

# Le RaspberryPi : Installation du système Gnu/Linux de base

Par X. HINAULT – Octobre 2012 – [www.mon-club-elec.fr](http://www.mon-club-elec.fr)



## Choix du système à installer

- Comme pour tout ordinateur, le raspberryPi nécessite un système d'exploitation pour être utilisable. Le système d'exploitation, c'est l'équivalent de Windows ou Gnu/Linux sur les PC, Mac Os X sur les Mac.
- Le RaspberryPi étant un matériel « low-cost » vendu à 30€, il est illusoire ( et idiot...) d'utiliser une système d'exploitation payant sur cette plateforme. Le choix logique et naturel est de se tourner vers une distribution Gnu/Linux, gratuite, opensource, configurable à volonté.
- Une des difficultés en ce qui concerne le RaspberryPi est le fait que **le processeur utilisé est un ARM** : toutes les distributions Gnu/Linux ne seront pas forcément compatibles, notamment Ubuntu par exemple. D'autre part **les contraintes matérielles réduisent aussi les possibilités** : avec 250 Mo de RAM et un processeur mono-coeur qui tourne à 700Mhz (l'équivalent d'un Pentium II... pour ceux qui connaissent), on sera obligé d'utiliser un système peu gourmand en ressources matérielles. Et c'est bien là que Gnu/Linux fait des miracles : il existe des gestionnaires de fenêtres (openbox par exemple) et des environnements de bureau (LXDE) très légers et non moins efficaces... Je ne vais pas m'étendre ici, mais **sachez que l'on va pouvoir installer grâce à Gnu/Linux un système graphique sur un raspberryPi qui n'utilisera que 40Mo de RAM au démarrage** soit moins du quart de la RAM disponible sur le RaspberryPi ! On pourra également ajouter, à la façon d'un lego logiciel, toutes sortes de fonctions utiles ( accès VNC, wifi, son, vidéo, synthèse, vocale, partage de fichier, etc... )
- On va utiliser un système Gnu/Linux, bien. Mais au pays de Gnu/Linux et du libre, il y a toujours pleins de possibilités... Les critères de choix du système vont être :
  - support du processeur ARM utilisé (condition incontournable!) avec support du calcul en virgule flottante (ça accélère le processeur)
  - maturité globale de la distribution avec une large documentation et un bon support de la communauté
  - attachement aux valeurs du logiciel libre ( à ce titre Android, bien que basé sur un noyau Linux, est à considérer à mon sens comme un système semi-libre)
  - nombre de paquets logiciels disponibles : plus il y en aura, mieux ça sera... et performance du gestionnaire paquet
  - légèreté du système minimal...
- Bon, une fois qu'on a dit ça, il reste encore pas mal de possibilités... Voici un listing du possible : [http://elinux.org/RPi\\_Distributions#Available\\_Distributions](http://elinux.org/RPi_Distributions#Available_Distributions)
- **Mon choix se portera sur la distribution Raspbian car basée sur Debian** (gestionnaire de paquet deb, 35000 paquets dispos, etc... ) Il se trouve que cette distribution est aussi celle conseillée en priorité par la fondation RaspberryPi. Sa mise en œuvre est par ailleurs assez simple comme nous allons le voir. Les habitués d'Ubuntu se sentiront par ailleurs en terrain connu une fois le système lancé. Le site de la distro est ici : <http://www.raspbian.org/>

## Préparation de l'installation

- La procédure est « classique » pour l'installation d'un système Linux : on grave une image sur un support de lancement, ici une carte mémoire SD puis on lance le système avec. Ceux qui ont déjà préparé un live-CD seront en terrain connu. Pour les autres, rien de bien sorcier.
- La procédure présente cependant quelques particularités liées :
  - à la cible utilisée, le RaspberryPi, plateforme ARM rappelons-le
  - au type de support (une carte mémoire SD) qui a ses particularités techniques.
- Je présente ici la procédure à partir d'un poste tournant sous Gnu/Linux. Si vous n'êtes pas sous Gnu/Linux, procurez-vous un live-CD Ubuntu par exemple et lancez votre système avec. Pour les détails, voir ici : [http://doc.ubuntu-fr.org/live\\_cd](http://doc.ubuntu-fr.org/live_cd) et là : [http://doc.ubuntu-fr.org/live\\_usb](http://doc.ubuntu-fr.org/live_usb)

### Matériel nécessaire

- Pour cette étape vous avez besoin uniquement d'un PC sous Gnu/Linux connecté à internet et disposant d'un connecteur pour carte SD ainsi qu'une carte mémoire SDHC de 8Go ou plus.

