

Migration de Lilo vers GNU Grub

par Arnaud Fontaine Patrice Vetsel Glenux Ptitlouis

Juillet 2003

Table of Contents

- [1 Qu'est ce qu'un chargeur d'amorçage ?](#)
- [2 Pourquoi passer à GNU Grub ?](#)
- [3 Installation de GNU Grub](#)
- [4 lilo.conf vs menu.lst](#)
- [5 Personnaliser GNU Grub](#)
- [6 Configuration automatique de GNU Grub lors de l'installation d'un nouveau noyau](#)
- [7 Liens](#)
- [8 Annexe : traduction du FAQ de GNU/Grub](#)
 - ◆ [8.1 J'ai des questions au sujet du GRUB !](#)
 - ◆ [8.2 En quoi GNU GRUB diffère-t-il du GRUB original d'Erich ?](#)
 - ◆ [8.3 GRUB peut-il démarrer mon système d'exploitation à partir d'un disque dur de plus de 8Go ?](#)
 - ◆ [8.4 Puis-je placer le Stage2 sur une partition au-delà du 1024ème cylindre ?](#)
 - ◆ [8.5 Comment créer une disquette de démarrage GRUB avec un menu ?](#)
 - ◆ [8.6 Comment spécifier une partition ? Cela semble très différent de mon système d'exploitation...](#)
 - ◆ [8.7 GRUB ne reconnaît pas ma partition GNU/Hurd.](#)
 - ◆ [8.8 J'ai installé une version récente de binutils, mais GRUB plante encore lors du lancement.](#)
 - ◆ [8.9 Je ne peux pas compiler GRUB sur Red Hat 7.0.](#)
 - ◆ [8.10 Comment puis-je spécifier une taille de mémoire arbitraire à Linux ?](#)
 - ◆ [8.11 Comment démarrer Windows \(ou DOS\) depuis un disque secondaire ?](#)
 - ◆ [8.12 J'ai une partition de démarrage GNU/Linux \(ou un autre UNIX-like\), et GRUB semble ne pas gérer cette situation correctement.](#)
 - ◆ [8.13 Comment désinstaller GRUB de mon disque dur ?](#)
 - ◆ [8.14 GRUB s'arrête quand j'accède à mon disque IDE de grande capacité.](#)
 - ◆ [8.15 GRUB s'arrête quand j'accède à mon disque SCSI.](#)
 - ◆ [8.16 J'ai installé GRUB, mais il vient juste de s'arrêter.](#)
 - ◆ [8.17 Pourquoi Linux \(FreeBSD, NetBSD, etc.\) ne devient-il pas compatible Multiboot ?](#)

1 Qu'est ce qu'un chargeur d'amorçage ?

Un chargeur d'amorçage (*bootloader*) est un petit programme de 512 octets qui s'exécute avant le lancement de votre système d'exploitation et juste après le BIOS. Pourquoi 512 octets ? Simplement parce que votre disque dur est divisé en secteur de 2⁹ octets soit 512 octets. Ce programme peut être stocké sur une mémoire permanente externe, à la différence de la mémoire centrale dont le contenu est réinitialisé après chaque arrêt de l'ordinateur. La mémoire permanente externe prend la forme de supports très différents tels que cédéroms, disquettes, disques durs. Ce programme réalise une tâche très simple et assez brève mais néanmoins primordiale au fonctionnement de votre ordinateur : il lit et charge en mémoire l'image du noyau d'un système d'exploitation quelconque avant que ce dernier ne prenne la main.

Dans cet article le démarrage du système d'exploitation se fera à partir d'un disque dur. Ainsi, le BIOS est chargé de trouver un système d'exploitation en lisant tout d'abord le MBR (*Master Boot Record*) qui contient la table des partitions du disque dur. Une fois cette tâche effectuée, deux alternatives permettent le démarrage de votre système d'exploitation : présence d'un chargeur d'amorçage dans le MBR ou présence d'une partition marquée comme active contenant un bloc d'amorçage placé au tout début de celle-ci. L'une de ces deux méthodes va permettre de charger un « *petit programme* » comme le stage2 de GNU Grub.

