Localisation: Adapter Slackware à votre Langue

Cette page contient les instructions qui permettent d'adapter votre Slackware à votre langue usuelle.

Vue d'ensemble

À la différence d'autres distributions, Slackware est une distribution nettement basée sur la langue anglaise. Son installateur n'est pas localisé - il n'est disponible qu'en anglais. L'installation vous permet seulement de choisir une disposition de clavier pour la console et l'emploi de l'encodage UTF-8 en mode console (ce qui est obtenu par un paramètre du noyau défini dans `/etc/lilo.conf`). Cependant il n'est pas difficile de localiser votre système Slackware pour qu'il affiche les commandes dans votre langue et qu'il emploie un environnement graphique de bureau configuré avec votre langue, votre système de mesures et autres localisations.

Plusieurs réglages disponibles sont indépendants les uns des autres à des degrés divers. La disposition du clavier pour la console peut être choisie à l'installation, mais elle est indépendante du choix de clavier dans l'environnement X11. le support des caractères Unicode (UTF-8) étend la table ASCII et permet d'afficher les caractères qu'on ne trouve pas dans la langue anglaise.
La variable d'environnement $LANG peut être configurée pour tout le système ou “par utilisateur”. Fondamentalement, cette variable indique la langue utilisée par les applications quand elles interagissent avec l' utilisateur.
Il faut aussi bien distinguer si l'ordinateur démarre en niveau 3 (console) ou en niveau 4 (accès à l'environnement graphique X-Window).

Le fuseau horaire

S'il vous faut changer le fuseau horaire de votre système vous pouvez tout simplement employer en ligne de commande l'outil “timeconfig” (en étant root). Il est possible de choisir votre fuseau horaire à la main, en faisant un lien symbolique de la chaîne qui correspond à ce fuseau horaire depuis `/usr/share/zoneinfo/` vers le fichier `/etc/localtime`(sous l'identité root).
Exemple:

```
ln -s /usr/share/zoneinfo/Europe/Paris /etc/localtime
```

Locale

Lisons le début de `man 7 locale`:

```
Une locale est un ensemble de règles liées à la langue et à la culture.
Elles traitent d'aspects comme
la langue des messages, les différents ensembles de caractères, les
```

```
conventions d'écriture, toutes ces
précisions là. Un programme doit être en mesure de déterminer sa locale et
se comporter en conséquence
pour bien s'adapter à différentes cultures.
```

La commande `locale` vous indique les langues gérées par votre système Linux . Demandez:

```
locale -av | more
```

qui vous donne une liste très complète. En voici un extrait à titre d'exemple:

```
locale: fr_FR           directory: /usr/lib64/locale/fr_FR
-------------------------------------------------------------------------
---
    title | French locale for France
   source | RAP
  contact | Traduc.org
    email | bug-glibc-locales@gnu.org
 language | French
territory | France
 revision | 1.0
     date | 2008-03-15
  codeset | ISO-8859-1

locale: fr_FR.utf8      directory: /usr/lib64/locale/fr_FR.utf8
-------------------------------------------------------------------------
---
    title | French locale for France
   source | RAP
  contact | Traduc.org
    email | bug-glibc-locales@gnu.org
 language | French
territory | France
 revision | 1.0
     date | 2008-03-15
  codeset | UTF-8

locale: fr_FR@euro      directory: /usr/lib64/locale/fr_FR@euro
-------------------------------------------------------------------------
---
    title | French locale for France with Euro
   source | Free Software Foundation, Inc.
  address | http://www.gnu.org/software/libc/
    email | bug-glibc-locales@gnu.org
 language | French
territory | France
 revision | 1.0
     date | 2000-08-20
  codeset | ISO-8859-15
```

Dans le système, une locale est constituée de variables d'environnement, certaines d'entre elles étant

plus utilisées que d'autres. Les plus intéressantes sont :

- LANG
- LANGUAGE
- LINGUAS
- LC_COLLATE
- LC_ALL

Si vous lisez cette page de manuel `locale(7)`, vous verrez qu'elle cite plusieurs des variables constitutives d'une locale et qui permettent de modifier le comportement de l'ordinateur vis à vis de la langue. Les noms de toutes ces variables ont pour préfixe “LC_”, comme LC_COLLATE, LC_CTYPE, LC_MONETARY, LC_MESSAGES, LC_NUMERIC et LC_TIME, ou la variable plus globale “LC_ALL” qui prendra le pas sur toutes les précédentes si vous la definissez. On peut aussi utiliser la variable LANG pour remplacer toutes ces variables de type “LC_”.