### Télécharger l'image du système Raspbian « Wheezy »

- Pour installer le système de base, le plus simple est de partir d'un système « tout prêt » qui est disponible sous la forme d'une grosse archive que l'on appelle « image ». On commence par télécharger l'image du système à installer ici : <http://www.raspberrypi.org/downloads> . On obtient ainsi un fichier \*.zip qui fait 400Mo environ. Ce fichier contient l'image au format \*.img . Choisir l'image **Raspbian « Wheezy »** (basée sur Debian « Wheezy » – 35 000 paquets compilés pour le Raspberry Pi disponibles ! )
- Pour info, l'autre façon de faire est d'utiliser un installateur qui chargera les paquets voulus directement sur le réseau. Mais nous ne détaillerons pas cette procédure ici.

### Vérifier l'intégrité de l'image téléchargée

- Une fois l'image téléchargée obtenue, il est conseillé et prudent de vérifier l'intégrité du fichier \*.zip. Commencer par noter la clé associée au fichier téléchargé qui est présentée sur la page de téléchargement où vous avez récupéré l'image. Dans mon cas, la page de téléchargement indique la valeur de somme de contrôle suivante SHA-1 Checksum : **3bc788d447bc88feaae8382d61364eaba1088e78**. Copier/coller cette valeur dans un simple éditeur de texte : dans Gedit par exemple sur un système Ubuntu.
- Ensuite, ouvrir un terminal (ligne de commande), se placer dans le répertoire de téléchargement de l'archive .zip et lancer la vérification de l'intégrité avec les commandes suivantes :

```
$ cd repertoire/de/telechargement/
```

```
$ sha1sum 2012-09-18-wheezy-raspbian.zip
```

- Cela dure quelques secondes. Vous devez obtenir un nombre hexadécimal assez long. Dans mon cas j'obtiens : **3bc788d447bc88feaae8382d61364eaba1088e78**. Copier/coller ce nombre dans l'éditeur où vous avez précédemment mis le nombre de vérification obtenu sur la page de Téléchargement. **Vérifier que les 2 nombres sont strictement identiques**. Si c'est bien le cas, le fichier \*.zip est valide et on peut passer à la suite. Sinon, recommencer le téléchargement et vérifier à nouveau l'intégrité du fichier obtenu.

## Extraire l'image à partir de l'archive obtenue

- Ensuite, nous allons extraire le fichier de l'image au format \*.img à partir de l'archive \*.zip précédemment téléchargée.

## Méthode graphique

- Ouvrir le gestionnaire de fichier, aller dans le répertoire où se trouve l'archive et faire un simple clic droit dessus : choisir « ouvrir avec le gestionnaire » d'archive puis « extraire » et « extraire ici », comme on le ferait pour n'importe quelle archive \*.zip.
- Attention, ça peut prendre un certain temps. Vous devez obtenir un fichier \*.img de 1,8Go environ.