2 Pourquoi passer à GNU Grub ?

Ainsi l'installation de Debian GNU/Linux procède, si vous le désirez, à l'installation d'un chargeur de démarrage appelé LILO (*Linux LOader*) qui permet de choisir, parmi les éventuels systèmes d'exploitation présents sur votre machine, lequel vous souhaitez démarrer et plus spécifiquement pour Linux, quel noyau vous souhaitez démarrer. GNU Grub (*G*Rand *U*nified *B*ootloader) est un autre chargeur de démarrage, dont l'installation n'est pas proposé lors de l'installation de la distribution et qui est beaucoup plus puissant que LILO pour les raisons suivantes :

- il fournit une interface en ligne de commande (*shell*) qui vous permet d'éditer les paramètres d'amorçage de votre système d'exploitation sans avoir à modifier le fichier de configuration et grâce à laquelle on peut utiliser les diverses fonctions de GNU Grub. Cela permet notamment de démarrer le système même si on s'est trompé dans le fichier de configuration que nous allons aborder un peu plus tard.
- « *Il reconnaît les systèmes de fichiers et les formats des noyaux, vous pouvez donc charger un système d'exploitation arbitraire comme vous voulez, sans enregistrer la position physique du noyau sur le disque* » (Manuel de GNU Grub).
- Il est possible de démarrer depuis le réseau en utilisant le protocole TFTP.
- Le respect de la Spécification Multiboot qui permet de charger plusieurs modules, très utile pour le démarrage du micro-noyau GNU/Hurd mais aussi pour FreeBSD.
- Il supporte de nombreux systèmes de fichiers et permet donc de lire des données sur vos partitions.

Cet article est donc destiné à vous aider à migrer de Lilo vers GNU Grub simplement et facilement juste après une installation standard de Debian GNU/Linux ou dérivées. Pour plus d'informations sur les caractéristiques complètes de GNU Grub consultez cette page :

<http://www.linux-france.org/article/sys/chargeurs/ix86/grub/grub-manual-fr.html#Caracteristiques>

3 Installation de GNU Grub

L'installation de GNU Grub et éventuellement de la documentation de GNU Grub se fait très simplement grâce à apt-get :

```
apt-get install grub grub-doc
```

Maintenant que nous avons installé le paquet, nous allons installer GNU Grub sur le disque dur. Si vous choisissez de l'installer sur une partition et non sur le MBR, il faudra que cette partition soit primaire **et** active sinon votre chargeur d'amorçage ne sera pas lancé au démarrage de votre ordinateur.

Une petit aparté au sujet des conventions de nommage des disques durs sous GNU Grub. Celui-ci ne nomme pas les partitions de la même façon que Linux. Prenons le partitionnement de mes deux disques durs afin de mieux comprendre comment cela fonctionne :

Nommage sous Linux Nommage sous GNU Grub

/dev/hda	(hd0)
/dev/hda1	(hd0,0)
/dev/hda5	(hd0,4)
/dev/hda6	(hd0,5)
/dev/hdb	(hd1)
/dev/hdb2	(hd1,1)
/dev/hdb3	(hd1,2)
/dev/hdb5	(hd1,4)

On remarque que la numérotation des disques et partitions commencent à 0 et que le nom du périphérique sont entre parenthèses.

Nous allons donc installer GNU Grub sur hda1 qui constitue ma partition */boot* marquée comme active :

```
# grub-install '(hd0,0)'
```

De plus l'exemple s'adapte facilement si vous voulez installer grub dans le MBR :

```
# grub-install '(hd0)'
```

Ces deux façons d'installer le grub vont permettre d'installer GNU Grub soit dans le secteur d'amorçage de /dev/hda1 dans le premier cas, soit de l'installer dans le MBR dans le second cas. Cela va créer sur votre système actuelle un répertoire */boot/grub/* contenant le stage2 de 512 octets ainsi que d'autres stages qui vont permettre de reconnaître différents systèmes de fichier :

```
# ls -l /boot/grub/
device.map
e2fs_stagel_5
fat_stagel_5
ffs_stagel_5
jfs_stagel_5
minix_stagel_5
reiserfs_stagel_5
stagel
stage2
vstafs_stagel_5
xfs_stagel_5
```