Cela signifie que si vous le voulez, votre ordinateur peut être configuré pour se conformer à une locale donnée de manière plus ou moins complète. Par exemple vous pouvez décider d'obtenir les réponses des commandes dans votre langue (en utilisant la variable **LANG**) tout en conservant l'ordre de tri par défaut suivant les traditionnelles valeurs ASCII (en faisant appel à la variable **LC_COLLATE**), ce qui rend les réponses de votre système plus fiables.

'C' est l'ancien réglage par défaut de Slackware (et d'UNIX), qui se base sur une table de caractères ASCII 127-bit conforme à la normalisation ANSI_X3.4-1968. Maintenant, il est préférable d'employer en_US ou un autre choix moderne pour $LANG, supportant les ensembles étendus de caractères.
Le support des locales UTF-8 est également disponible, mais sachez que quelques programmes ne peuvent pas encore fonctionner avec les caractères UTF-8 et ne se comporteront pas de façon satisfaisante. Dans ces cas, vous pourrez fixer la valeur LANG=C avant d'utiliser ces programmes là.

Précisions sur les variables d'environnement

Les variables d'environnement qui interviennent dans les configurations des locales utilisent des valeurs comme **`nl_NL`**, au début de la chaîne, en minuscules on trouve le ***codelangue*** tandis qu'à la fin, en majuscules, se trouve le ***codepays***. Le *codelangue* concerne la langue en général cependant que le *codepays* se rapporte aux spécificités de chaque pays où cette langue est parlée.
Exemple: fr_CA pour le français au Canada et fr_FR pour le français de France.

Configuration étendue au système

Slackware dispose d'un script de profil global qui vous permet d'établir la locale de votre système pour tous les utilisateurs d'un seul coup (à portée générale sur le système). Ce qui se fait simplement en éditant ce script au moyen d'un éditeur ASCII tel que `nano` ou `vi` ou `mcedit` ou autre. En étant root:

```
/etc/profile.d/lang.sh
```

(les utilisateurs de C-shell et ses dérivés doivent éditer `/etc/profile.d/lang.csh`).

Si vous voulez intervenir rapidement, repérez dans ce fichier la ligne qui indique:

```
# en_US is the Slackware default locale:
export LANG=en_US
```

et remplacez la valeur pour votre propre langue. Une famille francophone changera cela en:

```
# en_US is the Slackware default locale:
export LANG=fr_FR
```

Et si je veux aussi que mon ordinateur gère le texte Unicode, je changerai la ligne en:

```
# en_US is the Slackware default locale:
export LANG=fr_FR.UTF-8
```

Remarquons que Slackware n'est pas entièrement préparée pour Unicode. Quelques applications (comme les pages de manuel) n'afficheront pas correctement du texte Unicode. Pour certaines langues (en particulier les non-Latines) il faut obligatoirement activer la prise en charge d'Unicode (voir un peu plus loin) car sans cela les glyphes des caractères (ce sont les dessins ou tracés des lettres) ne s'afficheraient pas (vous ne verriez que des petits rectangles, à leur place).

Configuration par utilisateur

Si vous ne voulez pas prédéfinir une locale pour tout le monde, vous avez cependant à votre disposition les mêmes variables d'environnement pour un utilisateur individuel.