## En ligne de commande

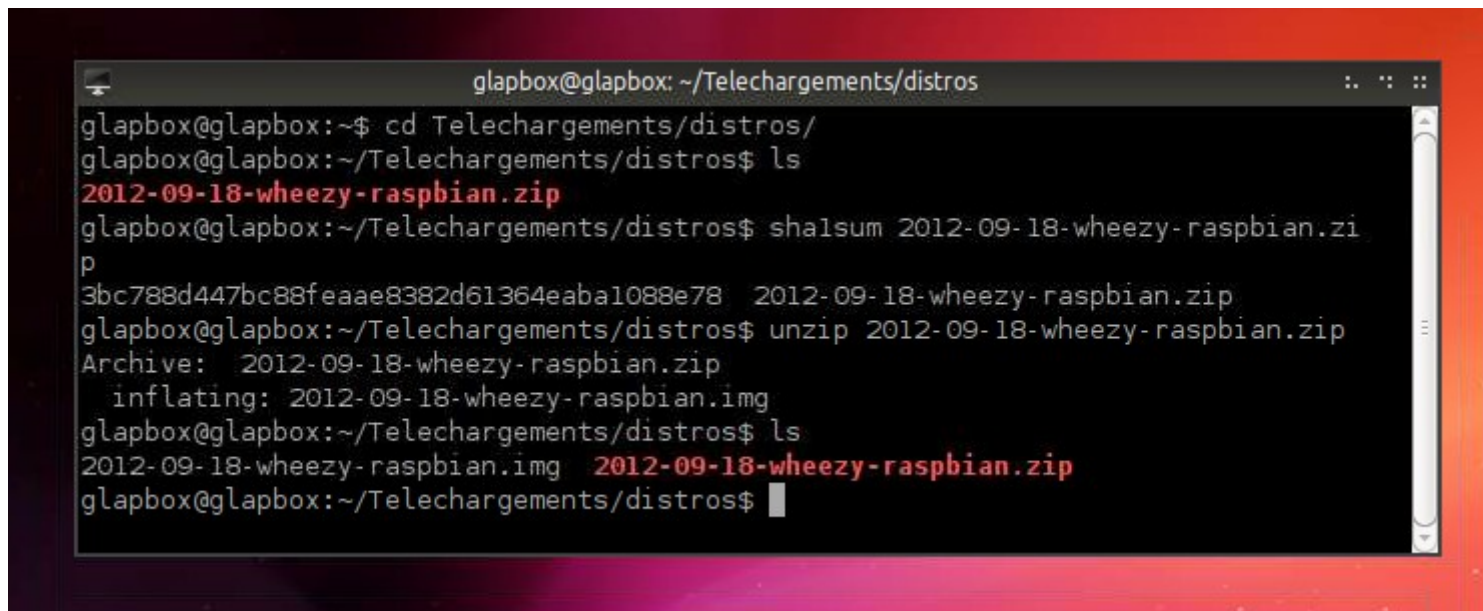
- Pour faire la même chose en ligne de commande, ouvrir à nouveau un terminal et saisir successivement les lignes de commande suivantes :

```
$ cd  
$ cd chemin/du/fichier/a/extraire  
$ unzip 2012-09-18-wheezy-raspbian.zip
```

- Attention, ça prend un certain temps car ça donne un fichier de 1,8 Go. On obtient donc un fichier \*.img. Pour vérifier la bonne obtention du fichier décompressé, saisir la commande suivante :

```
$ ls
```

- Voilà, vous êtes prêt pour la suite.



```
glapbox@glapbox: ~/Telechargements/distros  
glapbox@glapbox:~$ cd Telechargements/distros/  
glapbox@glapbox:~/Telechargements/distros$ ls  
2012-09-18-wheezy-raspbian.zip  
glapbox@glapbox:~/Telechargements/distros$ shasum 2012-09-18-wheezy-raspbian.zip  
3bc788d447bc88feaae8382d61364eaba1088e78 2012-09-18-wheezy-raspbian.zip  
glapbox@glapbox:~/Telechargements/distros$ unzip 2012-09-18-wheezy-raspbian.zip  
Archive: 2012-09-18-wheezy-raspbian.zip  
  inflating: 2012-09-18-wheezy-raspbian.img  
glapbox@glapbox:~/Telechargements/distros$ ls  
2012-09-18-wheezy-raspbian.img 2012-09-18-wheezy-raspbian.zip  
glapbox@glapbox:~/Telechargements/distros$
```

## Graver l'image sur la carte mémoire SDHC

### Information préalable

- Classiquement, les habitués d'Ubuntu et des systèmes Gnu/Linux en général le savent, pour graver une image, on va utiliser un utilitaire spécifique qui permet de créer un disque de démarrage. On peut notamment citer le logiciel Unetbootin qui permet de faire à peu près ce que l'on veut dans ce domaine. Mais ici, on va utiliser une carte mémoire SDHC qui est une mémoire FLASH et cela impose l'utilisation d'un utilitaire en ligne de commande un peu particulier, appelé **dd**.

