

Injector.doc

COLLABORATORS

	<i>TITLE :</i> Injector.doc		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		August 19, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Injector.doc	1
1.1	Documentation d'Injector - Français	1
1.2	1.1 Distribution et décharge	2
1.3	1.2 Crédits et remerciements	3
1.4	1.3 Configuration nécessaire	3
1.5	2.1 Introduction	3
1.6	2.2 Lancer Injector	4
1.7	2.3 Options & ToolTypes	5
1.8	L'option RequesterKey	5
1.9	L'option PasteFileKey	5
1.10	L'option PasteDirKey	6
1.11	L'option PastePathKey	6
1.12	L'option CXPri	6
1.13	L'option IconPath	6
1.14	L'option PubScreen	7
1.15	2.4 Terminer Injector	7
1.16	3.1 Invoquer le requester	7
1.17	3.2 Coller le contenu du requester	7
1.18	3.3 Créer des icônes pour vos fichiers	8
1.19	4.1 Définir des touches de commande	8
1.20	4.2 Historique	10
1.21	Contacteur l'auteur	11
1.22	Future	12

Chapter 1

Injector.doc

1.1 Documentation d'Injector - Français

Bienvenue à Injector 1.13 copyright 1993 Frédéric DELACROIX.

Ceci est le document qui doit toujours être distribuer avec les autres fichiers. Il est destiné à être affiché par AmigaGuide (copyright Commodore), mais peut aussi être lu par des yeux humains, le confort en moins.

TABLE DES MATIERES

1 AVANT-PROPOS :

1.1 Distribution
<- Important !

1.2 Crédits et remerciements

1.3 Configuration nécessaire

2 INSTALLATION D'INJECTOR :

2.1 Introduction

2.2 Lancer Injector

2.3 Options & Tooltypes

2.4 Terminer Injector

3 UTILISATION D'INJECTOR :

3.1 Invoquer le requester

3.2 Coller le contenu du requester

3.3 Créer des icônes pour vos fichiers

4 APPENDICE

4.1 Définir des touches de commande

4.2 Historique

4.3 Contacter l'auteur
<- Faites-le !

4.4 Futur

1.2 1.1 Distribution et décharge

Injector est Copyright 1993 par Frédéric DELACROIX. La permission de le copier et de le distribuer est accordée à quiconque respecte ces conditions (généralement connues sous le nom de SHAREWARE) :

- Tous les fichiers (ou toute l'archive) soient distribués ensemble (cela concerne l'exécutable, les fichiers de documentation, les fichiers catalogues, le fichier de transcription de catalogue, le tiroir kd_freq et tous les icônes). Il y a toutefois une exception: Si vous prévoyez une diffusion pour une communauté "mono-linguistique" (disons, par exemple, seulement la France), vous êtes autorisé à ne distribuer que les fichiers relatifs à votre langue, c'est-à-dire: le fichier de documentation et le catalogue. Tous les autres fichiers doivent être présents.

- Tous les fichiers distribués ne soient MODIFIES EN AUCUNE FACON (pas de message idiot du genre "distribué par..."). Si vous avez des commentaires à ajouter, faites-le dans un fichier séparé! L'archivage est toutefois autorisé, mais le compactage de l'exécutable n'est pas recommandé car le programme est auto-détachant (il a besoin de couper sa SegList).

- Vous ne pouvez pas demander d'argent pour ce programme. Une petite somme est autorisée pour la copie et l'envoi, mais vous ne pouvez PAS demander plus que Fred Fish pour un AmigaLib disk unique.

- Ceci est pour tous les utilisateurs d'Injector: comme le programme est distribué en SHAREWARE, vous devez m'envoyer une petite contribution de \$10 (ou équivalent) si vous continuez à l'utiliser après une courte période d'évaluation. Si vous voulez le source du programme (écrit pour Devpac 3), ajoutez simplement \$10 et je vous l'enverrai. Mon adresse se trouve à la fin de ce document.

Injector a été testé minutieusement, mais je ne peux pas garantir qu'il marchera toujours comme il le devrait. Je ne pourrai être tenu pour responsable pour aucun dommage/perte de données qui pourrait résulter de l'utilisation de ce programme. Noubliez pas: vous l'utilisez A VOS RIQUES ET PERILS.