Dans ce cas vous n'utiliserez pas `/etc/profile.d/lang.sh` mais vous définirez plutôt ces variables dans le fichier d'initialisation du shell de l'utilisateur. *Bash lit différents fichiers d'initialisation suivant qu'il est appelé en tant que login shell ou non.* Quand l'utilisateur se connecte dans la console, Bash est appelé en tant que *login shell* et se règle en fonction du contenu du fichier `~/.profile` dans le répertoire /home/<nom_utilisateur> de cet tilisateur. Par conséquent on ajoute la ligne

```
LANG=de_DE.utf8
```

au fichier `~/.profile` pour rendre effective la langue allemande, par exemple. Ceux qui emploient le C-shell éditeront plutôt `~/.cshrc`.

La console

Notez bien que seul root peut utiliser `liloconfig` et `pkgtool` !

Réglage de LILO pour Unicode

L'étape de configuration de LILO dans l'installeur de Slackware vous permet d'activer UTF-8 par défaut:

Si vous choisissez l'option par défaut (“No”), vous pourrez activer Unicode pour la console plus tard en ajoutant l'option “`vt.default_utf8`” dans votre fichier `lilo.conf`:

```
# Start LILO global section
# Append any additional kernel parameters:
append="vt.default_utf8=1"
boot = /dev/sdx
```

Ou en utilisant `liloconfig`:

```
liloconfig
```

Redémarrez votre système pour appliquer les changements.

Note: vous pouvez aussi appeler `liloconfig` depuis la section “setup” de `pkgtool`.

Choix de la police pour la console

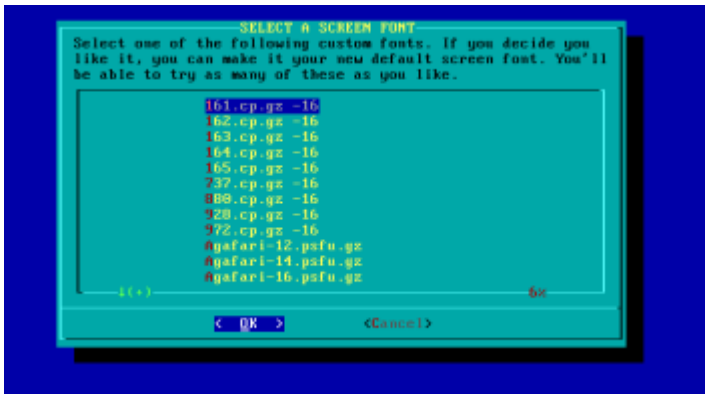
La section de configuration du système, dans l'installeur Slackware, vous permet de choisir une police qui correspond à vos goûts personnels:

Si vous choisissez “Oui”, il vous sera proposé de voir les polices et cela vous permettra éventuellement de choisir celle qui vous plaît.

Ces options sont enregistrées dans `/etc/rc.d/rc.font`.

Si vous n'avez pas opté pour des polices personnalisées, vous pourrez le faire plus tard: soit en remplaçant/ajoutant une police dans /etc/rc.d/rc.font, ou bien en lançant le script `setconsolefont` :

```
setconsolefont
```



Note: `setconsolefont` ne montre qu'une liste préétablie de polices. Aucune des polices que vous ajouteriez n'apparaîtra dans cette liste.

Note: On peut aussi appeler `setconsolefont` depuis la section “setup” de `pkgtool`.

Note: Les polices pour la console se trouvent dans `/usr/share/kbd/consolefonts`. Vous pouvez aussi utiliser un chemin d'accès absolu en argument.

Changement provisoire de police

Vous pouvez changer de façon temporaire la police de la console grâce à l'utilitaire `setfont` (qui fait partie du paquet kbd). Si vous spécifiez une police non-Unicode dans un mode console Unicode, `setfont` va automatiquement essayer de mettre en place des tables de caractères Unicode si elles sont disponibles. Actuellement la plupart des polices comportent leurs tables Unicode associées. Pour de plus amples informations et d'avantage d'options, voyez la page de manuel de `setfont`.

