

Pekwm

- Objet : Pekwm, configuration
- Niveau requis :
[débutant, avisé](#)
- Commentaires : *Un gestionnaire de fenêtres*
- Débutant, à savoir : [Utiliser GNU/Linux en ligne de commande, tout commence là !](#) 😊
- Suivi :
 - Création par [martin_mtl](#) le 13/12/2012
 - Testé par [arpinux](#) le 30/04/2023
- Commentaires sur le forum : [C'est ici](#)¹⁾

Nota :

Ce tuto est une gracieuseté offerte par Thuban !

Merci de ton attention Thuban ! 😊

Présentation

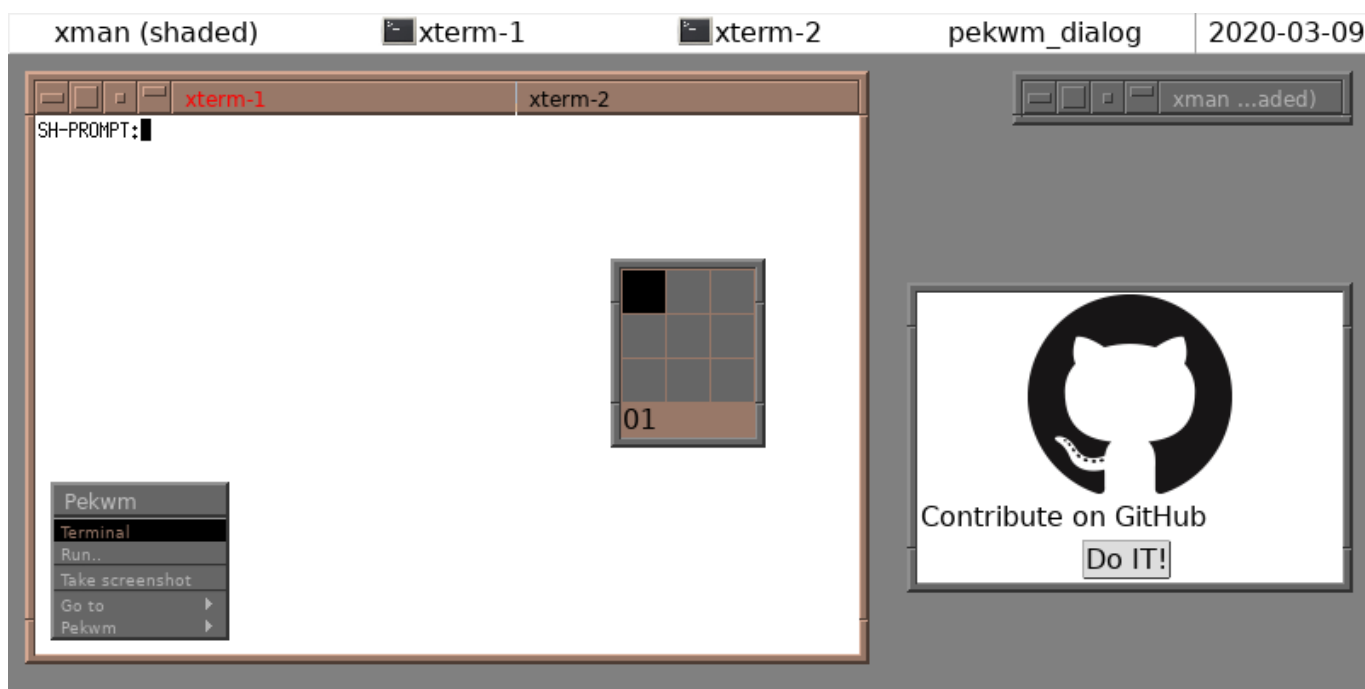
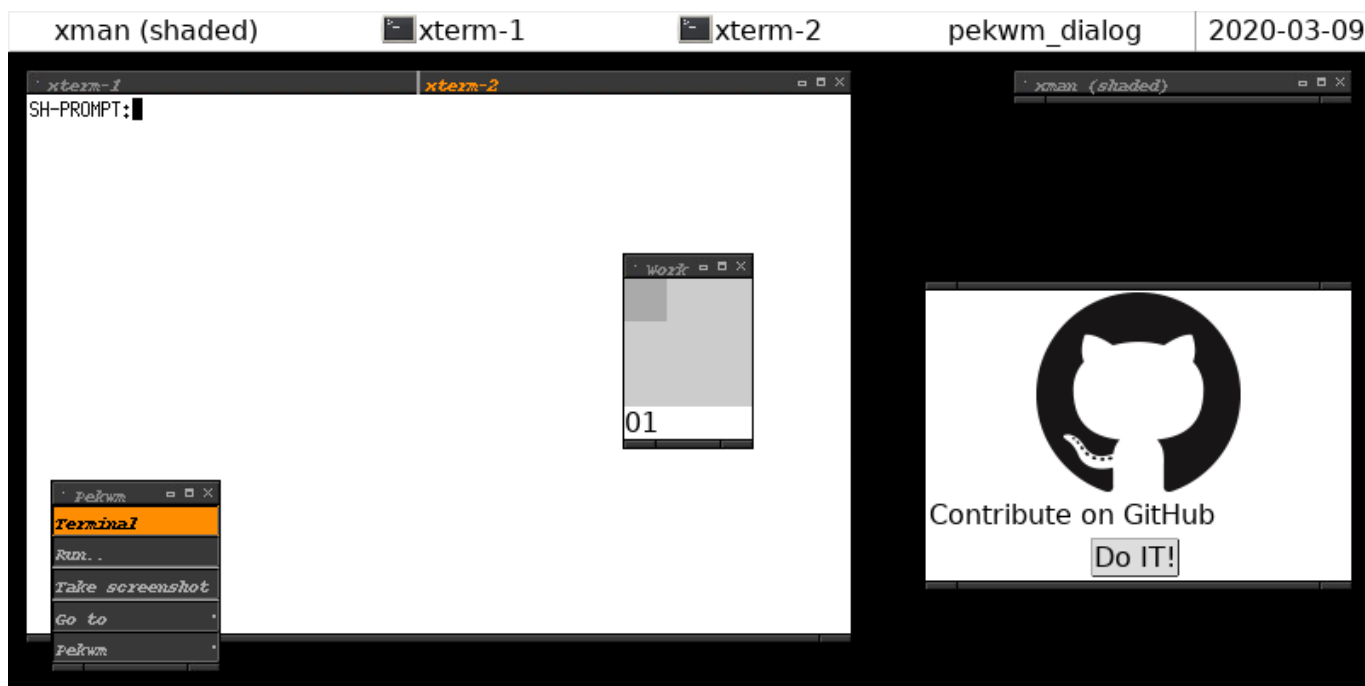
Ce gestionnaire de fenêtre est avant tout léger et configurable à souhait, comme vous pourrez le constater au fil de ce document. En effet, il ne dépend que de 24 librairies, devant 31 pour Openbox et 40 pour xfwm (le gestionnaire de xfce).