Note particulière: le catalogue et les fichiers de documentation sont disponibles uniquement en anglais et en français (mon espagnol d'école est si mauvais :-). Alors si vous pouvez traduire ces fichiers dans votre langue, j'apprécierais beaucoup que vous me les envoyiez pour que je les inclue dans une version future. Pour le fichier de documentation, c'est facile: éditez simplement une copie de celui-ci (ne changez pas le nom des nodes SVP, ils ne sont pas affichés de toute façon). Pour le catalogue, remplissez simplement le fichier .ct avec vos propres chaînes et envoyez-le moi. Je vous retournerai le catalogue correspondant. Notez que les gadgets Choisir/Annuler sont limités à 7 caractères, et que ce qui sont affichés dans l'extrabutton du filerequester (Icônes/Montrer) sont limités à 10 caractères. La chaîne de description pour commodities est aussi limitée en longueur (voyez le fichier include pour la longueur maximale).

1.3 1.2 Crédits et remerciements

Injector a été écrit avec le merveilleux Devpac 3 d'HiSoft, tournant sur un vieil A500 avec l'OS 2.1. Il utilise bien sûr la superbe kd_freq.library qui est Copyright Khalid Aldoseri. Le concept d'Injector a été trouvé par Khalid Aldoseri (il faisait partie du programme FR_Bypass).

J'aimerais remercier les personnes suivantes:

Khalid Aldoseri, Serge Hammouche, Philsart Dulippe, Yves Quiquempois, François Jacobs, Antoine Camerlync, Christophe Descamps.

1.4 1.3 Configuration nécessaire

Injector devrait marcher sur n'importe quel amiga qui ↔
remplit ces

conditions:

- Il vous faut le kickstart 2.04 or plus récent (V37+), ou Injector refusera de tourner. Si vous utiliser encore le vieil OS1.3, la seule chose à faire est: VOUS METTRE A JOUR!

- Il vous faut la kd_freq.library installée dans votre tiroir LIBS:. La version 3.10 est fournie dans cette distribution (trouvez-la dans le directory kd_freq).

- Pas obligatoire, mais utile: Injector utilise la locale.library pour fonctionner dans différentes langues. Pour l'instant, seuls l'anglais et le français sont supportés, voyez

distribution
pour la traduction dans

votre langue.

1.5 2.1 Introduction

Beaucoup d'entre vous connaissent probablement la fonction d'Injector. En effet, à la base il n'est pas différent de celui qui était inclus avec la distribution originale de la kd_freq.library (par Khalid Aldoseri). Ceci est une tentative pour rendre l'Injector plus pratique et plus respectueux du système.

C'est pourquoi Injector nécessite l'OS2.04: c'est maintenant une commodity (qui peut bien entendu être contrôlée par le programme Exchange).

De plus, Injector fait de son mieux pour ne pas allouer de la mémoire inutile. Par exemple, si vous lancez Injector mais ne l'utilisez jamais, aucun tampon n'est alloué, et la kd_freq.library (qui consomme elle-même beaucoup de mémoire) n'est même pas ouverte.

Injector est une commodity invoquée par des touches de commande: quand vous pressez une combinaison de touches pré-définies (par défaut amiga-gauche shift-gauche r), un filerequester apparaît et attend que vous sélectionniez un fichier ou un directory. Si vous confirmez la sélection, vous pourrez "injecter" the nom de fichier, tiroir ou chemin d'accès dans la chaîne d'InputEvent (comme si vous les aviez tapés au clavier) en pressant d'autres touches de commandes pré-définies (par défaut, f,d et s respectivement, toutes utilisées avec les touches amiga-gauche et shift-gauche). Ceci se révèle très utile pour les programmes qui n'ont pas de requester intégré (comme le Shell).

Une possibilité hautement intéressante d'Injector est qu'il est capable d'ajouter des icônes à des fichiers en deux clics de souris. Pour que cela marche sans fenêtres supplémentaires qui prendraient beaucoup de RAM, j'ai utilisé l'option ExtraButton de la kd_freq.library: vous sélectionnez simplement le fichier pour lequel vous voulez un icône et cliquez sur le gadget Icônes. Un autre filerequester apparaîtra vous demandant de sélectionner un fichier icône (.info). C'est aussi très utile.