4 lilo.conf vs menu.lst

Maintenant que nous avons survolé l'installation de GNU Grub sur le disque dur, nous allons maintenant le configurer en se basant sur l'exemple fournit par Debian (*/usr/share/doc/grub/examples/menu.lst*). Pour cela nous allons nous aider du fichier de configuration de Lilo généré pendant l'installation de votre distribution et qui doit ressembler à quelque chose comme ça si Lilo est installé sur /dev/hda1 (partition /boot) et que votre partition racine est /dev/hda7, par exemple :

```
#/etc/lilo.conf
# Début du fichier de configuration de Lilo
```

Migration de Lilo vers GNU Grub

```
# Emplacement du chargeur d'amorçage, ici /dev/hda1 sinon /dev/hda
# pour le MBR
boot=/dev/hda1
map=/boot/map
install=/boot/boot.b
prompt
# Temps d'attente avant de charger l'image du noyau par défaut qui porte # le label Linux
timeout=50
default=Linux
# Deux systèmes d'exploitation peuvent être amorcés
# D'abord Linux avec le support de votre graveur (/dev/hdc)
image=/boot/vmlinuz-2.2.20-idepci
    label=Linux
    read-only
    root=/dev/hda7
    append="hdc=ide-scsi" # Et enfin Windows
other=/dev/hda2
    label=Win
# Fin du fichier de configuration
```

D'après les conventions de nommage des partitions sous GNU Grub que l'on a vu ci dessus, on en déduit un fichier de configuration pour GNU Grub dont on va donner l'exemple ci dessous. Notez que nous avons réaffecté les touches du clavier pour qu'il soit en azerty car par défaut il est en qwerty au démarrage.

```
# /boot/grub/menu.lst
# Début du fichier de configuration de GNU Grub

# Temps d'attente avant de charger l'image du noyau par défaut qui sera
# la première image donnée ci dessous, si on n'arrive pas à la charger on
# essaie la suivante (fallback)
timeout 50
default 0
fallback 1

# Emulation d'un clavier azerty_FR (merci à Yannick)
setkey less backquote
setkey greater tilde
setkey ampersand 1
setkey 1 exclam
setkey tilde 2
setkey 2 at
setkey doublequote 3
setkey 3 numbersign
setkey quote 4
setkey 4 dollar
setkey parenleft 5
setkey 5 percent
setkey minus 6
setkey 6 caret
```

```

setkey backquote 7
setkey 7 ampersand
setkey underscore 8
setkey 8 asterisk
setkey backslash 9
setkey 9 parenleft
setkey at 0
setkey 0 parenright
setkey parenright minus
setkey numbersign underscore
setkey a q
setkey A Q
setkey z w
setkey Z W
setkey caret bracketleft
setkey dollar bracketright
setkey q a
setkey Q A
setkey m semicolon
setkey M colon
setkey bracketleft quote
setkey percent doublequote
setkey asterisk backslash
setkey bracketright bar
setkey w z
setkey W Z
setkey comma m
setkey question M
setkey semicolon comma
setkey period less
setkey colon period
setkey slash greater
setkey exclam slash
setkey bar question

```

```

# L'image suivante sera chargé par défaut, ici Linux, root spécifie
# l'emplacement du chargeur de démarrage, ici /dev/hda1 ou (hd0,0)
title Debian GNU/Linux, 2.2.20–idepci
# Si vous n'avez pas de partition /boot alors ce sera plutôt pour le MBR et # si votre partition racine est
/dev/hda5:
# root (hd0)
# kernel (hd0,4)/boot/vmlinuz–2.2.20–idepci root=/dev/hda7
root (hd0,0)
kernel /vmlinuz–2.2.20–idepci root=/dev/hda7 hdc=ide–scsi

# L'image suivante, ici Windows (/dev/hda2) sera démarré si la précédente
# image pose problème
title Windows 98
rootnoverify (hd0,1)
chainloader +1

```

Fin du fichier de configuration

Un peu plus sur les options ci dessus : l'option *chainloader* permet de charger un noyau non pris en charge par GNU Grub, utile pour Windows mais aussi pour OpenBSD par exemple. Ce court article ne contient que l'essentiel pour passer rapidement de Lilo à GNU Grub après une installation de Debian GNU/Linux, donc si vous souhaitez plus d'informations sur le fonctionnement de ce chargeur d'amorçage, nous vous invitons à visiter les liens dans la section du même nom.

