

0.1 Créer des *.eapp et éditer les menus d'E17 manuellement

E17, contrairement à E16, est basé sur des fichiers-icons-applications en *.eapp et des fichiers cache.order pour l'édition des menus du curseur. Pour fixer les idées je vais indiquer comment j'ai ajouté l'application `gedit` à mon menu. Mais cela devrait fonctionner de manière analogue pour toute autre application.

0.1.1 1. Pré-requis

Ces logiciels sont à installer dans cet ordre :

- eet
- evas
- ecore
- embryo
- imlib2
- edje
- enlightenment

Vous les trouverez sur le site d'Enlightenment¹. Conseil avant compilation :

- ajoutez `/usr/local/lib` au fichier `/etc/ld.so.conf` (s'il n'y est pas) puis faire un `ldconfig`.
- ajoutez `/usr/local/bin` au `PATH` global (s'il n'y est pas).
- consultez toujours le `./configure -help` ou le fichier `INSTALL` avant de compiler un logiciel.
- consultez ces didacticiels sur l'installation de E16².

0.1.2 2. Principe et conventions

0.1.3 2.1 Principes

E17 fournit une archive `icon_example.tar.gz` qui est un canevas à adapter et à partir duquel nous allons générer un fichier `.eapp`. Ce fichier `.eapp` sera placé dans votre `~/ .e/`. Puis nous modifierons un fichier `cache.order` pour lui indiquer qu'on veut ajouter une application au menu. Il existe bien-sûr des éditeurs de menu tel `e_util_eapp_edit` mais nous ne les utilisons pas exprès ! Sachez tout de même qu'avec ces utilitaires, on peut créer ces *.eapp en cliquant sur l'icône en haut, tout à gauche de toute fenêtre ouverte, d'une application lancée depuis le terminal.

0.1.4 2.2 Conventions

- Je suppose dans la suite que l'installation s'est faite correctement, que le dossier `cache.e` a été créé et que vos `PATH` et `/etc/ld.so.conf` sont à jour.
- E17 est livré avec un dossier "data" donc le chemin chez moi est `/usr/local/share/enlightenment/d`. Si le chemin vers ce dossier est différent chez vous, pensez à adapter dans la suite.

¹<http://www.enlightenment.org/>

²[/categorie.php?id=29](#)

0.1.5 3. Exemple avec gedit

0.1.6 3.1 Génération du fichier .eapp

Commencez par décompresser le canevas :

```
$ cd /usr/local/share/enlightenment/data/other/
$ su
# tar -xzvf icon_example.tar.gz
```

Cela créé un dossier "/usr/local/share/enlightenment/data/other/icon". En étant root, éditez le fichier /usr/local/share/enlightenment/data/other/icon/icon.edc. Dans ce fichier, pour avoir une icône 16x16 remplacez :

```
max 48 48 ;
```

par :

```
max : 16 16 ;
```

Toujours dans le dossier /usr/local/share/enlightenment/data/other/icon/, remplacez icon.png par une icône png de gedit (chez moi c'est /usr/share/pixmaps/gedit-icon.png) qui devra obligatoirement s'appeler icon.png. Cela peut se faire par :

```
# cd /usr/local/share/enlightenment/data/other/icon/
# cp /usr/share/pixmaps/gedit-icon.png .
# mv gedit-icon.png icon.png
```

En étant root, éditez le fichier /usr/local/share/enlightenment/data/other/icon/build pour l'application gedit, écrasez son contenu par :

```
-set-name "gedit" \
-set-generic "gedit" \
-set-comment "myEditor" \
-set-exe "gedit" \
-set-win-name "window_name" \
-set-win-class "Gedit"
```

Il y a d'autres commandes. Vous pouvez toutes les visualiser en tapant : enlightenment_eapp -help (voir le 3.2). Les paramètres à donner à ces commandes sont assez simples à adapter. Sauf peut-être pour -set-win-class. Comment obtenir la chaîne "Gedit" ? Lancez l'application gedit depuis le terminal :

```
# gedit
```

