

Petit guide d'installation d'Archlinux avec Gnome 3.18.x / Plasma 5.4.x / Xfce / Mate Desktop / Lxde / LXQt

Dans ce petit guide, je vais détailler l'installation d'Archlinux avec Gnome 3.18 et suivante, Plasma 5.4.x, Xfce, Mate Desktop, Lxde et LXQt. L'installation terminée proposera un environnement suffisamment étoffé pour être utilisable. C'est une adaptation des précédents guides.

Pour des raisons pratiques, je n'aborde nullement l'ajout de matériel comme les imprimantes, les scanners, ou encore les webcams. Je vous renvoie aux wikis anglophone <https://wiki.archlinux.org/> et francophone <http://wiki.archlinux.fr/Accueil> pour ce genre de manipulations.

NOTE 1 : ce tutoriel est **volontairement simplifié**. Il va vous permettre de voir comment installer une ArchLinux **en solo**. **Ensuite, s'il y a des spécificités liées à votre matériel, c'est au cas par cas qu'il faut regarder et compiler frénétiquement les wikis ci-dessus**. Si vous voulez installer une Archlinux **en parallèle** d'une installation de MS-Windows, c'est en dehors du cadre de ce document.

NOTE 2 : À compter du mois de septembre 2015, KDE SC 4.14.x n'est plus présenté dans le document. Pour les personnes désirant l'installer, il faudra se rapporter au tutoriel du mois d'août 2015, disponible à cette adresse : <http://frederic.bezies.free.fr/blog/?p=13134>

NOTE 3 : À compter du mois d'octobre 2015, Cinnamon 2.6.13 n'est plus présenté dans le document. Pour les personnes désirant l'installer, il faudra se rapporter au tutoriel du mois de septembre 2015, disponible à cette adresse : <http://frederic.bezies.free.fr/blog/?p=13282>

Pour cette version du guide, je me suis basé sur la dernière ISO officielle, celle qui utilise les scripts d'installation. En décembre 2015, c'est la 2015.12.01.

Merci à Ewolnux, Xarkam, Frédéric Sierra, Ludovic Riand, Vincent Manillier et Thomas Pawlowski pour leurs conseils et remarques. Ce document est proposé sous licence CC-BY-SA 3.0.

<http://creativecommons.org/licenses/by-sa/3.0/>

Ce tutoriel a été rédigé sous LibreOffice 5.0.3, avec VirtualBox 5.0.10, le tout avec une distribution Archlinux et Calibre pour la conversion au format ePub et mobi.

I. Installons notre base

Installer une Archlinux, c'est comme construire une maison. On commence par les fondations, et on rajoute les murs et le reste par la suite. L'image ISO utilisée est la archlinux-2015.12.01-dual.iso, mise en ligne début décembre 2015.

La machine virtuelle est une machine VirtualBox à laquelle j'ai rajouté un disque virtuel de 128 Go. Des points spécifiques concernant l'utilisation dans VirtualBox sont indiqués. Par défaut, le noyau proposé par Archlinux est un noyau « court terme ». Si vous voulez un noyau LTS, je vous expliquerai comment faire.

Dans cette partie, certaines sections seront dédoublées à cause des différences entre l'installation en mode Bios et en mode UEFI.

Commençons par une installation en mode Bios, du point de vue du partitionnement et de l'attribution des partitions. Si vous utilisez une machine réelle ou virtuelle avec l'UEFI, des instructions spécifiques sont détaillées par la suite.

1 Partitionnement et attribution des partitions en mode Bios :

Voici donc l'écran qui nous permet de démarrer. Comme vous pouvez le voir, on peut démarrer soit avec un noyau 32 soit avec un noyau 64 bits. Il n'y a plus grande différence, sauf au niveau de certains logiciels typiquement 32 bits, mais j'expliquerai plus bas comment faire cohabiter 32 et 64 bits.

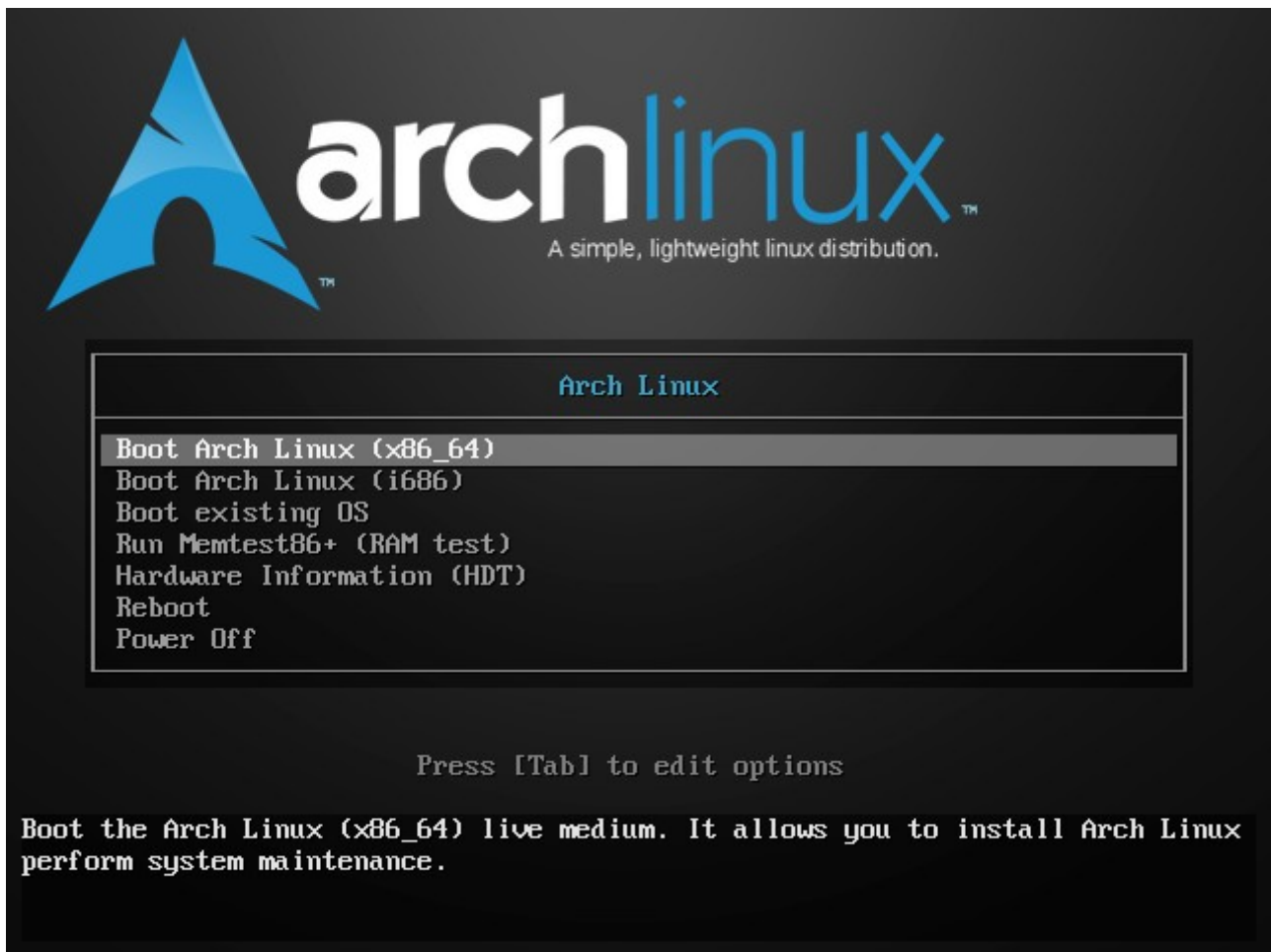


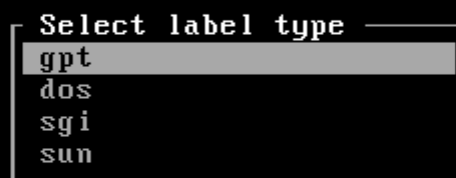
Illustration 1: écran de démarrage en mode Bios

La première chose à faire, c'est d'avoir le clavier français :

```
loadkeys fr
```

Pour le partitionnement, si vous avez peur de faire des bêtises, il est plus prudent de passer par un LiveCD comme gParted disponible à l'adresse : <http://gparted.org/>

Avec cfdisk, sur l'écran de démarrage suivant, on choisit l'option « dos » pour le « label type » à appliquer.



Device does not contain a recognized partition table.
Please, select a type to create a new disk label.

Illustration 2: premier démarrage de cfdisk

Pour le partitionnement en question :

Référence	Point de montage	Taille	Système de fichiers
/dev/sda1	/boot	512 Mo	ext2
/dev/sda2		Taille de la mémoire vive ou plus	swap
/dev/sda3	/	20 Go	ext4
/dev/sda4	/home	Le reste du disque	ext4

Il ne faut pas oublier de définir la partition attribuée à /boot comme démarrable (bootable). Ce qui donne l'écran suivant dans cfdisk.

```

Disk: /dev/sda
Size: 128 GiB, 137438953472 bytes, 268435456 sectors
Label: dos, identifier: 0xf7ea3727

Device      Boot    Start        End    Sectors    Size Id Type
/dev/sda1   *          2048     1050623     1048576    512M 83 Linux
/dev/sda2             1050624     9439231     8388608      4G 82 Linux swap / Solaris
/dev/sda3             9439232    51382271    41943040     20G 83 Linux
>> /dev/sda4          51382272    268435455    217053184   103.5G 83 Linux

Partition type: Linux (83)

[Bootable] [ Delete ] [ Quit ] [ Type ] [ Help ] [ Write ]
[ Dump ]

Write partition table to disk (this might destroy data)

```

Illustration 3: cfdisk en action

Pour le formatage des partitions, il suffit d'entrer les commandes suivantes :

```
mkfs.ext2 /dev/sda1
mkfs.ext4 /dev/sda3
mkfs.ext4 /dev/sda4
```

Sans oublier la partition de swap :

```
mkswap /dev/sda2
swapon /dev/sda2
```

On va ensuite créer les points de montage et y associer les partitions qui correspondent.

```
mount /dev/sda3 /mnt
mkdir /mnt/{boot,home}
mount /dev/sda1 /mnt/boot
mount /dev/sda4 /mnt/home
```

On peut passer ensuite à l'installation de la base.