Bon, voilà pour cette introduction rapide, maintenant jetez un œil aux sections de référence.

1.6 2.2 Lancer Injector

Il y a bien sûr deux moyens de lancer Injector: par le CLI et ←
par le workbench.

A partir du CLI, Injector se détachera automatiquement du CLI, permettant à la fenêtre de se fermer, ou à la startup-sequence de continuer. Il n'y a pas besoin d'utiliser la commande Run. Le format des options est le suivant:

```
REQUESTERKEY
/K,
PASTEFILEKEY
/K,
PASTEDIRKEY
/K,
PASTEPATHKEY
/K,
CXPRI
/N/K,
```

```
ICONPATH
/K,
PUBSCREEN
/K
```

/K signifie que le nom de l'option doit être donné (ex: ICONPATH DH0:Icons ou ICONPATH=DH0:Icons) pour que l'option soit prise en compte. /N signifie que l'argument est supposé être un nombre.

Je suggère que vous voyiez la section Options & Tooltypes de ce document pour une complète explication de ces mots-clés.

A partir du workbench, le meilleur endroit est bien sûr le tiroir WBStartup de votre disque de démarrage, pour qu'Injector soit lancé à chaque boot. Vous pouvez aussi double-cliquer sur son icône et Injector commencera. Les tooltypes reconnus sont:

```
REQUESTERKEY
,
PASTEFILEKEY
,
PASTEDIRKEY
,
```

PASTEPATHKEY

,

CXPRI

,

ICONPATH

,

PUBSCREEN

.

Ils s'utilisent de la même façon que les options du CLI. Voyez la section

Options & Tooltypes

.

1.7 2.3 Options & ToolTypes

Cette section décrit les différentes options, utilisés comme ↔
mots-clés à
partir du CLI ou ToolTypes à partir du Workbench. Ce sont :

REQUESTERKEY

CXPRI

PASTEFILEKEY

ICONPATH

PASTEDIRKEY

PUBSCREEN

PASTEPATHKEY

1.8 L'option RequesterKey

Cette option est utilisée pour définir la touche de commande ↔
qui sera
utilisée pour invoquer le file requester. Ce doit être une chaîne de
description valide pour la commodities.library (voyez

Définir des touches de commande
) . Par défaut c'est :
lcommand lshift r

1.9 L'option PasteFileKey

Cette option est utilisée pour définir la touche de commande ↔
qui sera
utilisée pour coller le nom de fichier. Ce doit être une chaîne de

description valide pour la commodities.library (voyez

```
Définir des touches de commande
). Par défaut c'est:
    lcommand lshift f
```

1.10 L'option PasteDirKey

Cette option est utilisée pour définir la touche de commande ← qui sera utilisée pour coller le nom de tiroir. Ce doit être une chaîne de description valide pour la commodities.library (voyez

```
Définir des touches de commande
). Par défaut c'est:
    lcommand lshift d
```

1.11 L'option PastePathKey

Cette option est utilisée pour définir la touche de commande ← qui sera utilisée pour coller le nom de chemin. Ce doit être une chaîne de description valide pour la commodities.library (voyez

```
Définir des touches de commande
). Par défaut c'est:
    lcommand lshift s
```

1.12 L'option CXPri

Ceci sert à spécifier un nombre qui sera utilisé comme priorité du Broker d'Injector dans la liste de commodities. C'est utile si vous voulez que les touches de commande d'Injector "écrasent" les touches d'une autre commodity (ou le contraire). Plus la priorité est haute, plus Injector reçoit les InputEvents tôt (relativement aux autres brokers). Par défaut, c'est 10.

1.13 L'option IconPath

Cette option dit à Injector où trouver les fichiers icônes. Si vous avez, sur votre disque dur, un tiroir où vous rangez tous les icônes dont vous vous servez souvent, parce que vous utilisez ToolManager ou que vous créez souvent des fichiers pour lesquels vous voulez un icône personnel, vous pouvez associer le nom de ce tiroir à l'option ICONPATH, pour que quand Injector vous demande de sélectionner un icône, le filerequester apparaisse dans ce directory. ex: ICONPATH SYS:Icons . Par défaut Injector utilise le directory à partir duquel il a été lancé. Vous pouvez bien entendu changer de directory dans le requester lui-même.