*Notez que **`setconsolefont` fait également appel à `setfont` en interne**. donc ceci s'applique aussi à setconsolefont.*

```
setfont font
```

Quand elle est invoquée sans option, la commande setfont met en place la police par défaut:

```
setfont
```

Mise en place d'un plan de clavier personnalisé

Vous pouvez utiliser l'outil `setkeys` pour reconfigurer le clavier employé . Par exemple:

```
loadkeys hu
```

Les plans de claviers se situent dans `/usr/share/kbd/keymaps`. C'est là qu'on trouve

habituellement des indications sur les encodages/et correspondances supplémentaires avec certaines touches comme par exemple: “sr-cyr” qui est une disposition de clavier Cyrillique Serbe. C'est un type de clavier ISO-8859-5. La touche majuscule droite y est liée au basculement entre US English et Cyrillique Serbe. Un autre exemple est le clavier Russe (UTF-8, Ctrl+maj comme touche de basculement).

Les fichiers de description des claviers sont des textes en format compressé aisément lisibles avec zcat. Par exemple:

```
zcat /usr/share/kbd/keymaps/i386/azerty/fr-pc.map.gz | head | grep "^#"
```

Qui nous permet de lire toutes les indications dans les 10 premières lignes du fichier:

```
# fr-pc.map for French PC keyboard (non-US 102 keys)
#
#  1997-3-5
#  Seak, Teng-Fong <seak@drfc.cad.cea.fr>
#
```

Consoles Unicode provisoires avec unicode_start/unicode_stop

Si vous voulez activer Unicode seulement pour un temps, c'est possible par l'appel des scripts `unicode_start` et `unicode_stop` (qui font partie du paquet kbd)

```
unicode_start
```

Cela enregistrera la définition de votre clavier courant dans `~/.kbd`, configurera la console virtuelle active pour UTF-8 et chargera la police par défaut (LatArCyrHeb-16). Vous pouvez aussi spécifier une police de votre choix en ajoutant ses paramètres à l'invocation du script:

```
unicode_start font
```

ou:

```
unicode_start font unimap
```

Pour la description des paramètres *font* et *unimap* ainsi que leur emplacement par défaut, voyez le manuel de `setfont`:

```
man setfont
```

Et pour revenir à votre environnement précédent:

```
unicode_stop
```

Diagnostics

Vous pouvez vérifier si votre console est bien configurée pour Unicode.

Lancez stty avec l'option “-a”:

```
stty -a
```

Si vous voyez “-iutf8” parmi les résultats, la console virtuelle active n'est pas configurée pour UTF-8, au contraire, si vous voyez “iutf8” (sans le signe moins), la console virtuelle *vc* est correctement configurée.

Lancez kbd_mode sans argument:

```
kbd_mode
```

Vous obtiendrez le mode du clavier employé en mode console (hors X11), qui est généralement XLATE or UTF-8.

Locales pour X11

Quand votre ordinateur démarre en *niveau de fonctionnement 4* et emploie un des gestionnaires de connexion (par exemple: gdm, kdm ou xdm), le fichier `~/.profile` est consulté dans tous les cas. Par conséquent, si votre système utilise les réglages “par utilisateur” (comme cela est décrit ci-dessus) pour $LANG, ces réglages restent valables dans X11. Les gestionnaires de fenêtres et les environnements de bureaux (sauf KDE) se présenteront avec leurs menus dans votre langue.

Quand vous lancez `startx` depuis le *niveau 3* vos réglages liés à $LANG pour la console, s'appliquent au gestionnaire de fenêtres.

Réglage du clavier dans X11

Ce qui est affiché quand on utilise le clavier dans l'environnement X-Window est indépendant de l'affichage-clavier obtenu dans le mode console-texte. Les réglages indiqués ci-dessous ne dépendent pas d'un bureau.

À retenir: quelques environnements de bureaux proposent leurs propres réglages de localisation qui de ce fait prennent le pas sur les réglages présentés ici. L'exemple le plus notable est KDE.