2 Partitionnement et attribution des partitions en mode UEFI :

Voici donc l'écran qui nous permet de démarrer en mode UEFI. Sauf erreur de ma part, l'UEFI n'est pas supporté en mode 32 bits.

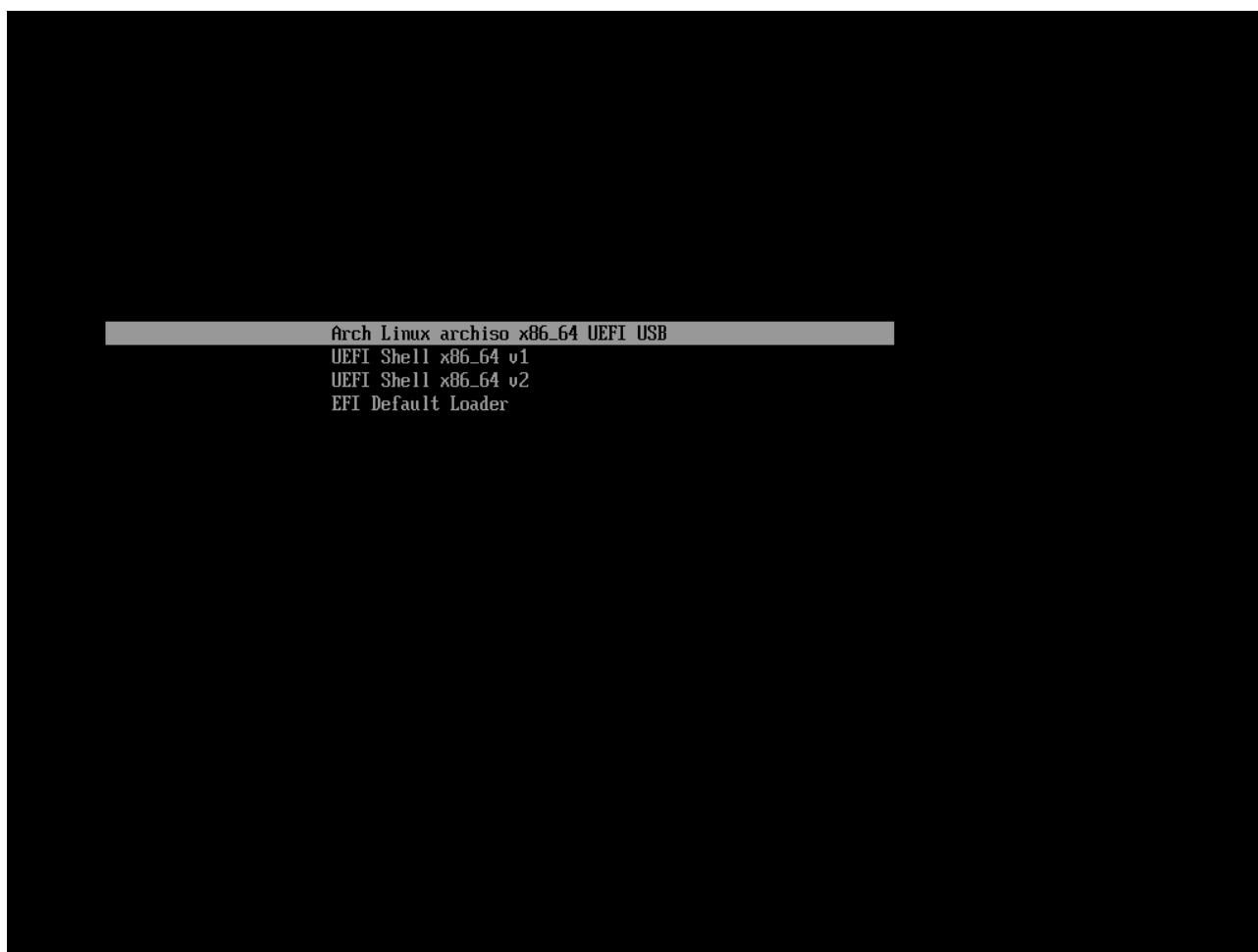


Illustration 4: démarrage en mode UEFI

Comme pour la section concernant le partitionnement en mode Bios, si vous craignez de faire des bêtises, vous pouvez utiliser gParted en mode liveCD : <http://gparted.org/>

Il faut se souvenir qu'il faut **obligatoirement** une table de partition GPT en cas d'installation en mode UEFI. Si vous n'êtes pas passé par gParted, il faut utiliser l'outil cgdisk.

Référence	Point de montage	Taille	Système de fichiers
/dev/sda1	/	20 Go	ext4
/dev/sda2	/boot	512 Mo	Fat32
/dev/sda3		Taille de la mémoire vive ou plus	swap
/dev/sda4	/home	Le reste du disque	ext4

Note : pour la partition /boot, il faut qu'elle soit étiquetée en EF00 à sa création. Pour le swap, c'est la référence 8200.

```
cgdisk 1.0.1

Disk Drive: /dev/sda
Size: 268435456, 128.0 GiB

Part. #   Size      Partition Type      Partition Name
-----
1         1007.0 KiB  free space
2         20.0 GiB   Linux filesystem
3         512.0 MiB   EFI System
4         4.0 GiB    Linux swap
4         103.5 GiB  Linux filesystem

[ Align ] [ Backup ] [ Delete ] [ Help ] [ Info ] [ Load ] [ name ] [ Quit ] [ Type ] [ Verify ]
[ Write ]

Change the filesystem type code GUID
```

Illustration 5: cgdisk en action pour un partitionnement avec un UEFI

Le partitionnement à appliquer ? C'est le suivant :

```
mkfs.ext4 /dev/sda1
mkfs.fat -F32 /dev/sda2
mkfs.ext4 /dev/sda4
```

Sans oublier la partition de swap :

```
mkswap /dev/sda3
swapon /dev/sda3
```

Et pour les points de montage :

```
mount /dev/sda1 /mnt
mkdir /mnt/{boot,home}
mount /dev/sda2 /mnt/boot
mount /dev/sda4 /mnt/home
```

On peut passer à l'installation de la base.

3 Installation de la base de notre Archlinux :

Après avoir procédé au partitionnement et à l'attribution des partitions, on peut attaquer les choses sérieuses, à savoir récupérer la base de notre installation. Mais avant toute chose, choisissons le miroir le plus rapide.

Note : si vous utilisez une connexion wifi, je vous conseille de voir cette page du wiki anglophone d'archlinux : <https://wiki.archlinux.org/index.php/Netctl>

Avec l'outil nano nous allons modifier le fichier /etc/pacman.d/mirrorlist pour ne garder qu'un seul miroir. Le plus proche géographiquement et aussi le plus rapide possible. Pour une personne vivant en France, c'est archlinux.polymorf.fr

```
GNU nano 2.4.3      File: /etc/pacman.d/mirrorlist
##
## Arch Linux repository mirrorlist
## Sorted by mirror score from mirror status page
## Generated on 2015-12-01
##
## Score: 0.3, France
Server = http://fooo.biz/archlinux/$repo/os/$arch
## Score: 0.3, France
Server = http://archlinux.polymorf.fr/$repo/os/$arch
## Score: 0.3, Germany
Server = http://mirror.pseudoform.org/$repo/os/$arch
## Score: 0.3, Denmark
Server = http://mirror.one.com/archlinux/$repo/os/$arch
## Score: 0.3, Germany
Server = http://mirror.hactar.bz/$repo/os/$arch
## Score: 0.4, Germany
Server = http://mirror.js-webcoding.de/pub/archlinux/$repo/os/$arch
## Score: 0.4, France
Server = http://arch.tamcore.eu/$repo/os/$arch
[ Read 471 lines ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text ^T To Spell ^_ Go To Line
```

Illustration 6: la liste des miroirs disponibles.

J'ai utilisé les raccourcis claviers : touches **Ctrl+W** puis **Ctrl+R**. Nano demandera la chaîne à chercher à savoir « Server » (sans les guillemets), puis celle à utiliser pour le remplacement, à savoir « #Server » (sans les guillemets). Une autre option est d'utiliser le raccourci clavier **ALT+R**.

On passe à l'installation de la base. La deuxième ligne rajoute certains outils bien pratique à avoir dès le départ.

```
pacstrap /mnt base base-devel
pacstrap /mnt zip unzip p7zip vim-minimal mc alsa-utils syslog-ng
```



```
mtools dosfstools lsb-release ntfs-3g (sur une seule ligne !)
```

Si on veut utiliser un noyau linux long terme, il faut rajouter à la deuxième ligne pacstrap le paquet linux-lts. Pour ntfs-3g, c'est utile si vous êtes amené à utiliser des disques formatés en ntfs. Si ce n'est pas le cas, vous pouvez l'ignorer allègrement.

Maintenant que les outils de base sont installés, on va passer aux choses sérieuses. Mais avant tout, il faut générer le fichier /etc/fstab.

```
genfstab -U -p /mnt >> /mnt/etc/fstab
```

On installe le chargeur de démarrage. Pour mon tutoriel, j'utilise Grub2 qui reconnaît automatiquement la version à installer (Bios ou UEFI). **Le paquet os-prober est indispensable si on installe Archlinux en dual-boot avec MS-Windows ou une autre distribution GNU/Linux.**

```
pacstrap /mnt grub os-prober efibootmgr
```

Note : le paquet efibootmgr est **indispensable** avec les ordinateurs avec UEFI.

On va passer aux réglages de l'OS qu'on vient d'installer. Il faut donc y entrer. On va utiliser la commande suivante :

```
arch-chroot /mnt
```

Avant d'aller plus loin, voici quelques infos pratiques. Cela concerne les pays francophones comme la Belgique, la Suisse, le Luxembourg ou encore le Canada francophone.

