

EVILWM

evilwm est le window-manager en floating **minimal** par excellence ! pas de panel, pas de menu, pas de décorations de fenêtres hormis une bordure de 1px, pas de raccourcis clavier hormis un terminal avec [Ctrl+Alt+Return], bref, rien pour perturber le boulot ... et tout pour faire sa configuration vraiment personnelle :).

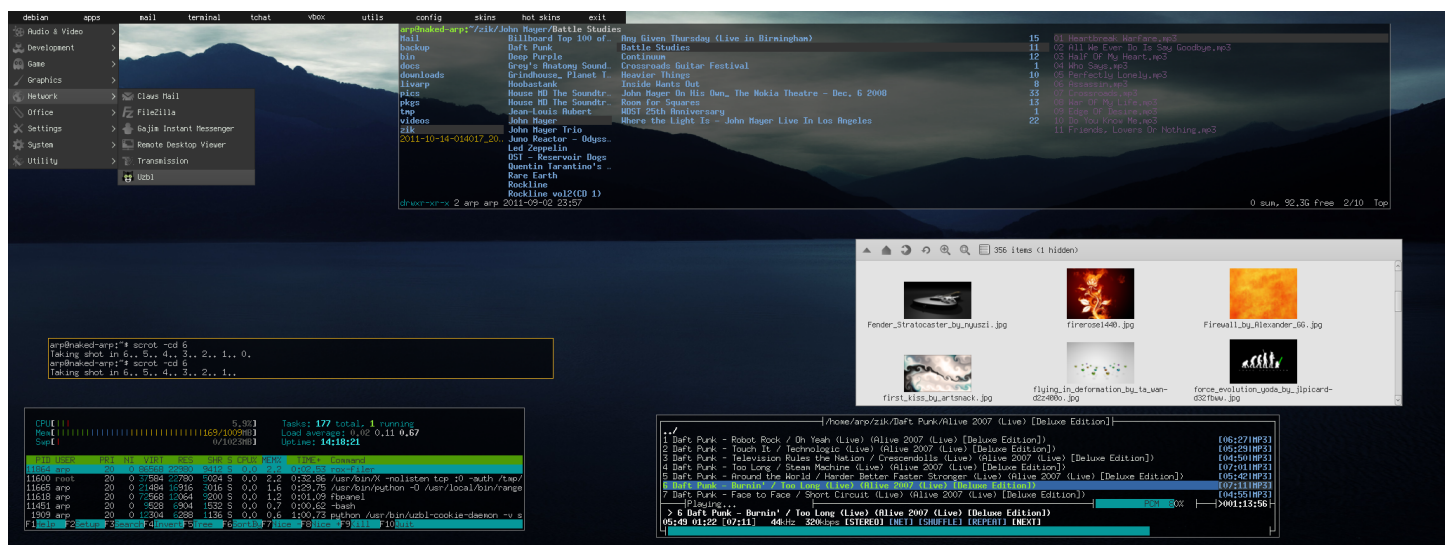
evilwm est rapide, *très* rapide, léger, et stable. basée à l'origine sur [aewm](#), le projet débute en 2000 avec **Ciaran Ancomb**, qui continue à maintenir le [site principal](#) à jour. la dernière version est evilwm-1.1.0 du 3 juillet 2011.

si vous désirez tester evilwm en live, allez faire un tour du côté du [livarp](#) ;) la version 0.3 a une session evilwm préconfigurée et documentée...

Sommaire

EVILWM.....	1
screenshot.....	1
installation.....	2
configuration.....	2
avec ~/.xinitrc.....	2
avec ~/.evilwmrc.....	3
utilisation.....	4
commandes de base.....	4
raccourcis clavier.....	5
gestion de la session.....	6
conclusion.....	6

screenshot



installation

si vous avez le bonheur d'utiliser **Debian**, un paquet est disponibles dans les dépôts:

```
# apt-get install evilwm
```

mais vous pouvez également compiler evilwm depuis les **sources**:

- soit en obtenant l'archive

```
$ wget http://www.6809.org.uk/evilwm/evilwm-1.1.0.tar.gz
```

- soit en passant par git

```
$ git clone http://www.6809.org.uk/repos/evilwm.git
```

la procédure de compilation reste classique (dépend de libpango-dev et libxrandr-dev):

```
$ cd ./evilwm_VERSION
$ make
# make install
```

vous voilà maintenant prêt à lancer votre session evilwm.

configuration

evilwm ne nécessite aucun fichier de configuration pour sa session. les options peuvent simplement être passées en arguments lors du lancement. comme il n'a pas de fichier "autostart", vos applications à lancer au démarrage doivent figurer dans votre ~/.xinitrc ou votre script de lancement de session. evilwm ne s'occupe pas de votre curseur non plus, il faudra le spécifier, idem pour le fond d'écran, ou les raccourcis clavier. evilwm ne prend pas en charge le logout non plus... je vous avais prévenu ... **ultra-minimal** :)

avec ~/.xinitrc

pour ceux qui utilisent gdm, je vous conseille d'utiliser la méthode décrite pour [compiz](#) afin de créer une session.