Si vous voulez un affichage de clavier autre qu'anglais avec X, il vous faudra adapter les fichiers de configuration. Les fichiers génériques sont dans `/usr/share/X11/xorg.conf.d/` et c'est depuis cet emplacement que vous aurez à copier ceux que vous voulez employer, vers le répertoire `/etc/X11/xorg.conf.d/`.
Le serveur X ne lit que ces fichiers de configuration là, ceux de l'intérieur du répertoire /etc/X11/xorg.conf.d/ !

```
cp /usr/share/X11/xorg.conf.d/90-keyboard-layout.conf /etc/X11/xorg.conf.d
```

Ensuite ouvrez le fichier avec votre éditeur préféré et trouvez la ligne:

```
Option "XkbLayout" "us"
```

Remplacez “us” par le code de votre langue, par exemple “it” pour l' italien. De plus, vous pouvez éditer la ligne:

```
Option "XkbVariant ""
```

vous pouvez aussi ajouter votre 'keyboard-variant', par exemple “nodeadkeys”

Employer plusieurs localisations de clavier

Dans quels cas est-ce utile?

- Plusieurs personnes utilisent le même ordinateur mais elles préfèrent des dispositions de clavier différentes.
- Des personnes qui emploient une autre écriture (ex. le grec ou l'alphabet cyrillique), elles ont toujours besoin de basculer vers le clavier anglais pour saisir une commande.

Par conséquent, cette configuration peut avoir une grande importance même si beaucoup d'utilisateurs n'en auront jamais besoin.

Vous pouvez aussi indiquer plus d'une disposition de clavier:

```
Option "XkbLayout" "us,rs"
```

On peut également préciser les différentes variantes utilisées:

```
Option "XkbLayout" "us,rs,rs"
Option "XkbVariant ",latin,"
```

Dans cet exemple on configure trois types de claviers - anglais US, serbe latin et serbe cyrillique. Remarquez le même nombre de virgules que dans l'option `XkbLayout`. Les “champs” vides indiquent qu'on utilise les variantes par défaut pour les plans de clavier dont la variante n'est pas précisée.

Changer facilement d'affichage

Vous pouvez aussi indiquer un raccourci pour changer aisément de clavier.

```
Option "XKbOptions" "grp:alt_shift_toggle"
```

Qui permet de changer le plan de clavier en pressant simultanément Alt+Shift.

On peut aussi établir d'autres combinaisons servant de raccourcis-clavier [1)]:

- grp:toggle - alt de droite (alt graphic)
- grp:lalt - alt de gauche
- grp:shifts_toggle - les deux touches shift enfoncées en même-temps
- grp:alts_toggle - les deux touches alt enfoncées en même-temps
- grp:ctrl_shift_toggle - ctrl + shift
- grp:alt_shift_toggle - alt + shift

- grp:alt_space_toggle - alt + barre d'espace
- grp:ctrl_alt_toggle - ctrl + alt
- grp:caps_toggle - verrouillage des majuscules
- grp:shift_caps_toggle - shift + verrouillage des majuscules
- grp:lwin_toggle - touche Win de gauche sur claviers “Windows”
- grp:rwin_toggle - touche Win de droite sur claviers “Windows”
- grp:menu_toggle - touche menu de contexte sur claviers “Windows”
- grp:lctrl_toggle - ctrl de gauche
- grp:rctrl_toggle - ctrl de droite*

Modifier le plan de clavier par setxkbmap

Les indications suivantes s'adressent aux utilisateurs qui ont besoin de réglages “par utilisateur” pour leur plan de clavier dans X11. Ce qui est le cas lorsque les réglages globaux pour X11 ne correspondent pas à leurs besoins et qu'on ne peut pas les changer. Lisez la page de manuel pour de plus amples détails.

Pour prendre connaissance de vos réglages actifs dans la disposition de clavier pour X11 appelez la commande

```
setxkbmap -query
```

Un aperçu de `setxkbmap` est disponible à

```
setxkbmap -help
```

La langue et la variante de clavier peuvent être facilement configurées ainsi (l'exemple qui suit concerne l'allemand et la variante “nodeadkeys”)

```
setxkbmap de
setxkbmap -variant nodeadkeys
```

Où mettre le code?