Nous allons par la suite créer des fichiers qui demanderont des valeurs précises. Les voici résumées ici :

Pays	Locale (LANG)	Clavier (KEYMAP)
Belgique	fr_BE.UTF-8	be-latin1
Canada	fr_CA.UTF-8	cf
France	fr_FR.UTF-8	fr-latin9
Luxembourg	fr_LU.UTF-8	fr-latin9
Suisse	fr_CH.UTF-8	fr_CH-latin1

Pour avoir le bon clavier en mode texte, créez le fichier /etc/vconsole.conf. Il suffira de l'adapter si le besoin s'en fait sentir.

```
KEYMAP=fr-latin9  
FONT=lat9w-16
```

Pour avoir la localisation française, le fichier `/etc/locale.conf` doit contenir la bonne valeur pour LANG. Pour un personne en France métropolitaine :

```
LANG=fr_FR.UTF-8
LC_COLLATE=C
```

Il faut vérifier que la ligne `fr_FR.UTF-8 UTF-8` dans le fichier `/etc/locale.gen` n'a pas de `#` devant elle. Ainsi que la ligne `en_US.UTF-8 UTF-8`. Évidemment, la valeur `fr_FR.UTF-8` doit être modifiée si besoin est. On va maintenant générer les traductions :

```
locale-gen
```

On peut spécifier la locale pour la session courante, à modifier en fonction de votre pays :

```
export LANG=fr_FR.UTF-8
```

Le nom de la machine ? Il est contenu dans le fichier `/etc/hostname`. Il suffit de taper le nom sur la première ligne. Par exemple : *fredo-archlinux-gnome*. À remplacer par le nom de la machine bien entendu.

Le fuseau horaire ? Une seule étape. Prenons le cas d'une installation avec le fuseau horaire de Paris. Tout dépend de votre lieu de résidence. On commence par créer un lien symbolique :

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

Ensuite, deux cas se présentent. Soit on a une machine en mono-démarrage sur Archlinux, et on peut demander à ce que l'heure appliquée soit UTC, soit un double démarrage avec MS-Windows. Restons dans ce premier cas.

```
hwclock --systohc --utc
```

Sinon, on ne touche à rien. MS-Windows est un goujat dans ce domaine.

Étape suivante ? Générer le bon fichier grub. Si vous voulez un noyau lts, il faudra utiliser `mkinitcpio -p linux-lts` en lieu et place du `mkinitcpio -p linux`.

```
mkinitcpio -p linux
grub-mkconfig -o /boot/grub/grub.cfg
```

Pour une installation en mode BIOS :

```
grub-install --no-floppy --recheck /dev/sda
```

Pour une installation en mode UEFI :

Il faut déjà vérifier la présence de efivars. Si en entrant la commande mount, vous avez dans la liste une réponse du style :

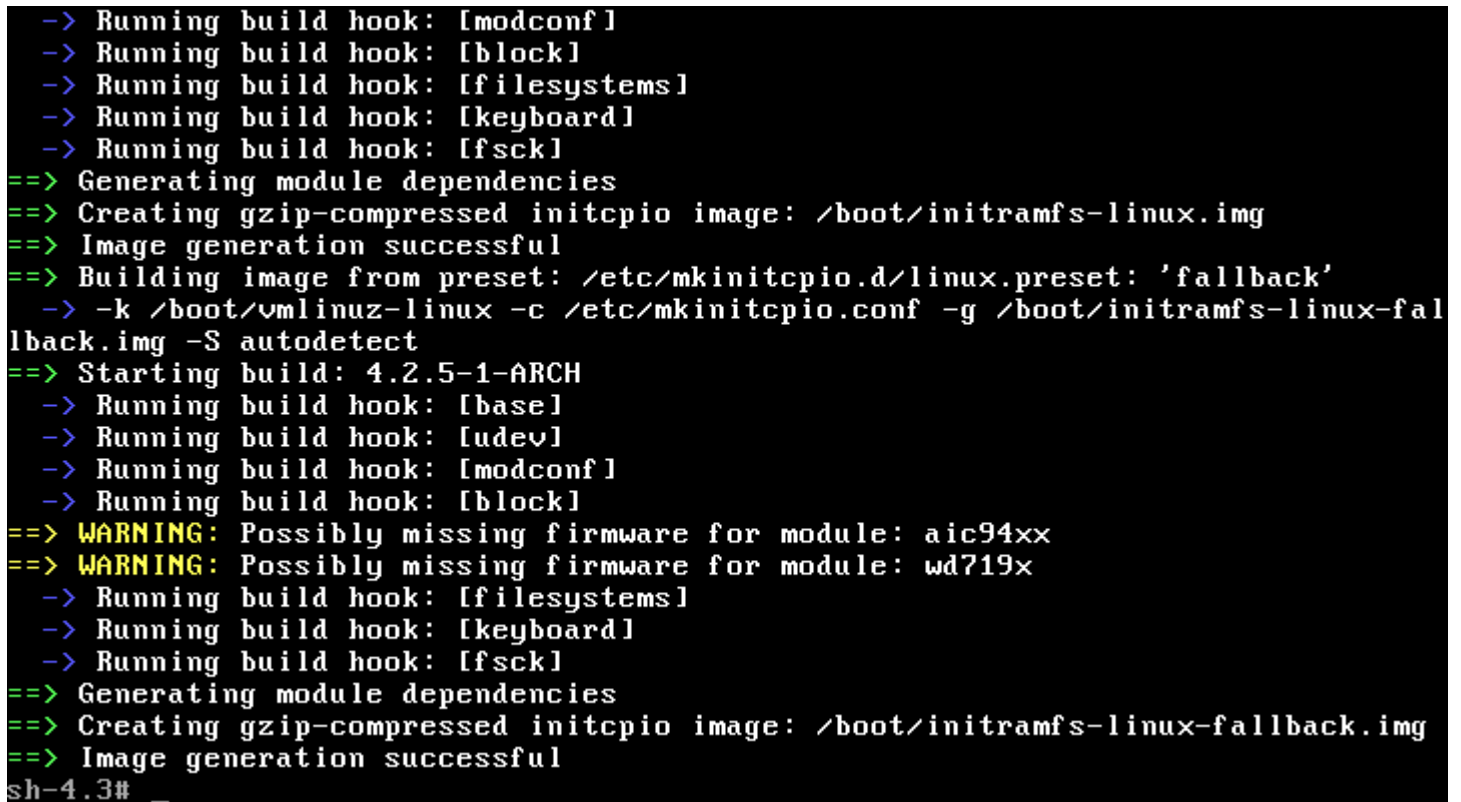
```
efivars on /sys/firmware/efi/efivars type efivars
(rw,nosuid,nodev,noexec,relatime)
```

Dans ce cas, vous pouvez aller directement à la ligne du grub-install, ci-dessous, et sauter la première ligne de commande.

```
mount -t efivarfs efivarfs /sys/firmware/efi/efivarfs
grub-install --target=x86_64-efi --efi-directory=/boot --bootloader-
id=arch_grub --recheck → sur une seule ligne !
```

De plus, pour éviter tout problème de démarrage par la suite, il est conseillé de rajouter les commandes suivantes :

```
mkdir /boot/EFI/boot
cp /boot/EFI/arch_grub/grubx64.efi /boot/EFI/boot/bootx64.efi
```



```
-> Running build hook: [modconf]
-> Running build hook: [block]
-> Running build hook: [filesystems]
-> Running build hook: [keyboard]
-> Running build hook: [fsck]
==> Generating module dependencies
==> Creating gzip-compressed initcpio image: /boot/initramfs-linux.img
==> Image generation successful
==> Building image from preset: /etc/mkinitcpio.d/linux.preset: 'fallback'
-> -k /boot/vmlinuz-linux -c /etc/mkinitcpio.conf -g /boot/initramfs-linux-fal
lback.img -S autodetect
==> Starting build: 4.2.5-1-ARCH
-> Running build hook: [base]
-> Running build hook: [udev]
-> Running build hook: [modconf]
-> Running build hook: [block]
==> WARNING: Possibly missing firmware for module: aic94xx
==> WARNING: Possibly missing firmware for module: wd719x
-> Running build hook: [filesystems]
-> Running build hook: [keyboard]
-> Running build hook: [fsck]
==> Generating module dependencies
==> Creating gzip-compressed initcpio image: /boot/initramfs-linux-fallback.img
==> Image generation successful
sh-4.3# _
```

Illustration 7 : Génération du noyau linux 4.2.5 début décembre 2015

Bien entendu, aucune erreur ne doit apparaître. On donne un mot de passe au compte root :

```
passwd root
```

Pour le réseau, étant donné qu'on va utiliser Gnome par la suite, installer et activer

NetworkManager est une bonne idée. Vous pouvez remplacer NetworkManager par le duo wicd et wicd-gtk **en cas de problème**. Pour wicd :

```
pacman -S wicd wicd-gtk
systemctl enable wicd
```

Et pour Networkmanager :

```
pacman -S networkmanager
systemctl enable NetworkManager
```

NOTE 1 : si vous n'utilisez pas NetworkManager, je vous renvoie à cette page du wiki anglophone d'Archlinux, qui vous aidera dans cette tâche : <https://wiki.archlinux.org/index.php/Netctl>

NOTE 2 : netctl et networkmanager rentrent en conflit et **ne doivent pas** être utilisés en même temps. D'ailleurs, netctl et wicd entre aussi en conflit. Une règle simple : un seul gestionnaire de connexion réseau à la fois !

NOTE 3 : si vous voulez utiliser des réseaux wifi directement avec NetworkManager et son applet, le paquet gnome-keyring est indispensable. Merci à Vincent Manillier pour l'info.

Dernier réglage, optionnel si on veut avoir accès à l'outil yaourt, il faut ajouter ceci au fichier /etc/pacman.conf à la fin. Une fois yaourt installé, on peut enlever **sans aucun risque** les lignes en question.

```
[archlinuxfr]
SigLevel = Never
Server = http://repo.archlinux.fr/$arch
```

Si vous voulez utiliser un outil comme Skype (qui est uniquement en 32 bits) et que vous installez un système 64 bits, il faut décommenter (enlever les #) des lignes suivantes :

```
#[multilib]
#Include = /etc/pacman.d/mirrorlist
```

On peut maintenant quitter tout, démonter proprement les partitions et redémarrer. C'est un peu plus délicat qu'auparavant. Mais au moins, on voit les étapes à suivre.

```
exit
umount -R /mnt
reboot
```

Voilà, on peut redémarrer. On va passer à la suite, largement moins ennuyeuse !

II. Installons maintenant l'environnement graphique !

Nous attaquons donc la partie la plus intéressante, l'installation de Gnome. C'est plutôt simple. Pour KDE SC, Xfce, Mate, Lxde ou LXQt, il faudra se référer aux addenda en fin de document, dès la section « Gnome ». **Toutes les étapes** jusqu'à l'installation de Gnome sont à suivre scrupuleusement, sauf celle en relation avec LibreOffice par exemple.