pour les autres, voici mon ~/.xinitrc et le startup script d'evilwm commenté:

```
#!/bin/bash
#####
# ~/.xinitrc by arpinux 2011 #
#####
## D-Bus ##
if which dbus-launch >/dev/null && test -z "$DBUS_SESSION_BUS_ADDRESS"; then
    eval "$(dbus-launch --sh-syntax --exit-with-session)"
fi
## trackpad ## tapbutton off by default ##
synclient VertTwoFingerScroll=1
synclient HorizTwoFingerScroll=1
synclient TapButton1=0
## dualscreen ## edit if needed ##
xrandr --output LVDS --mode 1024x768 --pos 0x0 --rotate normal --output VGA-0
```

```

--mode 1024x768 --pos 1024x0 --rotate normal
## set wallpaper / color the screen ##
#xsetroot -solid grey20 &
feh --no-xinerama --bg-scale ~/.arp_setups/bg.png ## uncomment to display default
wallpaper ##
#nitrogen --restore & ## uncomment to run your personal wallpaper ##
## launch terminal daemon ##
urxvtd -q- -f -o
## set session-killer
setxkbmap -option terminate:ctrl_alt_bksp
## launch session
exec ck-launch-session $HOME/bin/evilwm-start.sh

```

```

#!/bin/bash
# evilwm startup script by arpinux
## launch mail-checker
sleep 8 && claws-mail --online &
## set shortcuts
sleep 3s && xbindkeys
## launch systray
stalonetray &
## launch conky
sleep 5s && conkyd -c ~/.conkyrc_evil
## launch evilwm in a loop
while true; do
    evilwm -snap 10 -term urxvtd -nosoliddrag 2> ~/.evilwm.log
done

```

ce fichier suffit à lancer votre session evilwm avec une distance d'adhérence de 10px, urxvtd comme terminal par défaut. cependant, vous pouvez utiliser un **fichier de configuration** pour y mettre vos arguments:

avec ~/.evilwmrc

evilwm accepte un fichier de configuration: **~/.evilwmrc** mais notez que ce fichier n'est pas indispensable et que *les arguments passés dans le ~/.xinitrc écrasent ceux du ~/.evilwmrc*. un exemple de fichier evilwmrc (pas le mien, je n'en ai pas)

```

term urxvtd
snap 10
bw 2
fg #cccccc
fc #6699cc
mask1 mod4
bg #999999
fn lime:pixelsize=8

```

petite explication.. les valeurs:

- `display <string>` : sur quel X se lance evilwm. (utiliser seulement si vous en avez plusieurs)
- `term <nom_du_terminal>` : terminal utilisé par défaut lancé par [Ctrl][Alt][Return]
- `fn <fontname>` : police utilisé pour les infos des fenêtres
- `fg <#RRGGBB>` : couleur de la bordure des fenêtres actives
- `fc <#RRGGBB>` : couleur de la bordure des fenêtres fixes
- `bg <#RRGGBB>` : couleur de la bordure des fenêtres inactives
- `bw <n_en_pixel>` : taille de la bordure des fenêtres
- `snap <n_en_pixel>` : distance d'adhérence avec les bords et les autre fenêtres
- `mask1 <modifiers>` : touche de modification 1 [Ctrl][Alt] par défaut
- `mask2 <modifiers>` : touche de modification 2 [Alt] par défaut
- `altmask <modifier>` : touche de modification secondaire [Shift] par défaut
 - liste des <modifiers> admis par evilwm: shift, lock, control, alt, mod1, mod2, mod3, mod4, mod5.
- `app <name/class>` (définis par xprop): spécifier les propriétés des fenêtres citées. les options -g -v et -s qui suivent seront appliqués à cette application.
- `g <geometry>` : taille et position de la fenêtre selon la convention de X
- `v <vdesk>` : designe le bureau virtuel à utiliser
- `s` : fenêtre visible sur tous les bureaux
- `nosoliddrag` : ne dessine pas l'intérieur des fenêtres lors du déplacement
- `V` : version de evilwm

ces valeurs peuvent être passées en arguments dans `~/.xinitrc` précédées de `"-"`

utilisation

evilwm a un comportement simple:

- suivi du focus: le focus suit la souris
- raise on clic: clic pour mettre au premier plan
- floating wm: les fenêtres s'affichent "librement"(pas de tiling) sauf si une propriété a été spécifié dans la commande de lancement ou dans le `~/.evilwmrc`

commandes de base

evilwm est configuré avec quelques raccourcis par défaut. ce tableau reprend ces valeurs, si vous avez modifié les <modifier> avec l'options **-mask**, il faudra adapter.