#### Info

« L'utilitaire **dd** est une commande unix (et donc Gnu/Linux...) permettant de copier un fichier en effectuant des conversions, en ne sélectionnant qu'une partie des données à copier. Il est particulièrement adapté à la copie sur des périphériques blocs tel que des disques durs ou des lecteurs CD-ROM » et donc également des mémoires SD. Pour plus d'info voir : [http://fr.wikipedia.org/wiki/Dd\\_%28Unix%29](http://fr.wikipedia.org/wiki/Dd_%28Unix%29)

- La syntaxe de la commande **dd** est différente des autres commandes unix traditionnelles. La commande **dd** utilise des options de la forme **option=valeur** au lieu de la forme classique **-option valeur**. Les options de **dd** que nous allons utiliser ici sont les suivantes :
  - if=fichier\_entree** (Input File) : lit ce fichier en entrée. Cela peut être un fichier régulier comme un périphérique de type bloc. Par défaut, c'est l'entrée standard qui est utilisée (par exemple le clavier).
  - of=fichier\_sortie** (Output File) : écrit dans ce fichier en sortie.
- ATTENTION : L'utilitaire dd est susceptible d'effacer n'importe quelle partition ou disque de votre machine. Si vous n'utilisez pas le bon identifiant de partition dans les instructions qui suivent, vous pouvez très bien effacer par erreur la partition principale de votre système et perdre des données le cas échéant. Faites donc attention ! Pour éviter les « mauvais gags », une solution peut consister à utiliser un PC sans données sensibles dessus (netbook...).**

### Obtenir l'identifiant correspondant à la carte SDHC utilisée pour graver l'image

- Sur un système Gnu/Linux, les disques sont dénommés par un groupe de lettre de la forme sda, sdb, sdc, etc... Et sur chaque disque, les espaces utilisables, appelés également « partition » sont identifiés par un numéro suivant le nom du périphérique : par exemple sda1, sda2, etc... pour les partitions n°1 et n°2 d'un disque nommé sda.
- La première chose à faire dans notre cas est de connaître l'identifiant de notre carte SD. Ne connecter pas votre carte pour le moment. Ouvrez un terminal et saisissez la commande suivante qui permet de lister l'ensemble des postes présents sur votre système actuel.

```
$ sudo df -h
```

- Vous devez alors obtenir la liste de toutes les partitions présents sur le système. A présent, connecter la carte SD dans le lecteur SD de votre ordinateur. Puis saisissez à nouveau la commande précédente. Vous devez alors obtenir à nouveau la liste des partitions avec une nouvelle ligne qui correspond à la carte SD. Si par hasard la carte SD ne se monte pas automatiquement, ouvrez le gestionnaire de fichier et ré-exécutez la commande. Par exemple, on obtient :

```
/dev/sdc1 7,4G 32K 7,4G 1% /media/3932-3566
```

- Noter bien l'identifiant obtenu correspondant à votre carte (pour éviter de vous tromper, vérifier la taille du périphérique) !** Dans cet exemple, la carte SD a pour identifiant **sd**c et dispose d'une partition **sd**c1 de 7,4Go.

## Démonter la partition correspondant à votre carte SD

- A présent, pour pouvoir écrire des données et donc notre fichier image sur le système, il est nécessaire de démonter la partition présente sur la carte SD. Dans notre exemple, la partition **sdc1**. On utilisera donc la commande :

```
$ umount /dev/sdc1
```

- Si par hasard lors de l'exécution de la commande utilisée pour lister les partitions, plusieurs partitions étaient présentes sur la carte SD (sdc1, sdc2, sdc3...), il est nécessaire dans ce cas de les démonter une à une à l'aide de la même instruction umount suivie de l'identifiant correspondant.

## Graver l'image sur la carte mémoire SD

- A présent nous allons graver l'image du système sur la carte SDHC... Prêt ? C'est parti !