Note 1 : il y a une liste de services à activer un peu plus bas dans le document. Ils sont communs à **tous** les environnements du document.

Une fois le système démarré, on se connecte **en root**. Étant donné que j'ai installé NetworkManager (ou wicd selon les goûts) à l'étape précédente, le réseau fonctionne directement. On tape les lignes de commande suivantes pour mettre à jour les dépôts et installer yaourt un outil qui va nous simplifier grandement la vie.

Ainsi que ntp et cronie (pour les tâches d'administration à automatiser). L'installation de yaourt dépend de l'ajout du dépôt archlinuxfr, bien entendu.

Une fois yaourt installé (si on le désire), on peut enlever le dépôt archlinux du fichier /etc/pacman.conf car yaourt est disponible sur AUR.

```
pacman -Syy  
pacman -S yaourt ntp cronie
```

Note 2 : si on veut avoir les logs en clair en cas de problème, il faut modifier le fichier /etc/systemd/journald.conf en remplaçant la ligne :

```
#ForwardToSyslog=no
```

par :

```
ForwardToSyslog=yes
```

Les outils en place, on lance alsamixer, pour configurer le niveau sonore :

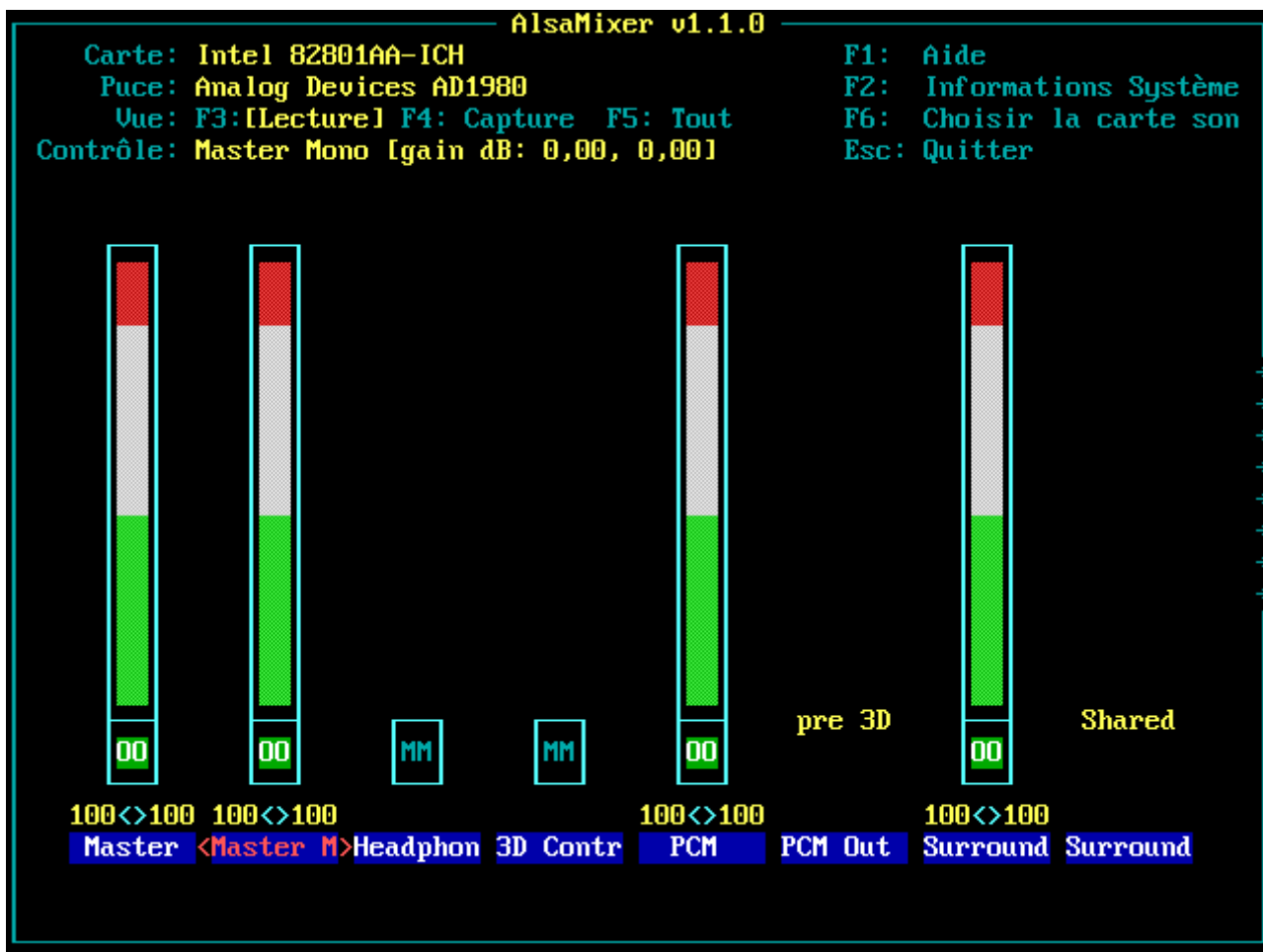


Illustration 8: alsamixer en action

Une fois l'ensemble configuré, pour le conserver tel quel, il suffit d'entrer :

```
alsactl store
```

Nous sommes dans le multimédia ? Restons-y.

On va installer l'ensemble des greffons gstreamer qui nous donneront accès aux fichiers multimédias une fois Gnome lancé. Si vous n'avez pas installé yaourt, il faudra le remplacer par **pacman -S** ou **sudo pacman -S** quand vous utiliserez votre compte utilisateur « normal ».

Pour l'exécution de la ligne suivante, il est demandé de choisir un support pour opengl. Pour le moment, on choisit MesaGL. La modification correspondant à votre matériel sera faite lors de l'installation de Xorg.

```
pacman -S gst-plugins-base gst-plugins-good gst-plugins-bad gst-  
plugins-ugly gst-libav
```

gst-libav ? Il prend en charge tout ce qui est x264 et apparenté.

Passons à l'installation de Xorg. Il suffit de rentrer :

```
pacman -S xorg-server xorg-xinit xorg-xmessage xorg-utils xorg-server-  
utils xorg-apps xf86-input-mouse xf86-input-keyboard xdg-user-dirs
```

Si on utilise un ordinateur portable avec un pavé tactile, il faut rajouter le paquet `xf86-input-synaptics`.

Il faut ensuite choisir le pilote pour le circuit vidéo. Voici les principaux pilotes, sachant que le paquet `xf86-video-vesa` englobe une énorme partie des circuits graphiques, dont ceux non listés dans le tableau un peu plus loin. En cas de doute : https://wiki.archlinux.org/index.php/Xorg#Driver_installation

Pour Nvidia, c'est un casse-tête au niveau des pilotes propriétaires. Le plus simple est de se référer au wiki d'archlinux : <https://wiki.archlinux.org/index.php/NVIDIA>

Circuits graphiques	Pilotes libres	Pilotes non libres (si existant)
AMD	<code>xf86-video-ati</code>	Catalyst (disponible sur AUR, le dépôt utilisateur)
Intel	<code>xf86-video-intel</code>	
Nvidia	<code>xf86-video-nouveau</code>	Nvidia (cf le wiki d'archlinux) pour la version à installer en fonction de la carte graphique

Dans le cas d'une machine virtuelle, j'ai utilisé le paquet `xf86-video-vesa`. On passe ensuite à l'installation des polices. Voici la ligne de commande pour les principales.

```
pacman -S ttf-bitstream-vera ttf-liberation ttf-freefont ttf-dejavu
```

Note : pour les polices Microsoft, le paquet `ttf-ms-fonts`, elles sont sur le dépôt AUR, donc il faut utiliser `yaourt` pour les récupérer et les installer.

Cependant, depuis l'arrivée de Pacman 4.2, il est impossible de construire un paquet avec l'option `--asroot`. Donc vous devrez lancer la commande ci-dessous en tant qu'utilisateur simple.

```
yaourt -S ttf-ms-fonts
```

Si vous faites une installation dans VirtualBox, deux étapes supplémentaires. D'abord installer le paquet **`virtualbox-guest-utils`** ou **`virtualbox-guest-utils-lts`** (si vous utilisez un noyau LTS).

Ensuite, il faut créer un fichier **`/etc/modules-load.d/vbox.conf`** qui contient les lignes suivantes :

```
vboxguest  
vboxvideo
```

```
vboxsfnet
```

Une autre possibilité est d'utiliser la commande systemctl suivante :

```
systemctl enable vboxservice
```

Note : si vous installez un jour VirtualBox sur une machine réelle je vous renvoie à cette page du wiki francophone : <https://wiki.archlinux.fr/VirtualBox>

Ensuite, on va rajouter des outils, histoire de ne pas voir un gnome vide au premier démarrage. On commence par tout ce qui est graphique : gimp, cups et hplip (si vous avez une imprimante scanner Hewlett Packard) :

```
pacman -S cups gimp hplip
```

Depuis la fin juillet 2014, il y a deux versions qui sont supportés par Archlinux pour LibreOffice, en conformité avec ce que propose la Document Foundation.

Pour la version stable et les utilisateurs prudents, on utilise la ligne de commande :

```
pacman -S libreoffice libreoffice-still-fr
```

Pour les utilisateurs plus aventureux, qui veulent la version récente :

```
pacman -S libreoffice-fresh libreoffice-fresh-fr
```

On rajoute ensuite Mozilla Firefox en français :

```
pacman -S firefox-i18n-fr
```

Vous préférez Chromium ?

```
pacman -S chromium
```

On crée un utilisateur avec la commande suivante, qui sera indispensable pour appliquer un des addenda si vous ne voulez pas utiliser Gnome.

```
useradd -m -g users -G wheel -c 'Nom complet de l'utilisateur' -s  
/bin/bash nom-de-l'utilisateur → sur une seule ligne !  
passwd nom-de-l'utilisateur
```

Avant de finir, on va configurer sudo en utilisant visudo. En effet, il nous suffit de modifier une ligne pour que l'on puisse accéder en tant qu'utilisateur classique aux droits complets sur la machine de manière temporaire.

Il faut aller jusqu'à la ligne :

```
##Uncomment to allow members of group wheel to execute any command
```

Et enlever le # sur la ligne qui suit. Modification à conserver, bien entendu !