Notes : En cas de doute sur les noms des partitions, vous pouvez utiliser le shell grub en tapant simplement :
grub puis en utilisant la complétion automatique pour compléter le nom des partitions mais aussi l'emplacement des fichiers. Cela peut donner quelque chose comme ça :

```
grub> kernel (hd0,<tab>
Possible partitions are:
Partition num: 0, Filesystem type is ext2fs, partition type 0x83
Partition num: 4, Filesystem type is ext2fs, partition type 0x83
Partition num: 5, Filesystem type unknown, partition type 0x82
Partition num: 6, Filesystem type is ext2fs, partition type 0x83
Partition num: 7, Filesystem type is ext2fs, partition type 0x83
Partition num: 8, Filesystem type is ext2fs, partition type 0x83
Partition num: 9, Filesystem type is ext2fs, partition type 0x83
grub> root (hd0,0)/<tab>
Possible files are: lost+found grub System.map config
System.map-2.4.21 config-2.4.21 vmlinuz-2.4.21
grub> kernel (hd0,0)/vmlinuz-2.4.21
grub> quit
```

5 Personnaliser GNU Grub

Tout comme Lilo, GNU Grub est personnalisable par la possibilité de mettre une image de fond au menu du chargeur d'amorçage. Néanmoins cette possibilité nécessite d'ajouter le support des *splashimage* à GNU Grub qui est disponible seulement dans le paquet de la version instable de la distribution. Vous pouvez tout de même installer cette version sur testing en téléchargeant simplement le paquet (à l'adresse suivante par exemple : http://http.us.debian.org/debian/pool/main/g/grub/grub_0.93+cv20030624-3_i386.deb) et en tapant la commande suivante car les versions des paquets dont dépend GNU Grub en testing sont les mêmes qu'en instable :

```
# dpkg -i grub_0.93+cv20030624-3_i386.deb
```

Maintenant que votre version de GNU Grub supporte les *splashimage*, nous allons convertir une image dans un format quelconque vers un format « lisible » grâce au paquet *imagemagick* que nous installons puis nous tapons la commande magique :

```
# apt-get install imagemagick
# mkdir -p /usr/share/grub/images
# convert -geometry 640x480 -colors 14 image.png
/usr/share/grub/images/image.xpm
# cd /usr/share/grub/images/ && gzip image.xpm
```

Enfin, nous modifions le fichier */boot/grub/menu.lst* :

```
# /boot/grub/menu.lst
timeout 50
default 0
fallback 1

# Emulation d'un clavier azerty_FR (merci à Yannick)
setkey less backquote
setkey greater tilde
setkey ampersand 1
setkey 1 exclam
setkey tilde 2
setkey 2 at
setkey doublequote 3
setkey 3 numbersign
setkey quote 4
setkey 4 dollar
setkey parenleft 5
setkey 5 percent
setkey minus 6
setkey 6 caret
setkey backquote 7
setkey 7 ampersand
setkey underscore 8
setkey 8 asterisk
setkey backslash 9
setkey 9 parenleft
setkey at 0
setkey 0 parenright
setkey parenright minus
setkey numbersign underscore
setkey a q
setkey A Q
setkey z w
setkey Z W
setkey caret bracketleft
setkey dollar bracketright
setkey q a
setkey Q A
setkey m semicolon
setkey M colon
setkey bracketleft quote
setkey percent doublequote
setkey asterisk backslash
setkey bracketright bar
setkey w z
setkey W Z
setkey comma m
setkey question M
setkey semicolon comma
setkey period less
```

```
setkey colon period
setkey slash greater
setkey exclam slash
setkey bar question

# Emplacement de l'image, chez moi la partition /usr est sur une partition
# séparée (/dev/hda8) d'où (hd0,7) sous GNU Grub
splashimage=(hd0,7)/share/grub/images/image.xpm.gz

title Debian GNU/Linux, 2.2.20-idepci
root (hd0,0)
kernel /vmlinuz-2.2.20-idepci root=/dev/hda7 hdc=ide-scsi

title Windows 98
rootnoverify (hd0,1)
chainloader +1
```