Si votre système démarre en **runlevel 3**, c'est à dire que vous vous identifiez en mode console puis tapez startx, vous pouvez mettre le code indiqué précédemment au début de votre fichier **~/.xinitrc**. Si votre système démarre en **runlevel 4** et que votre identification est faite par un gestionnaire comme gdm, kdm ou xdm, vous devrez placer ce code dans votre **~/.xprofile**. Ce fichier est consulté par chacun des gestionnaires d'identification. Notez bien que `~/.xprofile` n'est pas créé par défaut, c'est à vous de le créer.

Environnements de bureau

Les environnements de bureau, comme tous les autres programmes, s'adapteront eux-mêmes par défaut à la locale que vous avez définie avant de lancer votre session X-Window. Cependant, vous n'installez pas les paquets de localisation pour KDE. L'installation standard de KDE se fait seulement

en anglais et on dispose des paquets de localisation de l'environnement KDE dans les séries de paquets **KDEI**. C'est là que vous trouverez les paquets de localisation pour Calligra également - la Suite bureautique de KDE.

Localisation de KDE

Quand vous utilisez KDE il vous faut installer le paquet choisi dans le groupe des paquets 'kde-l10n' pour votre langue. Les paquets l10n de KDE se trouvent dans les séries de paquets kdei. Si vous vous procurez Slackware depuis un serveur miroir, évitez de prendre la totalité des paquets l10n, téléchargez uniquement ceux qu'il vous faut, il y en a un pour KDE un autre pour Calligra, pour chaque langue.

Note: Les réglages suivants ne sont pas nécessaires si vous lancez la commande `startx` en runlevel 3 avec vos réglages de langue déjà mis en place. Dans ce cas KDE adopte les réglages de votre environnement.

Quand le paquet l10n est installé et que vous en êtes au premier démarrage de KDE, ouvrez le lanceur d'applications et cliquez sur *System Settings*. Dans le panneau de System Settings cliquez sur Locale et choisissez votre pays dans le menu déroulant. Ensuite utilisez les différents onglets pour déterminer votre langue, le format numérique, le format monétaire, le format de calendrier, la présentation des dates et de l'heure et dans “other” les choix de dimensions de page et l'ensemble d'unités de mesure adéquates.
Vous pouvez également configurer la vérification orthographique dans cette fenêtre.
Quand KDE est configuré pour votre langue, la suite bureautique Calligra démarre automatiquement avec les bons réglages de langue, à condition que vous ayez installé le paquet l10n de Calligra.

Disposition de clavier

Il y a dans notre wiki une page consacrée à ce sujet: Keyboard Layout in KDE

Localisation de Xfce

XFCE n'a pas d'outil graphique -GUI- pour régler son interface internationalisée. Il se conformera à vos réglages de locale tels qu'ils sont dans l'environnement. Également il n'y a pas de paquets distincts pour les données de la locale - tout est inclus dans les paquets principaux de XFCE. N'oubliez pas que la plupart des localisations de XFCE sont sérieusement en retard pour plusieurs langues très répandues. Pour en savoir plus sur comment aider à localiser XFCE dans votre langue, voyez Xfce's localization page.

Disposition du clavier

Veuillez consulter notre page de Wiki Keyboard Layout in XFCE

Dictionnaires et correcteurs orthographiques

Correcteurs orthographiques

Par défaut Slackware comporte aspell, ispell et hunspell. Le dictionnaire Anglais est déjà installé. Pour les autres langues vous trouverez les dictionnaires dans le DVD ou sur les serveurs miroirs dans le répertoire `/extra/aspell-word-lists/`.

Dictionnaires d'applications

Thunderbird et Firefox sont fournis avec des correcteurs orthographiques, vous pouvez installer le dictionnaire de votre langue au moyen du menu des “Add-Ons” (les modules complémentaires Mozilla).

Pour les personnes qui rédigent des courriels dans plusieurs langues, un “sélecteur de dictionnaires” est disponible pour Thunderbird. Vous le trouverez dans le menu des “Add-Ons” de Thunderbird.