On passe enfin au morceau de choix : l'installation de Gnome, les extensions étant indispensables pour avoir le mode « Gnome Classique ». Le paquet telepathy permet d'ajouter le maximum de support pour les comptes utilisateurs en ligne.

```
pacman -S gnome gnome-extra system-config-printer telepathy shotwell  
rhythmbox
```

L'installation de Gnome est maintenant terminée. Pour être certain d'avoir le bon clavier au démarrage de GDM, voici une petite commande à lancer (en modifiant le clavier selon les besoins), ce qui est valable pour tous les autres environnements :

```
sudo localectl set-x11-keymap fr
```

J'ai eu un retour, courant juin 2015 de la part de Ludovic Riand, comme quoi l'ajout de cette commande provoquait certains problèmes au niveau de Xorg. Pour contourner le problème, il faut créer le fichier /etc/X11/xorg.conf.d/10-keyboard-layout.conf et y insérer les valeurs suivantes :

```
Section "InputClass"  
    Identifier      "Keyboard Layout"  
    MatchIsKeyboard "yes"  
    Option          "XkbLayout"    "fr"  
    Option          "XkbVariant"   "latin9"  
EndSection
```

D'autres retours me disent que ce fichier n'est pas utile. À vous de voir !

Les valeurs étant à adapter en fonction de la locale et du clavier, bien entendu. Étant donné que systemd est utilisé, voici la liste des services à activer, qui sera la même pour chaque environnement dans les « addenda » :

```
systemctl enable syslog-ng  
systemctl enable cronie  
systemctl enable avahi-daemon  
systemctl enable avahi-dnssconfd  
systemctl enable org.cups.cupsd  
systemctl enable bluetooth → uniquement si on a du matériel bluetooth  
systemctl enable ntpd
```

Note : dans un premier temps, on ne va pas activer gdm. Juste le lancer avec **systemctl start gdm**. Si GDM se lance, et que Gnome se charge, on pourra l'activer sans danger par la suite.

Au démarrage suivant, GDM nous accueille, et nous pouvons nous connecter.

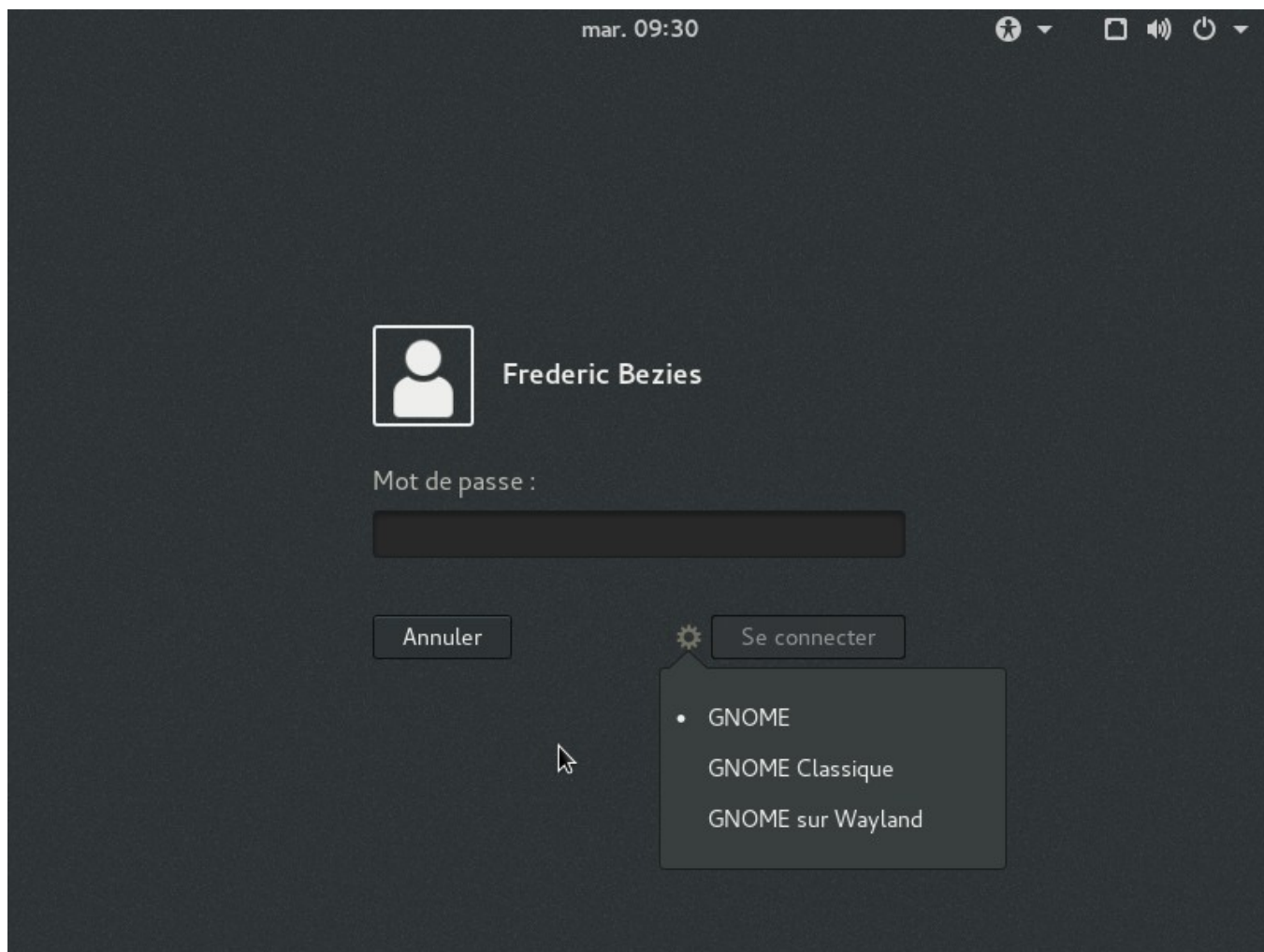


Illustration 9: GDM 3.18.2 avec l'expérimentale session Gnome sur Wayland

III. Finalisons l'installation de Gnome.

Note : à partir de maintenant, nous sommes connectés en tant qu'utilisateur classique.

Quelques outils à rajouter : xsane (pour le scanner), mais aussi unoconv (pour l'aperçu des fichiers dans Gnome Documents) et Adobe Flash... Même si ce dernier est accessoire et de moins en moins utilisé au fil des mois.

```
yaourt -S xsane unoconv flashplugin
```

Il faut penser à vérifier que le clavier est correctement configuré. Ce qui se fait dans menu système unifié, options de configuration.



Illustration 10: Gnome 3.18.2 en vue activités

Pour utiliser le mode Gnome Classique dans sa totalité, il faut activer l'option « Icônes sur le bureau » dans la catégorie Bureau de l'outil Gnome Tweak Tool.

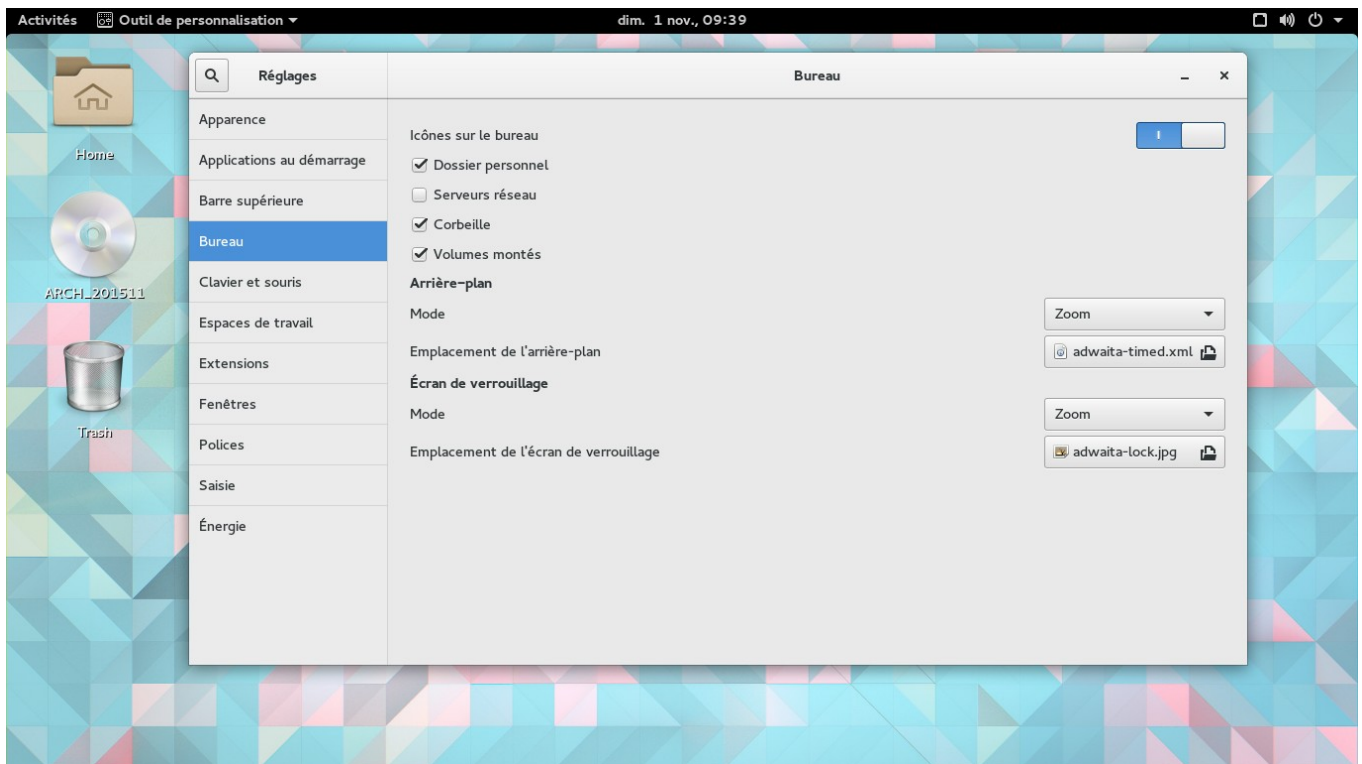


Illustration 11: Gnome Tweak Tool en action.

Pour finir des captures d'écran du mode « Gnome Shell » et « Gnome Classique ».