1.14 L'option PubScreen

Cette option dit à Injector d'ouvrir les requester sur l'écran public dont le nom est donné. Si l'écran public ne peut pas être verrouillé (parce qu'il n'existe pas, ou a été fermé, ou est en statut privé), l'écran public par défaut est utilisé. Pour utiliser l'écran du workbench, you donnez: PUBSCREEN=Workbench, pour utiliser celui de Devpac 3, utilisez PUBSCREEN=DEVPAC.1 .

Par défaut, Injector utilise l'écran le plus en avant si il est public, sinon l'écran public par défaut.

1.15 2.4 Terminer Injector

Il y a deux façons de terminer Injector. La première est d'utiliser le programme Exchange (qui est sur la disquette Workbench) et choisir "Supprimer". La seconde est de re-lancer Injector. Dans les deux cas, le processus résident d'Injector fermera tout et quittera. Cela sera rapporté dans un requester.

1.16 3.1 Invoquer le requester

Pour invoquer le requester, pressez la combinaison de touches ←
pré-définie

(choisie par

REQUESTERKEY

), et un filerequester apparaîtra. Si c'est la

première fois qu'il s'ouvre, il viendra dans le tiroir actuel d'Injector et commencera à lire les noms de fichiers dans ce tiroir. Vous pourrez alors naviguer dans l'arborescence des tiroirs comme dans n'importe quel file requester (voyez le fichier fr_User). Quand vous aurez trouvé le fichier que vous voulez, confirmez le requester. Vous pourrez alors coller ce nom comme expliqué dans les autres sections de ce fichier. Si le requester a déjà été invoqué auparavant, il n'aura pas besoin de re-lire le tiroir.

1.17 3.2 Coller le contenu du requester

Maintenant que vous avez sélectionnez un fichier ou un ←
tiroir dans le

requester, vous pouvez "injecter" le nom de fichier, de tiroir, ou de chemin complet comme si vous les aviez tapés au clavier. Il vous suffit de presser la touche de commande correspondante (choisie par

PASTEFILEKEY

,

PASTEDIRKEY

,

PASTEPATHKEY

). Si le nom contient des espaces, il

sera entouré de guillemets.

1.18 3.3 Créer des icônes pour vos fichiers

Vous vous rappelez de la dernière fois que vous avez appelé `requester` ? N'avez-vous pas remarqué un petit bouton sur la droite nommé "Icônes" ? Si ? (Vous êtes très observateur :-)

Injector a le pouvoir de copier un icône d'un fichier vers un autre. C'est très utile si vous avez un tiroir sur votre disque dur où vous les stockez tous. Je vais maintenant décrire la procédure à suivre pour ce faire.

Premièrement, vous devez appeler le requester normalement (`lcommand lshift r`), et choisir normalement votre fichier comme décrit précédemment. Mais au lieu de cliquer sur "Okay" pour confirmer votre sélection, cliquez sur "icônes". Après une pause (parce que la routine `MapANSI()` de la `keymap.library` est plutôt lente), un autre `filerrequester` apparaîtra. Notez qu'il n'y a AUCUN lien entre lui et celui que vous avez utilisé avant. Si il est défini,

`ICONPATH`

sera utilisé comme tiroir par défaut. Vous pouvez choisir un fichier icône (seuls les fichiers `.info` sont affichés). Vous pouvez remarquer encore une fois qu'il y a un `ExtraButton`. Il est nommé "Montrer" cette fois. Devinez ce qui se passe si vous cliquez dessus ?!? Non ? Eh bien, Injector vous montrera l'icône que vous venez de sélectionner dans une petite fenêtre. L'icône est en fait un gadget que vous pouvez sélectionner pour voir ses deux états.

Il y a une grosse différence entre la façon dont Injector montre les icônes et celle du Workbench. Le workbench a un mode "floodfill" qui n'est pas supporté par Intuition (il remplace le flag `GFLG_GADGHBOX`), et quand Injector rencontre un tel icône, il affichera l'état sélectionné avec un cadre autour de l'image. Ce sera peut-être réparé sur demande populaire.

Quand vous avez fait votre choix, confirmez le requester en cliquant Okay, et l'icône sera copié à l'endroit voulu. Si le Workbench a une fenêtre ouverte sur le tiroir de destination, il sera averti et le nouvel icône apparaîtra immédiatement. J'oubliais: avant d'être copié, l'icône sera "libéré", c'est-à-dire qu'il n'aura aucune position particulière dans la fenêtre de destination.

