector* se lance automatiquement au login, il faut au préalable éditer votre fichier `~/.profile` (`~/.zprofile` si vous utilisez `zsh`) et y ajouter les lignes suivantes en fin de fichier:

```
# startx automatique
if [ -z "$DISPLAY" ] && [ $(tty) == /dev/tty1 ]; then
    startx
fi
```

ainsi, *startx* se lancera juste après votre login en `tty1`. la commande **startx** lit le fichier `~/.xinitrc`. nous allons l'éditer et en profiter pour y rajouter les applications/réglages communs à vos sessions, définir un écran d'accueil et enfin lancer *session-selector*.

```
#!/bin/bash
#####
# ~/.xinitrc by arpinux #
#####

## D-Bus ##
if which dbus-launch >/dev/null && test -z "$DBUS_SESSION_BUS_ADDRESS"; then
    eval "$(dbus-launch --sh-syntax --exit-with-session)"
fi

## read ~/.Xresources file
xrdb -merge ~/.Xresources

## launch terminal daemon ##
```

```

urxvtd -q -f -o

## set cursor
xsetroot -cursor_name left_ptr

## set ctrl_alt_bksp as session-killer
setxkbmap -option terminate:ctrl_alt_bksp

## startup-wall
feh --no-xinerama --bg-center ~/bin/.startupwall-d.png

## clean-up conky daemon
if [ "$(pidof conkyd)" ]; then
    killall conkyd
fi
if [ "$(pidof conky)" ]; then
    killall conky
fi
#-----
exec ck-launch-session $HOME/bin/session_selector.sh

```

- l'écran d'accueil est affiché par la commande **feh**, si vous désirez afficher votre fond d'écran de travail, remplacer cette ligne par:

```
nitrogen --restore
```

- la section **conky** est ici car lorsque vous quittez votre session graphique, vous ne vous déconnectez pas de votre session utilisateur. les processus ne dépendant pas de X (comme conky) continuent de tourner, cette section est là pour éviter le cumul des processus ou les processus zombies. **conkyd** est une copie de conky dont je me sert pour installer en même temps conky et conky-cli (pour la statusbar de dwm et autres).

session_selector.sh

la dernière commande du ~/.xinitrc lance le session-selector: dans livarp, ce script renvoi à différents scripts de type 'autostart', un pour chaque session.

je vous livre ici un session-selector qui se suffit à lui-même (à rendre exécutable avec un "chmod +x session_selector.sh"):

```

#!/bin/bash
#####
# simple session-selector #
# demande zenity          #
#####
# qui est là?
HUMAN=`w -h | tail -n1 | awk '{print $1}'`
HOST=`hostname`
# liste des sessions
ACTION=`zenity --width=340 --height=385 --list\
    --title "SelectWm" --text "bienvenue sur $HOST, $HUMAN \n choisissez
une session >>"\
    --column "sessions" --column "description"\
    "dwm" "dynamic window manager"\
    "2wm" "dual window manager"\
    "catwm" "cute and tiny window manager"\

```

```

"mcwm" "mc window manager"\
"evilwm" "they sold their soul to X"\
"euclid-wm" "minimalist tiling window manager"\
"ratpoison" "Say good-bye to the rodent"\
"wmfs" "window manager from scratch"\
"cwmm" "calm window manager"\
"lunchbox" "tiling menu window manager"\
"compiz" "3D stand-alone session"\
"fluxbox" "fast light and famous"\
"openbox" "The *Box"\
"pekwm" "too many config'files ?"\
"echinus" "the fork of dwm"\
"restart" "restart your computer"\
"halt" "shutdown your computer"

# lancement des sessions
if [ -n "${ACTION}" ];then
  case $ACTION in
    dwm)
      # check mail
      sleep 8 && claws-mail --online &
      #lancement du panel
      sleep 1 && fbpanel &
      # fond d'écran
      feh --no-xinerama --bg-scale ~/.arp_setups/bg_debian.png
      # statusbar
      conkyd | while true; read line; do xsetroot -name "$line"; done &
      # lancement de dwm avec fichier de log
      while true; do
        /usr/local/bin/dwm 2> ~/.dwm.log
      done
      ;;
    2wm)
      # lancement du terminal
      sleep 4 && roxterm &
      # lancement de 2wm avec fichier de log
      while true; do
        /usr/local/bin/2wm 2> ~/.2wm.log
      done
      ;;
    catwm)
      # lancement du terminal
      sleep 4 && roxterm &
      # lancement de catwm avec un fichier de log
      while true; do
        /usr/bin/catwm 2> ~/.catwm.log
      done
      ;;
    mcwm)
      # lancement du monitoring
      sleep 2 && conkyd -c ~/.conkyrc_mcwm
      # lancement du terminal
      sleep 4 && urxvtc &
      # lancement de mcwm avec dmenu comme commande au clic
      while true; do
        /usr/local/bin/mcwm -t dmenu-bind.sh
      done
      ;;
    evilwm)