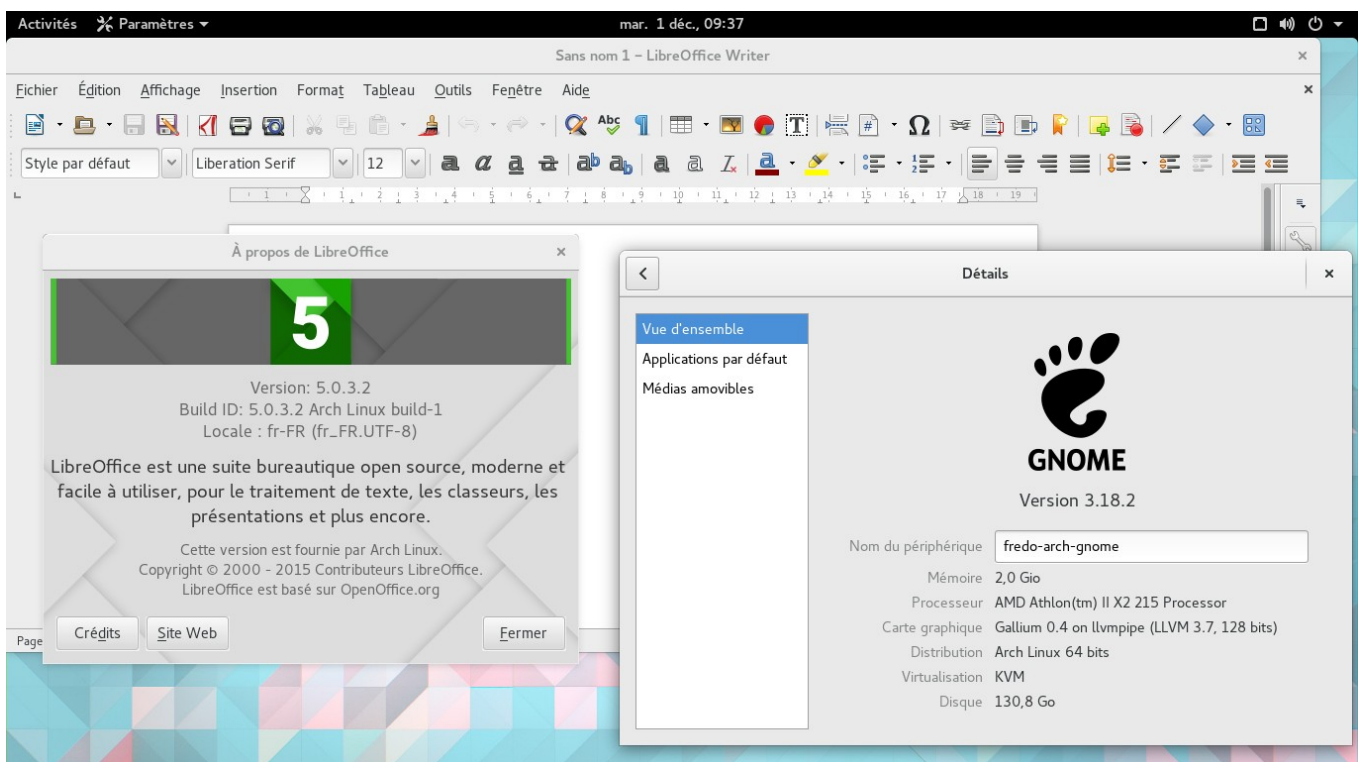


Illustration 12: Gnome Shell 3.18.2 et « LibreOffice-fresh »

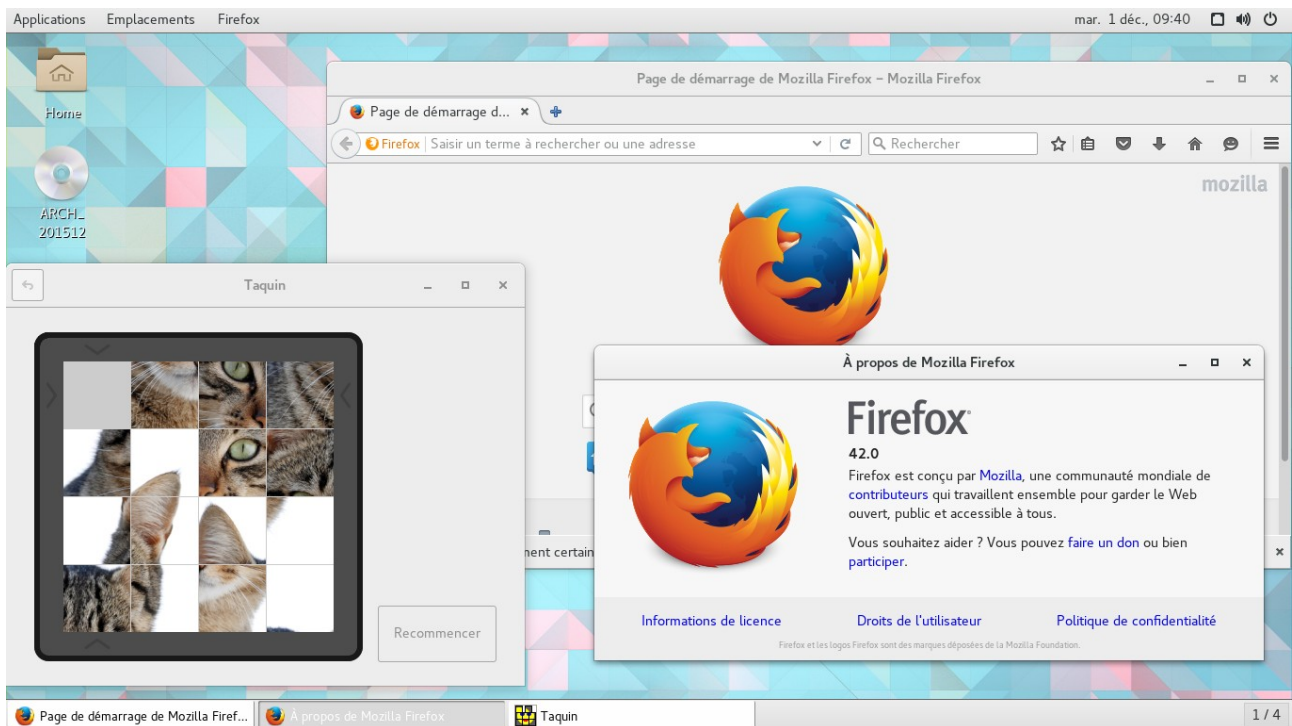


Illustration 13: Gnome classique 3.18.2 en action

Voilà, le guide est maintenant fini. Cependant, je n'ai pas abordé l'installation d'un pare-feu. C'est quelque chose de plus technique.

J'ai surtout voulu faire un **guide rapide**, histoire de montrer qu'en une petite heure on pouvait avoir un environnement installé et assez complet pour le finir par la suite.

Bonne découverte !

IV. Addendum 1 : installer Plasma 5.4.x

Note : commandes à entrer en tant qu'utilisateur classique.

Si vous voulez KDE SC 4.14.x, je vous conseille de vous reporter au tutoriel d'août 2015, comme cela est précisé en introduction. Pour Plasma 5, l'installation se déroule ainsi :

```
yaourt -S plasma-meta kde-applications amarok digikam kwebkitpart k3b  
breeze-kde4 kde-l10n-fr
```

Le paquet breeze-kde4 permet de donner un thème « KDE 5 » aux applications non encore adaptées pour le nouvel environnement.

Pour avoir le bon agencement clavier dès la saisie du premier caractère du mot de passe, il faut entrer la commande suivant avant de lancer pour la première fois sddm :

```
sudo localectl set-x11-keymap fr
```

Bien entendu, la valeur à utiliser après set-x11-keymap doit être identique à celle saisie plus haut quand on a configuré la base d'Archlinux. Sans oublier le correctif indiqué plus haut dans la section Gnome concernant le clavier français sous Xorg.

Pour lancer Plasma 5.4.x, dans un premier temps, il faut choisir la bonne session, à savoir Plasma, car **par défaut**, c'est le « media center » qui est sélectionné. Ce qui n'est pas des plus utiles pour une utilisation bureautique classique.

```
sudo systemctl start sddm
```

Si tout se passe bien, on peut utiliser :

```
sudo systemctl enable sddm
```

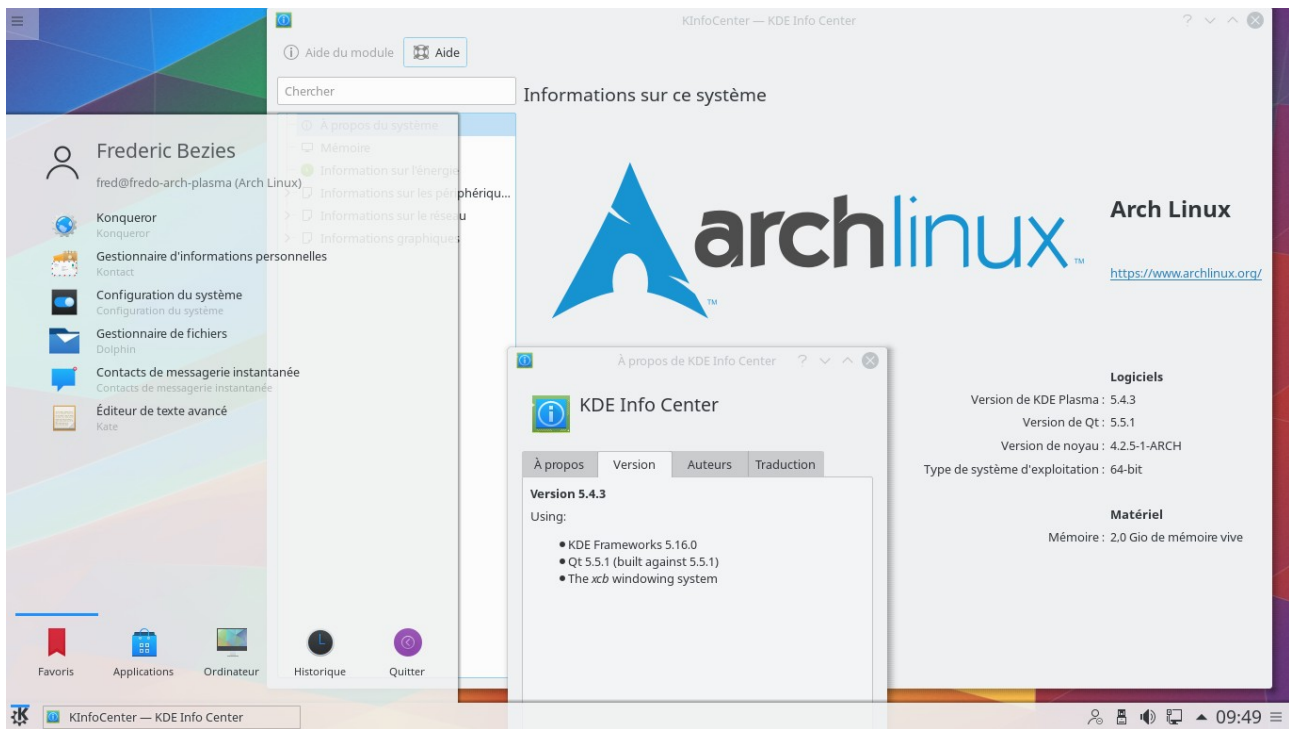



Illustration 14: Plasma 5.4.x – KDE Frameworks 5.16.0

V. Addendum 2 : installer Xfce

Note : commandes à entrer en tant qu'utilisateur classique.