La version distribuée par Debian est la 0.1.18 sous Debian 11 alors que la version git est la 0.3.0 pour obtenir la dernière version, rendez-vous sur [la page git du projet](#).

À quoi ça ressemble ?

Voici quelques captures d'écran pour vous donner une idée :



Comment je peux le configurer ?

Tout se passe dans le dossier placé dans votre `/home/votre_nom_d_utilisateur/.pekwm`

Au moins après avoir lancé pekwm via votre gestionnaire de connexion, ou un fichier `.xinitrc`, vous devrez y trouver ces quelques fichiers dans `~/.pekwm` :

- start
- autoproperties
- keys
- mouse
- menu
- vars

et éventuellement le sous dossier “themes”, dans lequel vous placerez tous vos thèmes personnels.

Le fichier start

Dans ce fichier, vous avez la possibilité d'écrire les applications que vous souhaitez lancer au démarrage de pekwm.

N'oubliez pas tout d'abord de le rendre exécutable :

```
chmod +x ~/.pekwm/start
```

Voici un exemple de ce à quoi il peut ressembler :

[fichier.txt](#)

```
#!/bin/sh
mpd & # un peu de musique ne fait pas de mal
export 000_FORCE_DESKTOP=gnome # Pour avoir openoffice tout joli
tint2 & # Pour avoir une barre de taches
feh --bg-scale ~/Wallpapers/background.jpg & # Pour avoir un joli
fond d'écran
```

Vous remarquerez que les programmes lancés sont suivis du caractère “&”. Ceci est essentiel pour que pekwm puisse démarrer après ces commandes.

Le fichier vars

Dans ce fichier, vous pouvez spécifier différentes variables. Étudions cet exemple :

Dans ce fichier, il est écrit : `$TERM="urxvt"`

Ainsi, à chaque fois que l'on écrira `$TERM` dans les autres fichiers de configuration de pekwm, il sera interprété comme étant urxvt. Cela est pratique pour ne pas avoir à changer toutes les lignes une par une des fichiers de configuration.

Le fichier config

C'est le centre de la configuration de pekwm. Voici en vrac quelques précisions sur quelques lignes :

- Icons = “~/pekwm/icons/”

Ici, vous définissez le dossier contenant les icônes utilisées dans le menu de pekwm.

- Workspaces = “9”
- WorkspacesPerRow = “9”
- WorkspaceNames = “Main;Web;E-mail;Musique;DL;Jeux;Travail;Divers 1;Divers 2”

Ici, vous choisissez le nombre d'espaces de travail par colonne, puis combien vous en voulez par ligne, puis enfin leur nom.

- Harbour {

Dans cette section, vous configurez le "harbour", ou dock de pekwm. C'est un réceptacle aux applications dockées.

- Screen {

Ici, vous pouvez configurer le placement des fenêtres. Pour ajouter des marges par exemple, c'est tout simple. Cela se passe dans ces options :

```
EdgeSize = "10 1 1 1"
EdgeIndent = "True"
```

Les chiffres correspondent aux pixels de chaque cotés de l'écran qui sont réservés, le EdgeIndent permet de réserver de la place. Avec cette configuration, on a une marge de 10 pixels sur le haut de l'écran. C'est t'y pas beau ?