Voilà! Je trouve cette possibilité très utile. J'espère que vous aussi.

1.19 4.1 Définir des touches de commande

Le texte qui suit ne dépend que du comportement de la `commodities.library` et pas sur celui d'Injector. C'est comme cela que `commodities` comprendra la combinaison de touches à laquelle vous voulez associer une action.

Une chaîne de description est constituée comme suit:

```
[<classe>] {[-][<qualificateurs>]} [-][upstroke] [<code>]
```

Tous les mots-clés sont insensibles aux majuscules.

`classe:` doit être une classe d'`InputEvent`. Les classes supportées sont `rawkey` pour les événements clavier, `rawmouse` pour les événements souris, `diskinserted` et `diskremoved`. Par défaut c'est `rawkey`, qui doit généralement être utilisée.

qualificateurs: c'est une série de mots-clés représentant l'état des qualificateurs du clavier (Shift,Alt etc...). Voici une liste de mots-clés connus. Ceux qui sont marqués par un * sont nouveaux pour la version 38 de commodities.library.

```

lshift,left_shift *:      Touche shift gauche.
rshift,right_shift *:    Touche shift droite.
shift:                   Une touche shift.
capslock,caps_lock *:    Touche Caps Lock.
caps:                    Touche Caps Lock ou shift.
control,ctrl *:          Touche Control.
lalt,left_alt *:         Touche Alt gauche.
ralt,right_alt *:        Touche Alt droite.
alt:                      Une touche Alt.
lcommand,lamiga *,left_amiga *,left_command *: Touche amiga gauche.
rcommand,ramiga *,right_amiga *,right_command *:Touche amiga droite.
numericpad,numpad *,num_pad *,numeric_pad *: Pour les touches du clavier
numérique.
leftbutton,lbutton *,left_button *: Bouton souris gauche.(1)
midbutton,mbutton *,middlebutton *,middle_button *:
Bouton souris milieu.(1)
rbutton:,rightbutton *,right_button *: Bouton souris droit.(1)
repeat:                   Répétition active.(2)

```

Notes: (1) la commodities.library V37 contenait une erreur qui l'empêchait d'utiliser leftbutton,midbutton et rbutton comme qualificateurs. Ce fut réparé pour la version 38.

(2) pour la classe rawkey uniquement.

(3) si une touche de commande doit être insensible à l'état d'un qualificateur, placez un - avant son nom.

upstroke: Normalement un évènement est généré seulement quand la touche est pressée. Vous pouvez changer ce comportement en ajoutant upstroke. Cela générera un évènement uniquement quand la touche sera relâchée. Si les faits de presser et de relâcher doivent tous deux générer un évènement, utilisez -upstroke.

code: Ceux-ci sont significatifs uniquement pour les classes rawkey et rawmouse. Pour rawkey,voici les codes de touches, * sont nouveaux pour la version 38 de commodities.

```

a to z, 0 to 9:          Caractères ASCII normaux.
f1 to f10,f11 and f12:  Touches de fonction.
up,cursor_up *:         Flèche haut.
down,cursor_down *:     Flèche bas.
left,cursor_left *:     Flèche gauche.
right,cursor_right *:   Flèche droite.
esc,escape *:           Touche Esc.
backspace               Espace arrière.
del                     Touche Del.
help                   Touche Help.
tab                     Touche Tab.
comma                  Touche virgule (,).
return                 Touche retour chariot.
space,spacebar *       Barre d'espace.
enter                  Touche Enter.(4)
insert *               Touche pad 0.(4)
delete *               Touche pad 1.(4)

```

page_up * Touche pad 9.(4)
page_down * Touche pad 3.(4)
home * Touche pad 7.(4)
end * Touche pad 1.(4)

Notes: (4) à utiliser avec le qualificateur numericpad.

Pour rawmouse, les codes valides sont: (uniquement pour Commodities V38):
mouse_leftpress: Bouton gauche pressé.(5)
mouse_middlepress: Bouton du milieu pressé.(5)
mouse_rightpress: Bouton droit pressé.(5)

Notes: (5) vous devez utiliser aussi le qualificateur correspondant.