Pour installer Xfce, il faut entrer :

```
yaourt -S xfce4 xfce4-goodies gvfs vlc quodlibet python2-pyinotify  
lightdm-gtk-greeter midori xarchiver claws-mail galculator evince  
ffmpegthumbnailer xscreensaver system-config-printer → (pour installer  
le support des imprimantes)
```

Vlc et Quodlibet ? Pour la vidéo et l'audio. Midori ? Pour la navigation internet. Pour les périphériques amovibles, gvfs est obligatoire. Claws-Mail ou Mozilla Thunderbird (avec le paquet thunderbird-i18n-fr) pour le courrier.

Lightdm étant pris, car plus rapide à installer. Le paquet python2-pyinotify est nécessaire pour activer le greffon de mise à jour automatique de la musicothèque sous Quodlibet.

Evince ? Pour les fichiers en pdf. On peut aussi remplacer xarchiver par fileroller. Quant à ffmpegthumbnailer, c'est utile si vous désirez avoir un aperçu des vidéos stockées sur votre ordinateur. Enfin, xscreensaver sert au verrouillage de l'écran.

Si vous désirez les notifications sonores, il faut rajouter pulseaudio et libcanberra. Si vous utilisez NetworkManager, vous pouvez rajouter l'applet pour gérer et surveiller votre réseau avec le paquet « network-manager-applet ».

Pour avoir le bon agencement clavier dès la saisie du premier caractère du mot de passe, il faut entrer la commande suivant avant de lancer pour la première fois lightdm :

```
sudo localectl set-x11-keymap fr
```

Pour lancer Xfce, il faut entrer dans un premier temps :

```
sudo systemctl start lightdm
```

Et si tout se passe bien, on peut utiliser :

```
sudo systemctl enable lightdm
```

Note 1 : pour avoir des plus jolies icônes, on peut installer le paquet AUR elementary-xfce-icons ou encore les mint-x-icons. Mais après, c'est à vous de voir !

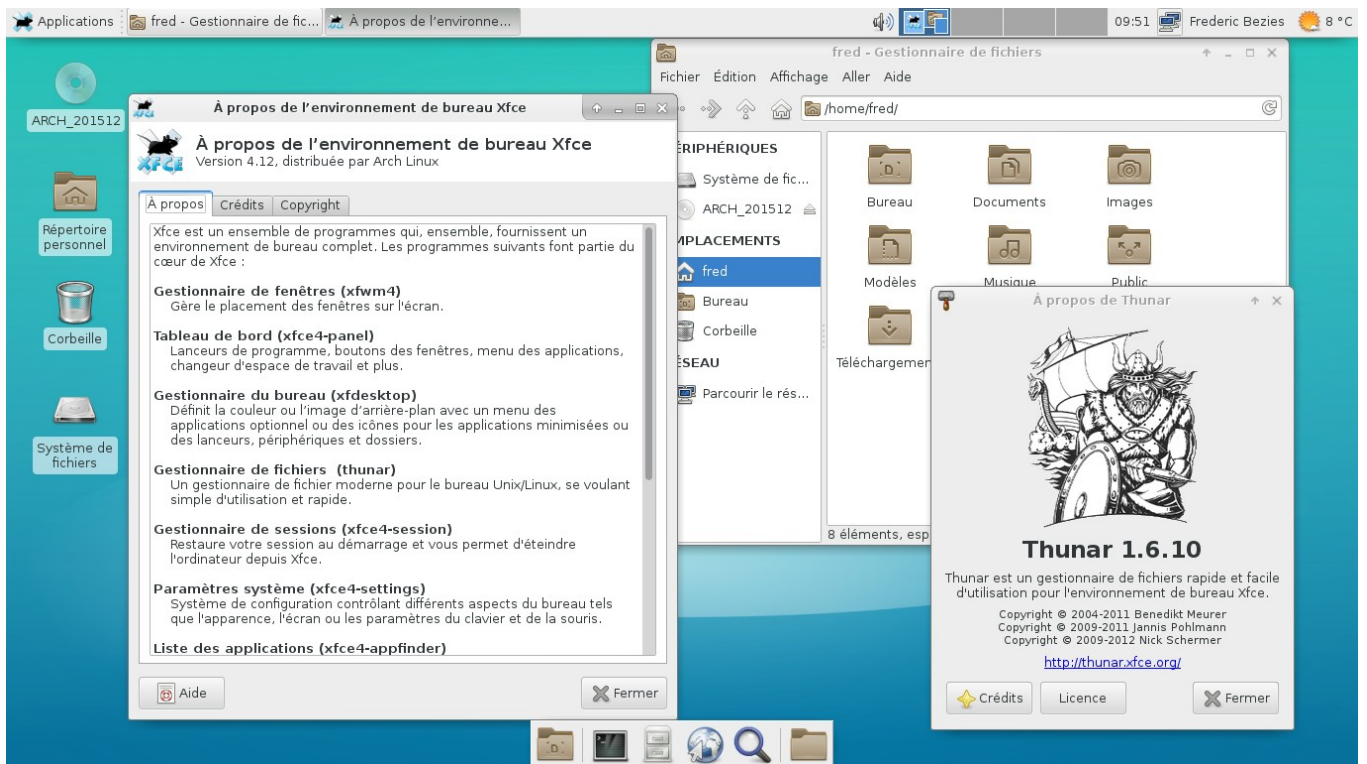


Illustration 15: Xfce 4.12.0 en action.

VI. Addendum 3 : installer Mate Desktop.

Note : commandes à entrer en tant qu'utilisateur classique.

L'installation ressemble à celle de Xfce, donc pour les explications des paquets, cf l'addenda consacré à Xfce. Idem pour l'utilisation de NetworkManager si vous le voulez. Il ne faut pas oublier de rajouter un outil de gravure, comme Brasero si nécessaire. Pour le navigateur, Mozilla Firefox, Chromium ou encore Midori. C'est selon les goûts !

```
yaourt -S mate mate-extra lightdm-gtk-greeter gnome-icon-theme vlc  
quodlibet python2-pyinotify accountsservice claws-mail system-config-  
printer → (pour installer le support des imprimantes)
```

Pour avoir le bon agencement clavier dès la saisie du premier caractère du mot de passe, il faut entrer la commande suivant avant de lancer pour la première fois lightdm :

```
sudo localectl set-x11-keymap fr
```

Pour lancer Mate Desktop, il faut entrer dans un premier temps :

```
sudo systemctl start accounts-daemon  
sudo systemctl start lightdm
```

Si tout se passe bien, on peut utiliser :

```
sudo systemctl enable accounts-daemon  
sudo systemctl enable lightdm
```

Note : il existe une version **expérimentale** avec gtk3. Elle s'installe en remplaçant mate et mate-extra par mate-gtk3 et mate-extra-gtk3 dans la ligne de commande d'installation.

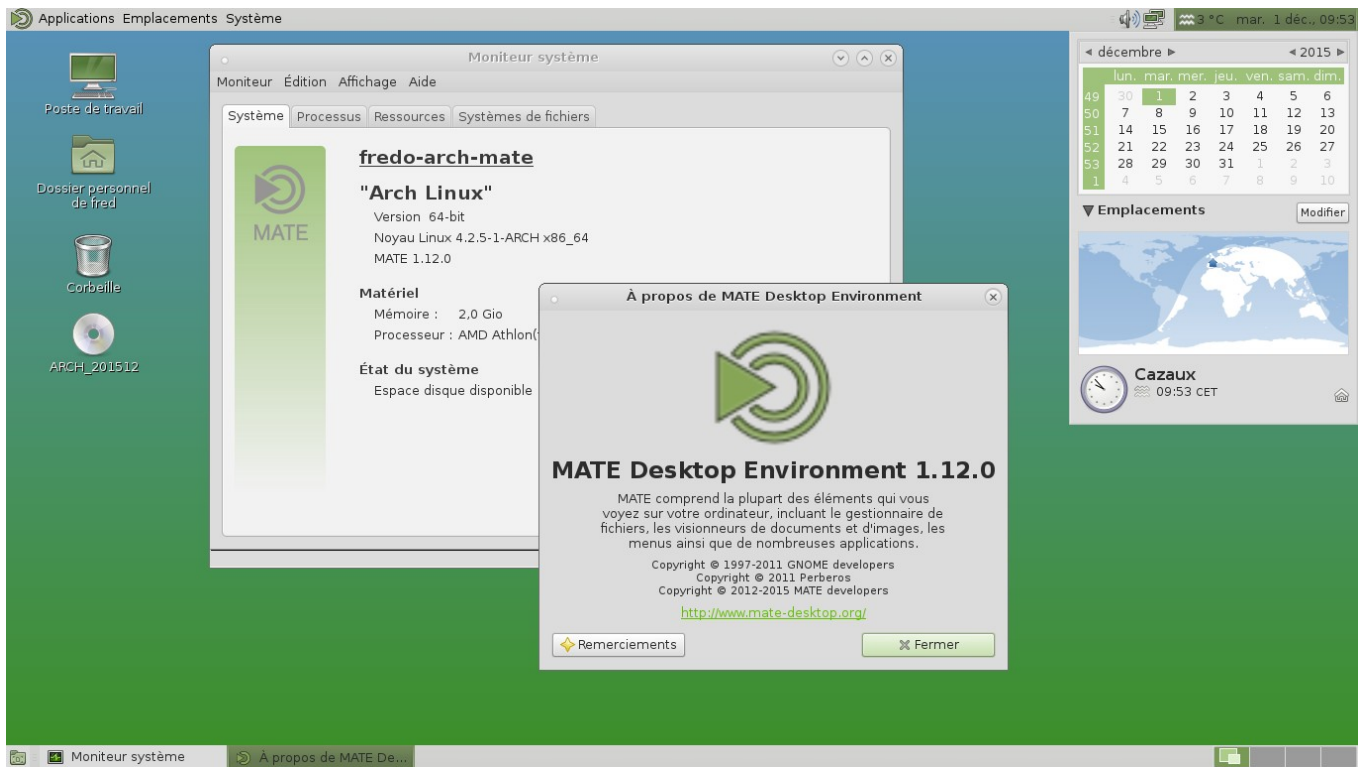


Illustration 16: Mate Desktop 1.12.0

VII. Addendum 4 : installer Lxde ou LXQt.

Note : commandes à entrer en tant qu'utilisateur classique.