Si vous avez installé LibreOffice depuis la page d'AlienBob, des paquets pour les langues y sont disponibles. Si vous avez installé le paquet l10n de votre langue, vous disposez alors du dictionnaire pour le correcteur orthographique.

Applications

Dans la Slackware la plupart des logiciels sont fournis avec le support multilingue d'origine. Leur comportement sera adapté à la locale que vous avez definie.
Il y a quelques rares exceptions remarquables. Les paquets Mozilla de Slackware pour Firefox, Thunderbird et Seamonkey ne sont pas localisés. Nous en traitons ci-dessous.

Certaines grosses applications logicielles tierces comme OpenOffice et LibreOffice sont livrées avec leur propre ensemble de paquets de langue.

Mozilla-Firefox

AlienBob fournit un fichier SlackBuild.script l10n qui semble bien être le moyen le plus commode de localiser Firefox.

On peut également procéder comme pour Mozilla-Thunderbird (voir ci-dessous). Après l'installation du fichier xpi vous tapez `about:config` dans la zone d'adresse de Firefox afin d'ouvrir le *config-editor*. Là vous saisissez 'useragent' et vous changez le *useragent.local* suivant votre langue. Ceci est expliqué plus en détail pour Thunderbird.

Mozilla-Thunderbird

Si vous voulez que les menus du logiciel de courrier Thunderbird soient dans votre langue, il vous faut

télécharger le fichier xpi pour votre langue depuis le site Mozilla.

Le problème avec Thunderbird, c'est qu'il faut pratiquement un nouveau fichier de localisation pour (presque) chaque version. Remarquez la structure des répertoires du site Mozilla. Pour chaque version il vous faudra suivre leur structure de répertoires depuis ftp://ftp.mozilla.org/pub/thunderbird/releases/ jusqu'à **/NUMER0_DE_VERSION/LINUX-i686 ou x86_64/xpi/**

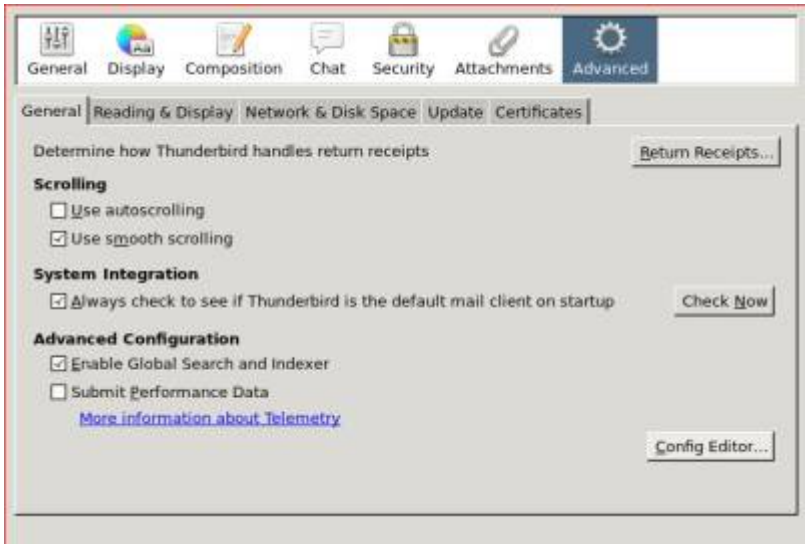
Les auteurs ont séparé les fichiers xpi suivant leurs versions 32bit et 64bit , mais je (AlienBob) les ai essayés et ces fichiers fonctionnent pour les deux versions.

L'installation d'un fichier xpi se fait “par-utilisateur”. Ouvrez dans le menu Tools→Add ons puis cliquez sur la petite flèche en haut à droite de la page, à côté du champ de recherche.