Vous pouvez trouver des images pré converties en xpm et très jolies en plus à cette adresse :
<http://ruslug.rutgers.edu/~mcgrof/grub-images/images/>

6 Configuration automatique de GNU Grub lors de l'installation d'un nouveau noyau

Lilo est le chargeur par défaut sous Debian, sa mise à jour est automatisée lors de l'installation d'un nouveau noyau. Et bien GNU Grub peut en faire tout autant, voire même mieux. Une fois que le paquet grub est installé sur votre système, il vous suffit de taper :

```
update-grub
```

Editez le fichier /boot/grub/menu.lst et modifiez vos lignes par défaut #groot et #kopt. N'enlevez pas les #. Cela devrait donner quelque chose comme ça quand vous éditez ce fichier :

```
## DO NOT UNCOMMENT THEM, Just edit them to your needs

## ## Start Default Options ##
## default kernel options
## default kernel options for automagic boot options
## e.g. kopt=root=/dev/hda7 ro
# kopt=root/dev/hda7 hdc=ide-scsi

## default grub root device
## e.g. groot=(hd0,0)
# groot=(hd0,0)

## should update-grub create alternative automagic boot options
## e.g. alternative=true
## alternative=false
# alternative=
```


Migration de Lilo vers GNU Grub

```
## should update-grub lock alternative automagic boot options
## e.g. lockalternative=true
## lockalternative=false
# lockalternative=

## altpoption boot targets option
## multiple altpoptions lines are allowed
## e.g. altpoptions=(extra menu suffix) extra boot options
## altpoptions=(recovery mode) single
# altpoptions=

## controls how many kernels should be put into the menu.lst
## only counts the first occurrence of a kernel, not the
## alternative kernel options
## e.g. howmany=all
## howmany=7
# howmany=

## ## End Default Options ##
```

Notez aussi que si vous disposez d'un fichier `/boot/initrd` (`# man initrd`), par exemple pour l'image du noyau se nommant `/boot/vmlinuz-2.2.20-idepci`, alors il faudra renommer `/boot/initrd` en `/boot/initrd-2.2.20-idepci` afin que la ligne `initrd /initrd-2.2.20-idepci` soit automatiquement ajoutée à l'image `kernel-2.2.20-idepci` dans ce cas. Relancez une dernière fois `update-grub`. Créez ou éditez ensuite le fichier `/etc/kernel-img.conf` et insérez les lignes suivantes :

```
# /etc/kernel-img.conf
postinst_hook=/sbin/update-grub
postrm_hook=/sbin/update-grub
```

Dorénavant lors de l'installation d'un nouveau noyau, `update-grub` sera lancé et scanner votre répertoire `/boot` en inventoriant les images des noyaux s'y trouvant, et il écrira une nouvelle entrée dans le fichier de configuration de GNU Grub. Si vous avez suivi la suggestion d'utiliser `--append-to-version=.aammjj` lors de la création de votre noyau à la sauce Debian, vos noyaux seront listés par version et par date comme décrit dans la documentation disponible sur [andesi](#) à propos de Compiler son noyau Linux à la sauce Debian. Le noyau par défaut de GNU Grub sera le premier de la liste, c'est à dire le nouveau noyau, tant que vous n'aurez pas fait un autre choix avant la limite de temps.

Lorsque vous serez en confiance avec grub, vous pourrez supprimer Lilo en tapant `dpkg -P lilo`, de la sorte vous ne serez plus questionné au sujet de Lilo, lors de l'installation d'un nouveau noyau. Pour plus d'informations concernant `update-grub`, vous pouvez consulter la page de manuel associé à cette commande (`# man update-grub`).