### ATTENTION (bis)

**L'utilitaire dd est susceptible d'effacer n'importe quelle partition ou disque de votre machine. Si vous n'utilisez pas le bon identifiant de partition dans les instructions qui suivent, vous pouvez très bien effacer par erreur la partition principale de votre système et perdre des données le cas échéant. Faites donc attention !** Pour éviter les « mauvais gags », une solution peut consister à utiliser un PC sans données sensibles dessus (netbook...).

**D'autre part, toutes les données présentes sur la carte mémoire SD seront également effacées !**

- Ouvrir un terminal. On commence par se placer dans le répertoire où se trouve l'image \*.img (si ce n'est déjà fait) :

```
$ cd
```

```
$ cd chemin/ou/se/trouve/le/fichier/image
```

- Ensuite, on va lancer la commande dd avec comme paramètres :
  - le fichier \*.img comme source
  - la carte SD comme destination
  - le paramètre bs qui correspond à la taille de bloc à utiliser (ici 4M. Si ça ne fonctionne pas, utiliser 1M)
- Bien lancer la commande en sudo (droits administrateurs) et en utilisant l'identifiant de la carte SD (sans les chiffres de la partition), ce qui donne :

```
$ sudo dd bs=4M if=2012-09-18-wheezy-raspbian.img of=/dev/sdc
```

- ça dure facile 5 minutes... On n'a l'impression qu'il ne se passe rien... mais ça bosse. Patienter, vous touchez presque au but !

## Finaliser la gravure de l'image sur la carte SD

- Une fois que vous avez à nouveau l'invite de la ligne de commande, la gravure est terminée. A ce stade, il ne vous reste plus qu'à exécuter en mode sudo la commande suivante pour finaliser la gravure de l'image sur la carte SD :

```
$ sudo sync
```

- Voilà, c'est fini pour cette étape ! **Bravo, vous allez enfin pouvoir lancer votre RaspberryPi !!**

## Liens utiles

- Pour plus d'info, la procédure complète en anglais et en ligne de commande sous Linux est décrite ici : [http://elinux.org/RPi\\_Easy\\_SD\\_Card\\_Setup](http://elinux.org/RPi_Easy_SD_Card_Setup)

## Connecter le Raspberry Pi aux différents périphériques pour le lancement initial du système

- Une fois que l'image système est gravée sur la carte mémoire SD, il ne reste plus qu'à connecter tous les périphériques sur le RaspberryPi avant de lancer le système :
  - connecter le réseau (câble ethernet dans un premier temps, le wifi par clé USB sera configuré ensuite)



- connecter le clavier USB et la souris USB

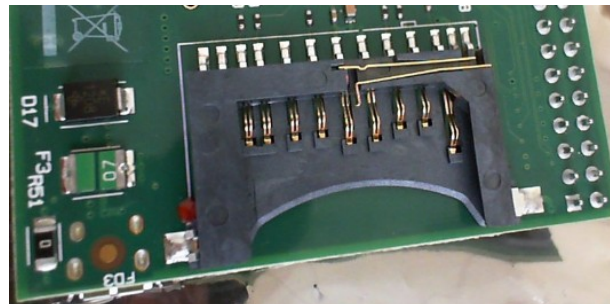


- connecter l'écran par HDMI ou RCA



ou

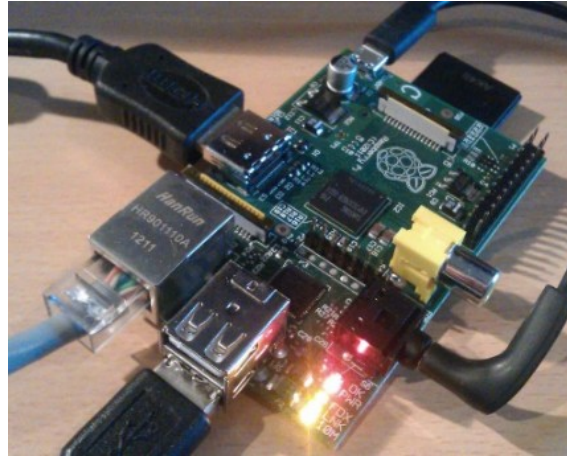
- mettre en place la carte SD dans le connecteur