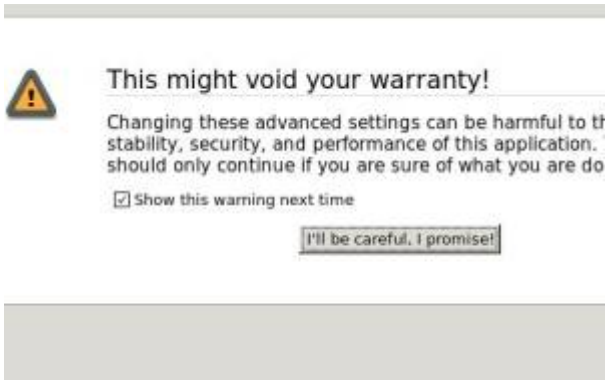
Choisissez Install Add on from file... et allez jusqu'au fichier xpi téléchargé.

Ensuite, il faut redémarrer Thunderbird.

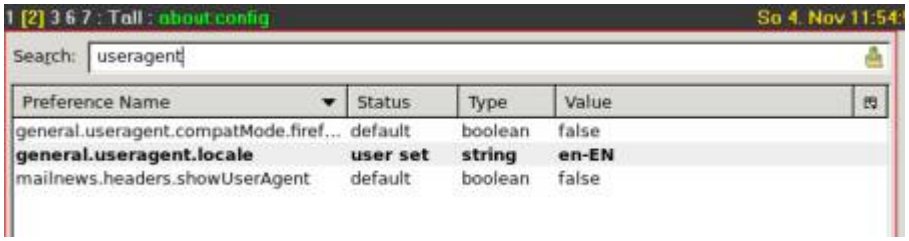
Il suffit de faire la dernière étape une seule fois pour chaque compte utilisateur, et le changement reste valide même si vous mettez à jour Thunderbird. Il vous faut éditer la configuration. Dans le menu de Thunderbird, ouvrez key>Edit</key> choisissez dans l'option Preferences l'onglet Advanced.



Ouvrez Config Editor... et cliquez sur le bouton I'll be careful, I promise!.



Alors l'éditeur de configuration s'ouvre et vous pouvez saisir `useragent` dans le champ du haut de la page.



Dans la ligne `general.useragent.locale` double-cliquez sur en-EN

et changez pour votre langue, par exemple fr-FR.

Ensuite redémarrez Thunderbird, maintenant vous devez avoir les menus dans votre langue.

Seamonkey

Vous trouverez un répertoire *langpack* sur le serveur miroir de Mozilla dans le répertoire pour votre édition. Ce sont des fichiers xpi qui s'installent de façon identique à l'installation des fichiers xpi de Thunderbird (décrite précédemment). Après avoir installé le fichier xpi, tapez `about:config` dans le champ d'adresse du navigateur Seamonkey afin d'ouvrir l'éditeur de configuration *config-editor*. Là vous saisissez 'useragent' et changez le *useragent.local* pour votre langue. Ceci est expliqué plus en détail ci-dessus pour Thunderbird.

LibreOffice

AlienBob fournit les paquets-Slackware pour LibreOffice et en complément les l10n-packages pour lui.

Lectures utiles

Pour ceux d'entre vous qui sont intéressés par davantage d'informations, voici quelques liens. Remerciements à Didier Spaier pour ses contributions.

- opengroup.org Character Set
- opengroup.org Locale
- opengroup.org Environment Variables
- University of Michigan - Primer on Unicode
- utf-8.com UTF-8 and Unicode Standards
- linuxdocs.org The Unicode HOWTO
- For Programmers: Locales in GLIBC
- unicode.org FAQ
- w3.org Unicode Character Pickers
- Data on Languages
- Dugan Chen's Homepage - Getting Unicode/UTF-8 support - Slackware related

Sources

- Originally written by Eric Hameleers
- Contributions by Bojan Popovic

- Contributions by Markus Hutmacher

slackware, localization, author alienbob, translator pierreaverseng
[1)]

Consultez la liste complète des options dans `/usr/share/X11/xkb/rules/base.lst`

From:
https://docs.slackware.com/ - **SlackDocs**

Permanent link:
https://docs.slackware.com/fr:slackware:localization

Last update: **2018/04/09 12:56 (UTC)**