7 Liens

Voici quelques liens que j'ai trouvés particulièrement intéressants sur le sujet :

- « *Utiliser GRUB* », <http://www.linux-france.org/article/sys/chargeurs/ix86/grub/grub-manual-fr.html>
La documentation de GNU Grub en français, très complète et très intéressante pour ceux qui voudraient en apprendre plus sur ce chargeur d'amorçage.
- « *GRUB-HOWTO* », <http://www.cri74.org/linux/howto/grub-howto.html>
Une autre documentation en français sur GNU Grub aussi intéressante que la précédente...
- « *GRUB Splash Image Howto* », <http://ruslug.rutgers.edu/~mcgrof/grub-images/>
Un howto sur les splashimage d'où est tirée la section précédente, très intéressant si vous souhaitez personnaliser Grub.
- « *Installation et configuration de GRUB* », <http://www.linux-france.org/article/sys/chargeurs/ix86/grub/installation.html>
Une page destinée à la découverte de GNU/Grub qui vous permet de comprendre assez rapidement le fonctionnement de ce chargeur d'amorçage.

Nous tenons à remercier Yannick, CSCMEU, OcamlScheme et GnuX pour leur suggestion et leur aide dans l'écriture de cet article.

8 Annexe : traduction du FAQ de GNU/Grub

Ceci est la version traduite par GnuX et relue par Ptitlouis et Arnaud Fontaine du FAQ de GNU/Grub disponible dans sa version originale en anglais à l'adresse suivante :
<http://www.gnu.org/software/grub/grub-faq.html>.

8.1 J'ai des questions au sujet du GRUB !

Avant tout, lisez très attentivement la documentation livrée avec la distribution du GRUB. Vous devriez être en mesure d'obtenir la plupart des informations à son sujet dans ce document. Cette FAQ vous donne juste quelques pistes.

Aussi, remarquez que nous n'acceptons pas les bugs provenant de versions non-officielles, telle que celle de Red Hat. Beaucoup de versions distribuées avec GNU/Linux sont modifiées selon la distribution avec leurs propres rustines, ainsi tous les bugs ne sont pas présents dans les versions officielles. Essayez donc la dernière version officielle, avant de nous rapporter des bogues.

8.2 En quoi GNU GRUB diffère-t-il du GRUB original d'Erich ?

GNU GRUB est le successeur du superbe GRUB de Erich. Il ne pouvait plus travailler sur le GRUB à cause de quelques autres travaux en cours, ainsi le mainteneur actuel Gordon Matzigkeit pris la suite de Erich, et ouvrit le développement de façon à ce que tout un chacun puisse y participer.

Techniquement parlant, GNU GRUB a de nombreuses fonctionnalités qui n'étaient pas présent dans le GRUB original. Par exemple, GNU GRUB peut être installé sur un système d'exploitation de type Unix (comme GNU/Linux) via l'interpréteur de commande `grub /sbin/grub`, il supporte le mode Logical Block Address (LBA) qui résout le problème de la limite du 1024ème cylindre, et TAB complète un nom de fichier lorsqu'il est unique. Bien sûr, beaucoup de corrections de bugs sont faites, ainsi il est recommandé d'utiliser GNU GRUB.

8.3 GRUB peut-il démarrer mon système d'exploitation à partir d'un disque dur de plus de 8Go ?

Cela dépend de votre BIOS et de votre système d'exploitation. Vous devez vous assurer que votre disque est accessible dans le mode LBA. Généralement, c'est configurable dans votre BIOS . Lisez le manuel de votre BIOS pour plus d'informations.

De plus, quelques systèmes d'exploitation (comme DOS) ne peuvent accéder à des disques de grande capacité, ainsi le problème n'est résolu par aucune sorte de chargeur de démarrage. Pour autant que je le sache, au moins GNU/Hurd et GNU/Linux peuvent démarrer depuis des disques d'une telle capacité.

8.4 Puis-je placer le Stage2 sur une partition au-delà du 1024ème cylindre ?

Oui, vous pouvez, si votre BIOS supporte le mode LBA. Bien que vous ayez à utiliser un Stage1 spécial dans la version précédente, ce n'est plus nécessaire, parce que le Stage1 supporte maintenant les deux modes.