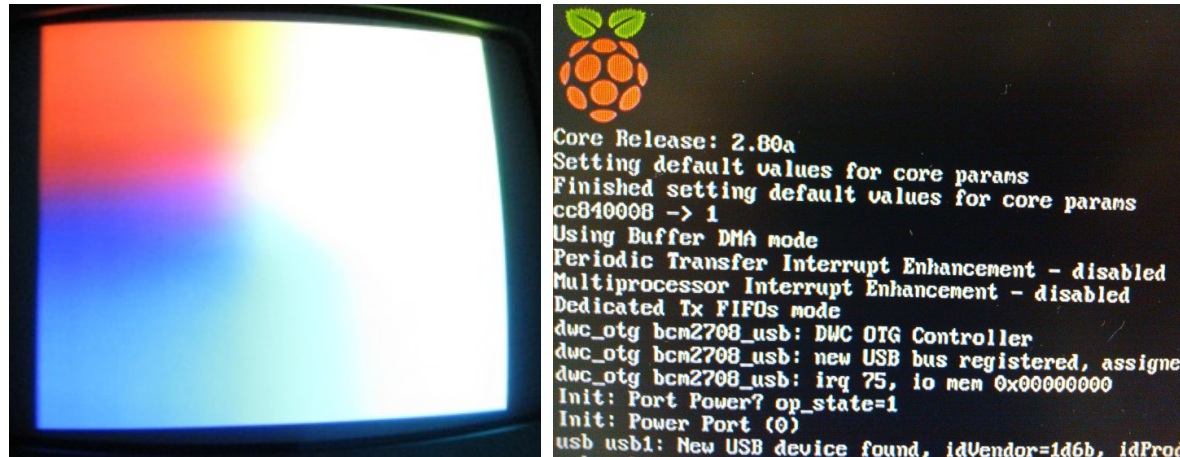
## Lancement du RaspberryPi avec le système gravé sur la carte SD

### Mise sous tension du RaspberryPi

- Une fois que tout est connecté, vous êtes prêt. Vérifier que tous les éléments de votre réseau sont sous tension, que votre écran est sous tension.
- A présent, on va mettre le RaspberryPi sous tension... Minute de vérité... et le début d'une aventure sympa... Tatata... Connecter le câble USB d'alimentation micro-USB au RaspberryPi. Le câble d'alimentation micro-USB peut-être connecté sur un port USB du PC dans un premier temps, ou bien sur un adaptateur secteur 5V/500mA disposant d'un connecteur micro-USB.



Première mise sous tension... Ici, enceintes connectées. Tatata...



Premier affichage : un écran coloré suivi d'une série de messages texte correspondant au lancement du système. C'est parti !

## Au premier boot : l'écran de configuration de l'utilitaire Raspi-config

- Une fois que le système se lance et on arrive, au premier boot, sur un écran de menu Raspi-config. Il s'agit d'un utilitaire intégré à la distribution Raspbian et qui permet de configurer un certain nombre de paramètres et d'option. Ces paramètres une fois fixés seront utilisés pour les lancements ultérieurs. Cet utilitaire pourra être lancé à tout moment ultérieurement.

```
Raspi-config

info          Information about this tool
expand_rootfs Expand root partition to fill SD card
overscan      Change overscan
configure_keyboard Set keyboard layout
change_pass   Change password for 'pi' user
change_locale Set locale
change_timezone Set timezone
memory_split  Change memory split
overclock     Configure overclocking
ssh           Enable or disable ssh server
boot_behaviour Start desktop on boot?
update        Try to upgrade raspi-config

                <Select>                <Finish>
```

- Techniquement, cette étape et cet utilitaire servent à paramétrer un fichier de configuration (le fichier /boot/config.txt). Tous les changements faits à ce stade peuvent être modifiés ultérieurement.
- Si vous êtes pressé, ce que je comprends tout à fait, voici les choix par défaut que je conseille (se déplacer avec les flèches HAUT/BAS) :
  - activer l'option « expand\_rootfs » : cette option permet d'utiliser toute la capacité de la carte SD pour le système installé.
  - activer l'option « boot\_behaviour » : de cette façon le système démarrera en mode graphique immédiatement
  - ... et c'est tout. Allez sur finish avec la touche TAB puis valider. Le système va redémarrer... en mode graphique cette fois : Yes !