Lancez un autre terminal. Dans ce dernier tapez :

```
# xprop WM_CLASS|cut -d \, -f 2
```

Le terminal reste muet et semble attendre quelque chose, c'est normal. Cliquez sur la fenêtre de gedit (que vous venez de lancer) et lisez ce qui s'affiche dans le terminal précédemment muet. Il devrait afficher "Gedit", c'est pourquoi j'ai mis dans le fichier build.sh : -set-win-class "Gedit". Fermez gedit et le terminal précédemment muet. Exécutez le script build.sh par l'une ou l'autre de ces commandes :

```
# sh build.sh
# bash build.sh
```

Finissez par :

```
# exit
$ cd /usr/local/share/enlightenment/data/other/icon/
$ cp icon.eapp ~/.e/e/applications/all/gedit.eapp
```

0.1.7 3.2 Mise à jour du menu

A ce stade vous avez votre gedit.eapp dans ~/.e/e/applications/all/. Il ne vous reste plus qu'à mettre à jour votre menu. En tant que simple utilisateur, tapez ceci :

```
$ mkdir ~/.e/e/applications/favorite/perso
```

Dans le dossier /home/votre_login/.e/e/applications/favorite/perso, créez un fichier cache.order et ajoutez-y gedit.eapp et tout ce que vous voulez de ~/.e/e/applications/all/. Ca y est ! C'est fini ! Vérifiez (cliquer sur le bureau) que le menu est à jour dans "/Favorite applications", tout en bas et que gedit se lance en cliquant dessus. Sinon faites un "Restart enlightenment".

0.1.8 3.3 Pour aller plus loin

Ce qui suit concerne ceux qui veulent en faire un peu plus.

modifier la barre de tâche

il suffit d'éditer le fichier .order de ~/.e/e/applications/bar/.

arrêter l'ordinateur depuis le menu

Il vous suffit de créer un halt.eapp comme nous l'avons vu pour gedit. Simplement, comme build.sh, utilisez celui-ci :

```
-set-name "halt" \  
-set-generic "halt" \  
-set-comment "arreter le pc" \  
-set-exe "enlightenment_remote -shutdown && sleep 3 &&  
  halt" \  
-set-win-name "window_name" \  
-set-win-class "halt"
```

Vous pouvez adapter le paramètre du -set-exe à votre guise. Le `sleep 3` permet de faire une pause de 3 secondes entre la fermeture d'E17 et l'extinction du PC.

l'application enlightenment_eapp

Pour changer une option dans un .eapp créé (vous pouvez tout changer sauf l'image) il n'est pas nécessaire de tout refaire depuis le début : utilisez `enlightenment_eapp` ! Toutes les options possibles s'obtiennent avec la commande :

```
enlightenment_eapp -help
```

Vous obtenez quelque chose comme :

-lang LANG : Set the language properties to modify
-set-name NAME : Set the application name
-set-generic GENERIC : Set the application generic name
-set-comment COMMENT : Set the application comment
-set-exe EXE : Set the application execute line
-set-win-name WIN_NAME : Set the application window name
-set-win-class WIN_CLASS : Set the application window class
-set-startup-notify [1/0] : Set the application startup notify flag to on/off
-set-wait-exit [1/0] : Set the application wait exit flag to on/off
-del-name : Delete the application name
-del-generic : Delete the application generic name
-del-comment : Delete the application comment
-del-exe : Delete the application execute line
-del-win-name : Delete the application window name
-del-win-class : Delete the application window class
-del-startup-notify : Delete the application startup notify flag
-del-wait-exit : Delete the application wait exit flag

exemple : soit icon.eapp une application-icon créé

– pour ajouter une option :

```
enlightenment_eapp -set-win-class "l'option" icon.eapp
```

– pour enlever une option :

```
enlightenment_eapp -del-win-class "l'option" icon.eapp
```

`enlightenment -help` donne des informations sur le xinerama.