```

```

# check mail
sleep 8 && claws-mail --online &
# fond d'écran
feh --no-xinerama --bg-scale ~/pics/evil\ artwork/evilwm_evildark.png
# raccourcis clavier
sleep 3s && xbindkeys
# systray indépendant
stalonetray &
# monitoring
sleep 5s && conkyd -c ~/.conkyrc_evil
# lancement d'evilwm avec urxvtc par défaut avec un fichier de log
while true; do
    evilwm -snap 10 -term urxvtc -nosolddrag 2> ~/.evilwm.log
done
;;
euclid-wm)
# lancement du terminal
sleep 4 && urxvtc &
# lancement d'euclid depuis le .desktop
exec start-euclid
;;
ratpoison)
# lancement du terminal
sleep 4 && urxvtc &
# lancement de ratpoison depuis le .desktop
exec ratpoison
;;
wmfs)
# check mail
sleep 8 && claws-mail --online &
# statusbar
conkyd -c ~/.conkyrc_wmfs | while true; read line; do wmfs -s 0
" `date` "; wmfs -s 1 "$line"; done &
# lancement de wmfs
exec wmfs
;;
cwm)
# lancement du panel
fbpanel &
# lancement de cwm depuis le dossier de bin perso
exec /home/arp/bin/cwm
;;
lunchbox)
# lancement du terminal
sleep 4 && urxvtc &
# lancement de lunchbox
exec lunchbox
;;
compiz)
# check mail
sleep 8 && claws-mail --online &
# lancement du panel
sleep 2 && fbpanel &
# fond d'écran
nitrogen --restore
# monitoring
sleep 2 && conkyd -c ~/.arp_setups/fusion/conkyrc
# lancement de compiz en stand-alone

```

```

    exec compiz ccp
    ;;
fluxbox)
    # lancement de fluxbox depuis le .desktop
    exec /usr/bin/startfluxbox
    ;;
openbox)
    # lancement d'openbox depuis le .desktop
    exec /usr/bin/openbox-session
    ;;
pekwm)
    # check mail
    sleep 8 && claws-mail --online &
    # fond d'écran
    feh --no-xinerama --bg-scale ~/pics/livarp\ artwork/livarp_pekwm.png
    # lancement du panel
    sleep 1 && fbpanel -p fb_mini_pek &
    # monitoring
    sleep 2 && tint2 -c ~/.config/tint2/tint2rc_pekwm &
    sleep 3 && conkyd -c ~/.conkyrc_pekwm &
    # lancement de pekwm depuis le .desktop
    exec pekwm
    ;;
echinus)
    # check mail
    sleep 8 && claws-mail --online &
    # fond d'écran
    feh --no-xinerama --bg-scale ~/.echinus/bg.png
    # lancement du panel
    sleep 1 && tint2 -c ~/.config/tint2/tint2rc_echinus &
    # lancmeent des barres d'infos dzen2
    sleep 2 && ~/bin/dzen_menu.sh
    sleep 1 && ~/bin/dzenbar.sh
    ~/bin/dzenbar_mocp.sh
    ~/bin/dzen_wall.sh &
    ~/bin/top_cpu_bar.sh
    # lancement de la session echinus
    exec /usr/local/bin/echinus
    ;;
edit)
    geany -s /home/$USER/bin/session_selector.sh
    exec ck-launch-session /home/$USER/bin/session_selector.sh
restart)
    # redémarrer l'ordinateur
    sudo shutdown -r now
    ;;
halt)
    # éteindre l'ordinateur
    sudo shutdown -h now
    ;;
esac
fi

```

bien sûr, vous n'êtes pas obligé de tester autant de session que moi, mais vous pouvez vous inspirer de ce script pour lancer vos sessions favorites. l'intérêt de ce script est qu'il s'intègre parfaitement à votre thème puisque zenity utilise gtk. vous pouvez également choisir facilement votre écran d'accueil en modifiant une ligne en tant que simple utilisateur.

dmenu-selector

une variante utilisant [dmenu](#) afin de sélectionner la session: le dmenu-start.sh:

```
#!/bin/bash
# a simple session-selector
#####
# session list
# -----
choice=`echo -e "01: vtwm\n02: dwm\n03: ratpoison\n04: wmfs\n05:
dwm_reloaded\n06: spectrwm\n07: evilwm\n08: openbox\n09: pekwm\n10: awesome\n11:
fluxbox\n12: scripts de lancement\n13: eteindre" | dmenu -fn "snap" -nb
"#222222" -nf "#7D7D7D" -sb "#005885" -sf "#D7D7D7" -p "choisir votre session:" |
cut -d ':' -f 1`
# session launchers
# -----
case $choice in
  01) exec $HOME/bin/start/vtwm_start.sh ;;
  02) exec /usr/bin/dwm ;;
  03) exec $HOME/bin/start/ratpoison_start.sh ;;
  04) exec $HOME/bin/start/wmfs_start.sh ;;
  05) exec $HOME/bin/start/dwm_start.sh ;;
  06) exec $HOME/bin/start/spectrwm_start.sh ;;
  07) exec $HOME/bin/start/evilwm_start.sh ;;
  08) exec /usr/bin/openbox-session ;;
  09) exec $HOME/bin/start/pekwm_start.sh ;;
  10) exec $HOME/bin/start/awesome_start.sh ;;
  11) exec $HOME/bin/start/fluxbox_start.sh ;;
  12) geany -s bin/start/*_start.sh .config/openbox/autostart .xinitrc; dmenu-
start.sh ;;
  13) sudo shutdown -h now ;;
  *) dmenu-start.sh ;;
esac
exit 0
```

ce script renvoi aux scripts de lancement respectifs des wm's utilisés.

vous pouvez désormais tester autant de sessions que vous désirez (à installer dans votre ~/bin) sans passer par le super-utilisateur, et sans risquer de dommages sur votre système.

contributeur:[arpinux](#)