### Info

On pourra relancer **raspi-config** à tout moment quand on le souhaitera en ligne de commande ultérieurement.



## Les options disponibles de l'utilitaire raspi-config

- Vous pouvez passer cette étape pour y revenir ultérieurement. Voyons plus en détails les principales options disponibles et les choix possibles. Le détail des options est décrit ici : [http://elinux.org/RPi\\_raspi-config](http://elinux.org/RPi_raspi-config)

<b>info</b>	Cette option donne des informations générales sur l'utilitaire.
<b>expand_rootfs</b>	Etend la partition du système à la totalité de la carte SD : <b>à activer</b> pour disposer du maximum de place.
<b>overscan</b>	Permet d'étendre l'affichage sur une télévision.
<b>configure_keyboard</b>	Permet de configurer le clavier utiliser : <b>configurer pour l'utilisation d'un clavier français</b>
<b>change_pass</b>	Permet de modifier le mot de passe de l'utilisateur par défaut pi. Le mot de passe par défaut est raspberry. <b>C'est plus prudent de le modifier</b> , mais n'oubliez pas votre mot de passe.
<b>change_locale</b>	Permet de configurer le jeu de caractère utilisé sur le système. <b>Activer les locales FR-fr</b>
<b>change_timezone</b>	Permet de modifier le fuseau horaire : <b>Choisir le fuseau Europe Paris si vous êtes en France</b> . A adapter à votre situation
<b>memory_split</b>	Permet de fixer la quantité de RAM allouée pour les opérations utilisant la vidéo. Ne pas modifier par défaut.
<b>overclock</b>	L'overclocking permet d'augmenter la cadence maximale de l'horloge du processeur pour une plus grande rapidité d'exécution. En contrepartie, cela entraîne une consommation plus grande et une élévation de la température. Ne pas utiliser sans radiateur ?
<b>ssh</b>	Le serveur SSH permet de se connecter à distance de façon sécurisée à une système. Activer si besoin. Utilisateurs avertis.
<b>boot_behaviour</b>	Permet d'activer le démarrage en mode graphique avec auto-connexion automatique de l'utilisateur pi. <b>A activer de préférence</b> . Si cette option n'est pas activée, vous obtiendrez une invite en mode texte.
<b>update</b>	Permet la mise à jour de l'utilitaire. Ne pas activer par défaut.

- Une fois les options activées, aller sur finish avec la touche TAB et valider.

### En synthèse

Les deux options qu'il est très fortement conseillé d'activer : boot\_behaviour et expand\_rootfs  
Les options à activer à votre convenance sont configure\_keyboard, change\_pass, change\_locale, change\_timezone.

## Au second boot...

### En ligne de commande

- Si on n'a pas activé l'option **boot\_behaviour**, on se retrouve avec une invite texte pour se logger. Utiliser le login par défaut **pi** et le mot de passe par défaut **raspberry** (à moins que vous l'ayez modifié).
- A ce moment là, vous devez obtenir un prompt de ligne de commande qui vous permet de faire ce que vous voulez (utilisateurs avancés)