J'ai regroupé les deux environnements légers qui sont tout de même assez proches. À noter que l'installation de Lxde est moins « complexe » que celle de LXQt qui demande la compilation de certains logiciels.

Comme pour l'installation de Xfce ou encore de Mate Desktop, j'ai pris quelques logiciels types. Libre à vous de remplacer Midori par Mozilla Firefox par exemple.

Commençons par Lxde. Son installation est assez simple. Pour cet environnement, on peut très bien remplacer yaourt par `sudo pacman`. Xterm est installé, car il est indispensable pour faire fonctionner *l'appliquette* de gestion du niveau du son.

```
yaourt -S lxde gvfs xarchiver mousepad midori claws-mail xscreensaver gnome-  
mplayer xterm system-config-printer → (pour installer le support des  
imprimantes)
```

Avant de lancer `lxdm`, il faut entrer la commande pour avoir le clavier français dans `lxde`, étant donné que l'on ne peut pas régler son agencement en mode graphique, donc :

```
sudo localectl set-x11-keymap fr
```

Pour lancer Lxde, il faut entrer dans un premier temps :

```
sudo systemctl start lxdm
```

Si tout se passe bien, on peut utiliser :

```
sudo systemctl enable lxdm
```

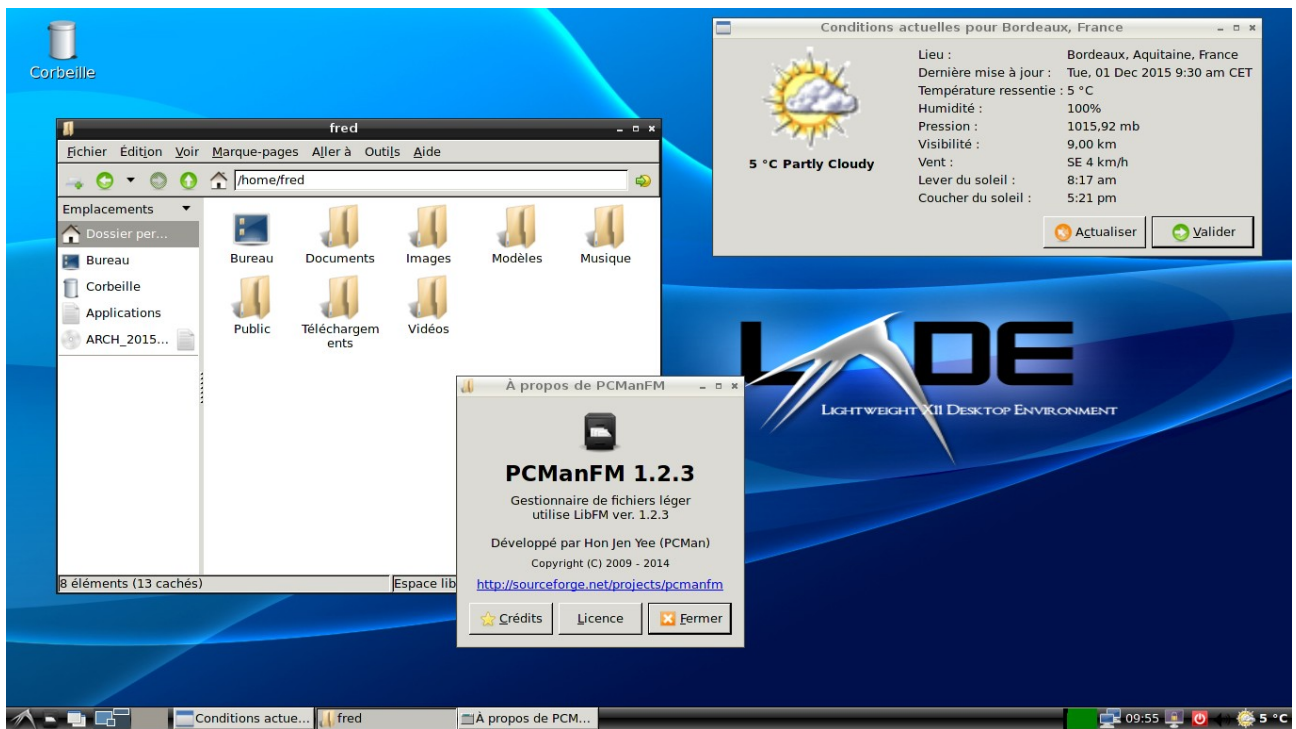


Illustration 17 : Lxde "0.99" en action.

L'installation de LXQt 0.10.0 est plus *chatouilleuse*. Elle se divise en deux parties. La première est facile, car il s'agit de paquets précompilés : il y a les icônes, le gestionnaire de tâches ou encore le navigateur internet et le lecteur audio.

Comme pour Lxde, on peut remplacer yaourt par sudo pacman sans problème. Cependant, il faut installer les deux séries avant de lancer LXQt, sinon le menu déroulant sera incomplet.

```
yaourt -S lxqt breeze-icons sddm qupzilla vlc xarchiver gvfs
xscreensaver qps qmmp libstatgrab lm_sensors xcursor-themes claws-mail
system-config-printer → (pour installer le support des imprimantes)
```

La deuxième partie des paquets nécessitent d'être récupérés depuis AUR, et donc yaourt va nous simplifier la tâche. Vous pouvez remplacer notepadqq-git par tea ou encore mousepad pour avoir un éditeur de texte basique. Qastools s'occupe de l'audio, lximage-qt des images et qterminal de proposer un terminal en Qt basique. Screengrab s'occupe de faire des captures d'écran.

```
yaourt -S qastools-qt5 lximage-qt qterminal notepadqq-git screengrab-
git
```

Ensuite, comme pour lxde, on passe le clavier dans le bon agencement avant de lancer pour la première fois sddm.

```
sudo localectl set-x11-keymap fr
```

Pour lancer LXQt, il faut entrer dans un premier temps :

```
sudo systemctl start sddm
```

Si tout se passe bien, on peut utiliser :

```
sudo systemctl enable sddm
```

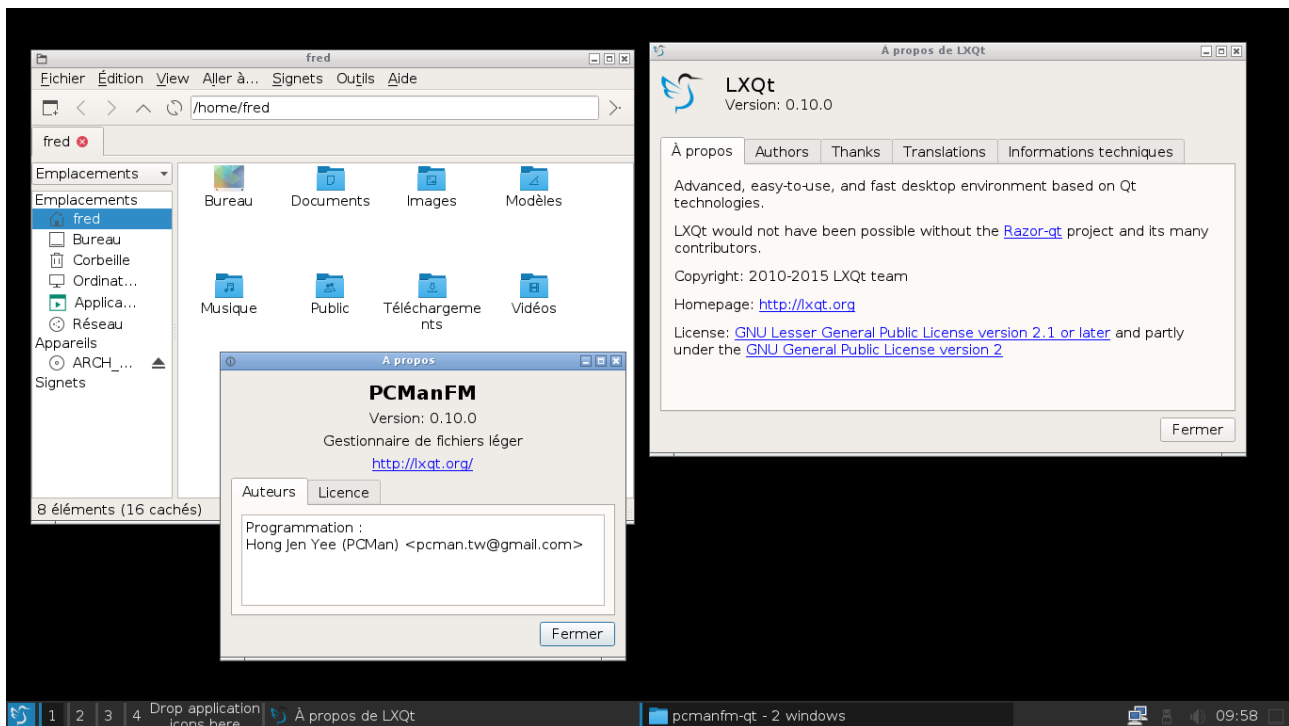


Illustration 18 : LXQt 0.10.0 en action.