8.5 Comment créer une disquette de démarrage GRUB avec un menu ?

1. Créez un système de fichiers sur votre disquette(par exemple `mke2fs /dev/fd0`).
2. Montez la disquette quelque part, disons, `/mnt`.
3. Copiez les images du GRUB dans le répertoire `/mnt/boot/grub`. Seuls les fichiers `stage1`, `stage2` et `menu.lst` sont nécessaires. Vous pouvez ne pas copier les fichiers `*stage1_5`.
4. Démontez la disquette.
5. Exécutez les commandes suivantes (remarquez que l'exécutable `grub` peut résider dans un répertoire différent de votre système, par exemple, `/usr/sbin`):

```
/sbin/grub --batch --device-map=/dev/null <<EOF
device (fd0) /dev/fd0
root (fd0)
setup (fd0)
quit
EOF
```

8.6 Comment spécifier une partition ? Cela semble très différent de mon système d'exploitation...

Oui, la syntaxe des périphériques GRUB est très différente des autres, mais ressemble quelque peu à DOS et Windows 98. GRUB ne fait pas la distinction entre IDE et SCSI, ainsi vos disques durs sont représentés ainsi (`hd%6d`) (%d est un entier, décompté à partir de *zéro*). De même, vos disquettes sont représentées ainsi (`fd%d`).

Le point qui engendre beaucoup de confusion chez les utilisateurs est le style de numérotation des partitions. GRUB compte les partitions à partir de zéro, tandis que presque tous les systèmes d'exploitation comptent à partir de un. Un jour nous corrigerons cette incohérence.

8.7 GRUB ne reconnaît pas ma partition GNU/Hurd.

Je ne sais pas pourquoi, mais les auteurs du programme FDISK ont assigné le type 0x63 à la partition GNU/Hurd de façon incorrecte. Ainsi utilisez 0x83 si la partition contient un système de fichiers ext2, et utilisez 0xA5 si la partition contient un système de fichiers ffs, que la partition soit occupée par Hurd ou non. Nous utiliserons 0x63 pour le système de fichiers GNU Hurd qui n'a pas encore été implémenté.

8.8 J'ai installé une version récente de binutils, mais GRUB plante encore lors du lancement.

Veuillez vérifier la version utilisée de binutils avec cette commande :

```
$ ld -v
```

Cela vous montrera deux versions, mais seul la seconde est importante. Si la version est identique à celle que vous avez installée, l'installation n'est pas mauvaise.

Bien, essayez :

```
$ gcc -Wl,-v 2>&1 | grep "GNU ld"
```

Si ce n'est pas identique au résultat ci-dessus, vous devrez spécifier le répertoire où vous avez installé binutils pour le script configure, comme ceci :

```
$ ./configure --with-binutils=/usr/local/bin
```

Si vous suivez les instructions ci-dessus mais que GRUB plante encore, il y a probablement un sérieux bogue dans GRUB. Reportez-le s'il vous plaît grâce au système de suivi des bogues.

8.9 Je ne peux pas compiler GRUB sur Red Hat 7.0.

Avez-vous lu ce message ? Nous n'avons jamais supporté une version non-officielle de gcc.

8.10 Comment puis-je spécifier une taille de mémoire arbitraire à Linux ?

Mettez l'option mem= dans votre noyau Linux, comme ceci :

```
grub> kernel /vmlinuz mem=128M
```

Vous pouvez ajouter d'autres options de la même façon. Reportez vous à la documentation de Linux, pour les options disponibles.

8.11 Comment démarrer Windows (ou DOS) depuis un disque secondaire ?

Migration de Lilo vers GNU Grub

Utilisez la commande `map`, pour échanger *virtuellement* les disques du BIOS, comme ceci :

```
grub> map (hd0) (hd1)
grub> map (hd1) (hd0)
```

8.12 J'ai une partition de démarrage GNU/Linux (ou un autre UNIX-like), et GRUB semble ne pas gérer cette situation correctement.

C'est souvent reporter comme étant un *bogue*, mais ce n'est pas vraiment un bogue. C'est une *fonctionnalité* de GRUB.