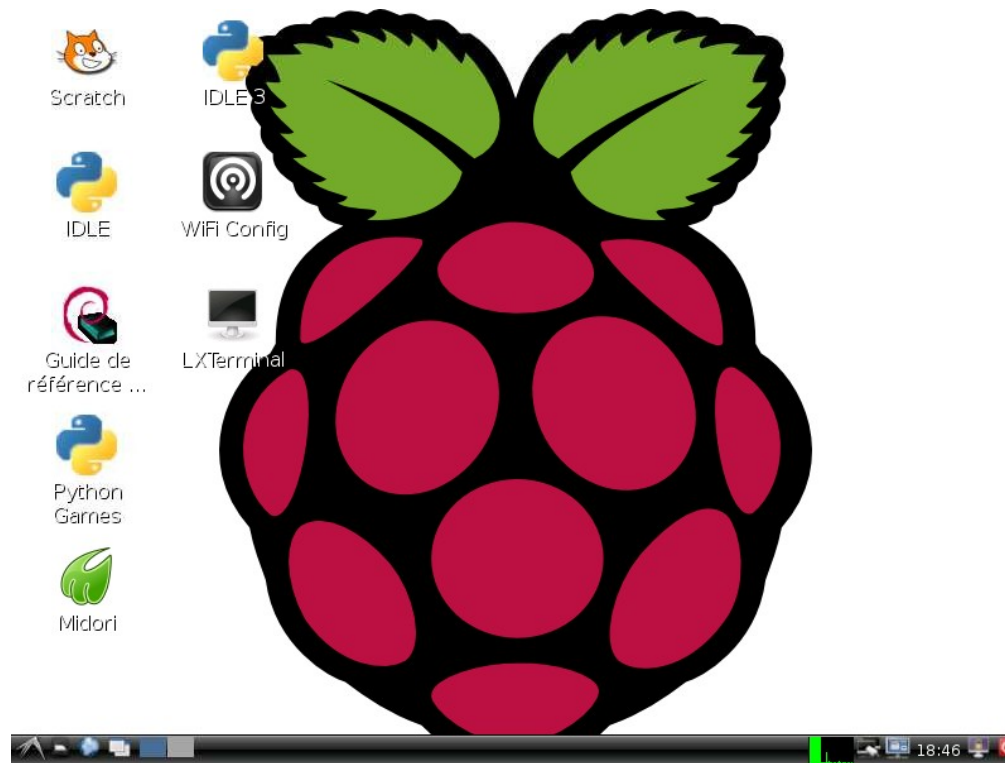
```
$
```

- A ce stade, vous pouvez également lancer le système graphique avec la commande :

```
$ startx
```

### Mode graphique

- Si on a bien activé l'option **boot\_behaviour**, on doit obtenir le lancement immédiat de l'interface graphique. Vous devez vous retrouver face à l'écran d'une session graphique LXDE classique.



**Vous obtenez bien cet écran ? Bravo, vous avez réussi votre installation et votre RaspberryPi est opérationnel !**

## Et après ?

- A ce stade, vous disposez d'un RaspberryPi opérationnel, plusieurs possibilités s'offrent maintenant à vous.
  - **Si vous ne connaissez pas du tout Gnu/Linux**, je vous conseille fortement, avant de passer à autre chose, de vous familiariser avec le système installé et notamment :
    - découvrir les rudiments de l'utilisation de la ligne de commande
    - découvrir le menu de l'environnement LXDE et les logiciels installés.
  - **Si vous estimez être à l'aise avec un environnement Gnu/Linux** et si la ligne de commande ne vous effraie pas, vous allez pouvoir personnaliser votre système :
    - pour configurer les paramètres du système « aux petits oignons » : langue française, heure, etc.. à l'aide de l'utilitaire raspi-config
    - pour le rendre « embarqué » en activant l'accès au bureau distant à partir d'un poste fixe Windows, Mac Gnu/Linux ou Android
    - pour activer le wifi et rendre le système accessible sans fil
    - pour tester et installer des fonctionnalités utiles, la vidéo, le son, la synthèse vocale, etc...
    - pour installer le logiciel Arduino pour programmer, contrôler à distance des dispositifs
    - pour installer la chaîne de développement d'interfaces graphiques en Python + Qt,
    - etc..
  - Ensuite, vous pourrez explorer et utiliser votre RaspberryPi dans toutes sortes d'utilisations et d'applications : domotique, robotique, etc... ce ne sont pas les idées qui manquent !

### **Note technique :**

- Pour adapter la résolution de l'écran fournit par le RaspberryPi, il faut éditer le fichier /boot/config.txt :

```
$ sudo apt-get install geany  
$ sudo geany /boot/config.txt
```

- Puis décommenter et modifier les lignes :

```
framebuffer_width=800  
framebuffer_height=600
```