Le fichier menu

Ici, vous allez pouvoir configurer le menu. Un des grands avantages de pekwm est qu'il dispose de menus dynamiques, c'est-à-dire qui peuvent changer selon vos choix!

Par ailleurs, dans le menu, vous aurez la possibilité d'exécuter des programmes, mais aussi de faire tout un tas d'autres actions, comme redémarrer pekwm, changer de thème... etc. Regardez la section "Les Actions possibles" à la fin du texte pour plus de renseignements.

Un menu normal

Tout devra être compris entre les accolades suivantes : RootMenu = "Pekwm" { }

La syntaxe du fichier est la suivante :

- Entry = "Texte qui s'affiche dans le menu" { Actions = "Exec commande à lancer au clic &" }

Exemple : Entry = "vim" { Actions = "Exec \$TERM -title vim -e vim &" }

- Submenu = "Nom du sous-menu" {

Différentes entrées

...

...

}

Ceci vous permet d'ajouter un sous-menu, avec ce que vous voulez dedans.

- Separator { }

C'est pour ajouter une ligne séparatrice :)

Des entrées dynamiques

Pour cela, il faut utiliser la syntaxe suivante :

- Entry = "" { Actions = "Dynamic /chemin/vers/le/script" }

Bien sûr, il peut s'agir d'un script ou d'un programme python... etc, du moment qu'il retourne ce qu'il faut. On peut préférer avoir un sous-menu, ceci dit. Voici un script qui donnera la date et l'heure dans le menu :

script

```
#!/bin/sh
#pek-date.sh
#Ajoutez ceci dans le fichier ~/.pekwm/menu :
#Entry = "" { Actions = "Dynamic /path/to/the/script/pek-date.sh" }
date=$(date +%A\ %d\ %b\ \%H\:%M)
echo "Dynamic {"
echo "  Entry = \"$date\" { Actions = \"Exec source /dev/null & \" }"
echo "}"
exit 0;
```

Le "Exec source /dev/null" permet de ne rien lancer au clic, mais n'importe quelle application peut être spécifiée ici.

Voilà, vous pouvez créer vos scripts à partir de maintenant afin de réaliser vos rêves les plus fous.



Autogénérer le menu avec menumaker

Menu maker est un programme permettant de générer un menu. Vous pouvez le trouver [ici](#).

Le fichier keys

Ce fichier vous permet d'assigner tout un tas de raccourcis claviers, et de modifier ceux existant déjà.

À noter :

- Mod4 correspond à la touche  Win de votre clavier
- Mod1 à la touche  Alt

Dans ce fichier, comme dans le fichier menu, les actions peuvent être paramétrées. Voir la section "Les Actions possibles" à la fin du texte.

Le fichier mouse

Vous l'aurez deviné, ce fichier permet de configurer les actions possibles exécutées via la souris.

Voici un exemple, extrait d'un fichier mouse :

Extrait de fichier mouse

```
ScreenEdge {  
  Down {  
    ButtonRelease = "3" { Actions = "ShowMenu Root" }  
    ButtonRelease = "1" { Actions = "ShowMenu GotoClient" }  
  }
```

Ici, vous voyez que l'on configure la partie "screenedge", c'est-à-dire les bords de l'écran, en particulier le bord inférieur (Down). Ici, on voit que lorsque l'on fait un clic droit sur ce bord (buttonrelease 3), on affiche le menu root. Ceci est très pratique lorsque l'on ne veut pas utiliser de barres de tâches et que l'on veut avoir accès au menu lorsque les fenêtres sont ouvertes.

La deuxième ligne écrite ci-dessus permet d'afficher un menu listant toutes les fenêtres ouvertes via un clic-gauche sur le bord de l'écran : avez-vous vraiment besoin d'une barre de tâches ? :)

Il existe ainsi plusieurs menus.

Vous pouvez adapter les raccourcis souris à vos besoins. La section du "bureau", et la section Root.

Le fichier autoproperties

Ce fichier vous permet de définir le comportement de certaines fenêtres. Regardons un exemple :

Exemple de fichier autoproperties

```
Property = "Download,Iceweasel" {  
  ApplyOn = "New Start"  
  Workspace = "5"  
}
```

Ici, on voit que la fenêtre nommée Download, de classe Iceweasel, ira sur l'espace de travail 5 lors de son lancement. Le fichier est bien commenté pour comprendre toutes les possibilités.

Vous aurez sans doute besoin de lancer xprop en console, pour connaître les propriétés des fenêtres.

Les actions possibles

Au lieu de tout recopier, voici le lien vers la documentation vous expliquant tout en détail : [Les actions](#).

Liens externes

- [Site officiel](#)
- [Documentation officielle](#)
- [Thèmes officiels](#)
- [Site avec des thèmes intéressant](#)

¹⁾

N'hésitez pas à y faire part de vos remarques, succès, améliorations ou échecs !

From:

<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:

<http://debian-facile.org/doc/environnements:x11:pekwm>

Last update: **30/04/2023 23:37**