Parce que GRUB est un gestionnaire de démarrage et il ne tourne normalement sous aucun système d'exploitation particulier, il ne sait pas où une partition est montée dans votre système d'exploitation. Ainsi, si vous avez une partition `/boot` et que vous avez installé les images de GRUB dans le répertoire `/boot/grub`, GRUB reconnaît les images présentes dans le répertoire `/grub` mais pas `/boot/grub`. Et c'est bien ainsi, puisqu'il n'y a aucune garantie que tous les systèmes d'exploitation montent la même partition sous `/boot`.

Il y a plusieurs solutions à cette situation :

1. Installez GRUB dans le répertoire `/boot/boot/grub` au lieu de `/boot/grub`. Cela peut sonner faux mais ça marche bien.
2. Créez un lien symbolique avant d'installer GRUB, comme `cd /boot; ln -s . boot`. Cela marche seulement si le système de fichiers présent sur la partition de boot supporte les liens symboliques et GRUB supporte aussi cette fonctionnalité.
3. Installez GRUB avec la commande `install`, pour spécifier les chemins vers les images de GRUB de façon absolue. voici un exemple :

```
grub> root (hd0,1)
grub> install /grub/stage1 d (hd0) /grub/stage2 p /grub/menu.lst
```

8.13 Comment désinstaller GRUB de mon disque dur ?

Il n'y a pas de concept de *désinstallation* pour les gestionnaires de démarrage, parce que si vous *désinstallez* un gestionnaire de démarrage, il ne vous restera qu'une machine ne pouvant démarrer. Ainsi, vous devez simplement restaurer le gestionnaire de démarrage que vous voulez avoir sur votre disque, c'est-à-dire, installer le gestionnaire de démarrage sans désinstaller GRUB.

Par exemple, si vous voulez installer le gestionnaire de démarrage de Windows, faites simplement `FDISK /MBR` sur Windows. Si vous voulez installer LILO (Bien que je ne puisse pas imaginer pourquoi vous voudriez faire une telle chose), faites `/sbin/lilo` sur GNU/Linux.

8.14 GRUB s'arrête quand j'accède à mon disque IDE de grande capacité.

Si votre disque a une capacité supérieure à 32 Go, mettre à jour votre BIOS résoudra probablement votre problème. Ce bogue est bien connu et beaucoup de vendeurs devraient proposer des versions corrigées. Par exemple, si vous avez une ASUS-P3BF, mettez à jour le BIOS à V1007beta1 ou plus récent peut corriger cela. S'il vous plaît, demandez à votre vendeur, pour plus d'information.

8.15 GRUB s'arrête quand j'accède à mon disque SCSI.

Vérifiez si vous avez activé le support de l'extension INT 13 (LBA). Si c'est ainsi, désactivez le support et voyez si GRUB peut maintenant accéder à votre disque SCSI. Cela permettra de savoir si votre SCSI BIOS fonctionne ou non.

Pour le moment, nous savons que le matériel suivant ne supporte pas le mode LBA :

* Adaptec AIC-7880

Dans le cas où malheureusement vous avez un tel contrôleur SCSI, vous ne pouvez pas utiliser le mode LBA, bien que GRUB fonctionne encore bien dans le mode CHS (le si connu problème du 1024ème cylindre vient de nouveau à votre rencontre).

8.16 J'ai installé GRUB, mais il vient juste de s'arrêter.

Il y a plusieurs possibilités, mais c'est probablement qu'il y a quelque chose d'incorrect dans votre fichier catalogue des périphériques. Jetez un oeil au fichier `/boot/grub/device.map`. Le format est bien documenté dans le [manuel](#).

S'il y a quelque chose qui ne va pas, modifiez le fichier de façon appropriée et relancez `grub-install`. Si vous avez encore un problème ou que vous ne savez pas comment le modifier, ce sera plus facile d'installer GRUB avec une disquette d'amorçage plutôt que de s'accrocher à `grub-install`.

8.17 Pourquoi Linux (FreeBSD, NetBSD, etc.) ne devient-il pas compatible Multiboot ?

Veuillez demander aux mainteneurs. Si tous les noyaux libres étaient compatible Multiboot, le monde serait une utopie...

Modifications du document :

– Mises à jour effectuées le 08 Novembre 2003 : relecture effectuée par [Iglinux](#).

This document was translated from $L^A T_E X$ by [H^E-V^E-A](#).