1.20 4.2 Historique

Revision V1.13

Ca y est enfin ! Je suis arrivé à la toute première distribution publique, en SHAREWARE. Comme d'habitude, j'ai fait un peu de rangement et tout rassemblé dans un beau directory avec de jolis fichiers readme. (10-Oct-93 14:31:10)

Revision V1.12

Ajouté une chaîne de version 2.0 et écrit des doc AmigaGuide Pas de doc ASCII pour ce programme, il vous faut AmigaGuide (Fish 870) pour les lire. Optimisé le placement de la fenêtre icône (droite du requester). La distribution publique est pour la prochaine révision.

Revision V1.11

Le tooltype CXPRI foirait, réparé maintenant. J'ai écrit un catalogue français et j'ai réparé l'interface avec locale. Fait quelques réparations et optimisations diverses. Je prévois une distribution publique bientôt. Promis!

Revision V1.10

Eh bien, j'ai enlevé la traduction automatiquement (implémentée avec les objets translates de Commodities). Avantages: cela raccourci le programme, commodities ne cause plus de gurus étranges en accédant de façon asynchrone des données au mauvais moment. Inconvénient: Impossible d'utiliser les hotkeys quand le requester est actif.

Revision V1.09

J'ai fait marcher les tooltypes PubScreen et IconPath et j'ai enlevé quelques bugs. Je pense à écrire un fichier de doc maintenant...

Revision V1.08

J'ai fait un squelette pour le support des options CLI et tooltypes. Options supportées pour l'instant: toutes les hotkeys, priorité Commodities. Bientôt: IconPath and PubScreen.

Revision V1.07

Ecrit mes propres routines InvertString()/FreeIEvents() pour qu'Injector n'interprète plus mal les noms de fichier avec des <>. Aussi réparé l'auto-détachant du Workbench pour permettre l'utilisation du tiroir WBStartup.

Revision V1.06

Maintenant injector se détache du CLI, plus besoin d'utiliser Run. Si la fenêtre de l'icône est déplacée, la prochaine fois elle s'ouvre au même endroit.

Revision V1.05

L'icône affiché est finalement un gadget réellement sélectionnable, permettant de voir les deux états actif et inactif. Notez qu'il peut y avoir une différence avec les icônes du Workbench, qui a un mode "floodfill" non supporté par les gadgets Intuition standards. Peut-être plus tard.

Revision V1.04

Ah, une cool nouvelle option: le requester d'icône peut maintenant afficher les icônes eux-mêmes dans leur propre petite fenêtre, en utilisant encore un joli ExtraButton. Pour l'instant, l'icône n'est pas un vrai gadget (à peine une image), mais ce sera fait bientôt.

Revision V1.03

Ca y est, Injector est maintenant capable de copier des icônes pour des tiroirs, choisissez simplement un nom de fichier vide.

Revision V1.02

Ajouté une nouvelle option intéressante: Injector peut maintenant copier des icônes d'un endroit à un autre, en utilisant l'option ExtraButton de la kd_freq. C'est très utile. Il faut encore que je m'occupe des icônes de tiroirs.

Revision V1.00

--- Initial release ---

1.21 Contacter l'auteur

Je vous rappelle qu'Injector est SHAREWARE. Je sais que la plupart d'entre vous ne me paieront rien après leur période d'évaluation, mais sachez que j'ai passé beaucoup de temps à écrire et déboguer ce programme, et que je voudrais juste une petite compensation qui me permettra d'acheter

un A1200. Payer \$10 vous fournira un support pour les prochaines versions. Je garde le source au chaud pour ceux d'entre vous qui sont intéressés par la programmation du merveilleux système d'exploitation de l'amiga, pour \$10 de plus.

Pour tout ce qui concerne l'enregistrement, les commentaires, rapport de bugs, demandes d'améliorations, cartes postales, vous pouvez m'écrire à:

Frédéric DELACROIX
5 rue d'Artres
59269 QUERENAING
FRANCE.

1.22 Future

Injector est bien entendu sujet à évolution. La prochaine version pourra comporter des choses comme:

- une interface ARexx complète
- support de la multisélection
- prise en compte du flag GFLG_GADGHBOX des icônes
- peut-être des bugfixes
- tout ce qui pourra vous intéresser !