modifier	touche/bouton	fonctions
[Alt]	clic gauche	place la fenêtre au premier plan, deplace la fenêtre
	clic central	redimensionne la fenêtre
	clic droit	passse la fenêtre en arrière plan
	[Tab]	navigue entre les fenêtres
[Ctrl][Alt]	return	ouvre le terminal par défaut ou celui spécifié avec l'option -term
	i	affiche les informations sur la fenêtre
	=	(des)active la maximisation verticale

	x	(des)active la maximisation complète
	d	affiche/masque les docks, pager ou panels
	f	(des)active la visibilité sur tous les bureaux
	1-8	va au bureau N(1-8). ne fonctionne pas avec tous les claviers
	left,right	va au bureau de gauche/droite
	a	va/revient au/du précédent bureau
	s	switch/permute les écrans
	escape	ferme la fenêtre
	insert	passse la fenêtre en arrière-plan
	h,j,k,l	deplace la fenêtre de 16px. direction: gauche, bas, haut, droite
	y,u,b,n	deplace la fenêtre dans un coin d'écran. direction: top-left, top-right, bottom-left, bottom-right
[Ctrl][Alt][Shift]	h,j,k,l	redimensionne la fenêtre de 16px. direction: gauche, bas, haut, droite
	=	(des)active la maximisation horizontale

raccourcis clavier

L'utilisation de base de **evilwm** se résume à pouvoir ouvrir un terminal et naviguer entre les bureaux ou les terminaux ouverts. Pour ajouter un peu de fonctionnalité, nous allons utiliser **xbindkeys** (dispo dans les dépôts).

on commence par l'installer

```
# apt-get install xbindkeys
```

puis on crée un fichier de configuration par défaut dans ~/.xbindkeysrc

```
$ xbindkeys --default > ~/.xbindkeysrc
```

enfin on édite le fichier créé

```
vim ~/.xbindkeyrc
```

pour y ajouter ses raccourcis. la commande `xbindkeys -k` vous permettra de taper votre combinaison de touche et de déterminer la syntaxe à utiliser dans `xbindkeysrc`.

pour l'exemple, voici mes modifications sur ~/.xbindkeysrc:

```
# For the benefit of emacs users: -*- shell-script -*-
#####
# xbindkeys configuration #
#####
#
# Version: 1.8.3

# arp keybinds
"urxvtc"
  m:0x4 + c:36  ## Control + Return ##
"urxvtc -e ranger"
  m:0x8 + c:27  ## Alt + r ##
```

```
"rox-filer"
 m:0x9 + c:27  ## Alt+Shift + r ##
"uzbl http://start.arpinux.org"
 m:0x8 + c:52  ## Alt + w ##
"firefox"
 m:0x8 + c:41  ## Alt + f ##
"urxvtc -e vim"
 m:0x8 + c:26  ## Alt + e ##
"geany"
 m:0x9 + c:26  ## Alt+Shift + e ##
"urxvtc -e mcabber"
 m:0x8 + c:53  ## Alt + x ##
"compiz-deskmenu"
 m:0x8 + c:65  ## Alt + space ##

#####
# End of xbindkeys configuration #
#####
```

notez la ligne "*compiz-deskmenu*" qui me permet d'avoir un menu personnalisé accessible par les touches [Alt]+[space]. si vous désirez installer ce menu, rendez-vous sur [cette page](#).

il suffit ensuite de lancer xbindkeys depuis votre ~/.xinitrc

```
## set keybindings ##
xbindkeys
```

vous avez désormais une session fonctionnelle de evilwm: un script pour vos applications au démarrage (~/.xinitrc), un fichier de configuration pour vos raccourcis (~/.xbindkeysrc) et un gestionnaire de session (xclock).. vous êtes prêts. :)

gestion de la session

comme dit plus haut, evilwm ne prend pas en charge la gestion de la session. pour quitter evilwm, il faut **killall evilwm** (et encore.. des fois ça freeze). l'autre option est de rétablir la fonction "kill X": [Ctrl]+[Alt]+[BackSpace] pour quitter le serveur X, en ajoutant à votre ~/.xinitrc ou evilwm-start.sh

```
setxkbmap -option terminate:ctrl_alt_bksp
```

conclusion

evilwm est tout à fait adapté à vos besoins si vous utilisez trois applications *navigateur web, lecteur audio/video, editeur de texte* (comme 70% des utilisateurs de pc), mais peut se révéler particulièrement efficace dans un environnement de travail intensif. testez le et vous verrez (15 minutes avec le wiki pour avoir une session propre). j'avoue avoir été réellement surpris par ce gestionnaire de fenêtre qui ne demande aucune connaissance en informatique et se gère assez facilement.

sources: site principal de [evilwm](#)