

Installation Gentoo Linux

Y. Morère

Décembre 2005

Résumé

Le but de cet article est de présenter l'installation d'une distribution Gentoo sur une architecture AMD64, en utilisant un système full 64 bits afin d'utiliser au maximum notre matériel fraîchement installé. Cet article est principalement basé sur la documentation Gentoo disponible à l'adresse suivante : <http://www.gentoo.org/doc/fr/list.xml>. Son principale intérêt est qu'il regroupe en une seule page l'installation complète d'une station de travail basée sur un athlon64 sur carte mère Nforce4 et une carte graphique Nvidia.

Table des matières

1	Introduction	2
2	Démarrage	3
3	Préparation du disque	4
4	Installation des fichiers d'installation Gentoo	6
5	Installation d'un système de base Gentoo	9
6	Configurer et compiler le noyau	11
7	Configurer le système	12
8	Installation des outils système	14
9	Configurer le chargeur de démarrage	15
10	Finaliser l'installation	17
11	Utiliser la commande emerge	17
12	Mise à jour de /etc	20
13	La commande rc-update	20
14	Mettre à jour votre noyau	21

15 Installation d'une station de travail	26
15.1 Installation de Xorg	26
15.2 Installation des pilotes Nvidia	30
15.2.1 Installation des pilotes Gentoo	30
15.2.2 Configuration avancée	32
15.2.3 Installation des Pilotes Nvidia	33
15.2.4 Configuration Bi-ecran	34
15.3 Installation du Son via Alsa	43
15.4 Installation d'Alsa	44
15.4.1 Configurer Alsa	45
15.4.2 Résolution de problèmes	47
15.5 Installation de Gnome	49
15.5.1 Introduction	49
15.5.2 Installer GNOME	49
15.5.3 Connexion graphique GNOME	50
15.6 Mise à l'heure automatique	50
15.7 Petits trucs en vrac	52
15.7.1 Flash et Firefox	52
15.7.2 Imprimante HP en réseau et Cups	52
15.7.3 Utilisateur lambda et groupes	52
15.7.4 Tout ce qui est lancé au démarrage	53
15.7.5 Mise à jour vers php5	54
16 Installation rapide	55
17 Conclusion	78

1 Introduction

La distribution linux choisie est une Gentoo. En effet cette dernière possède un bon support de l'architecture AMD64. De plus elle a la particularité de reconstruire (compiler) la quasi totalité des logiciels que l'on installe. On est ainsi sûr que l'application est très bien adaptée et optimisée à la machine dont on dispose.

Gentoo Linux est une variante de Linux qui peut être optimisée automatiquement et paramétrée pour répondre à tout usage ou besoin spécifique. Ses possibilités d'adaptation, ses performances extrêmes et sa grande communauté d'utilisateurs et de développeurs sont les principales caractéristiques de Gentoo.

L'outil Portage peut faire de Gentoo Linux un serveur sécurisé idéal, une station de développement ou de bureautique professionnelle, une console de jeux, une solution embarquée ou autre. Vous décidez de ce que vous voulez en faire. À cause de ses multiples possibilités d'adaptation, nous appelons Gentoo Linux une métadistribution.

Le site originel de la distribution est <http://www.gentoo.org/> La documentation en français se trouve à l'adresse suivante <http://www.gentoo.org/doc/fr/list.xml> Ensuite on peut télécharger une image ISO de la distribution AMD64 à l'adresse suivante <http://www.gentoo.org/>

main/en/where.xml où plus directement à l'aide du lien suivante <http://bouncer.gentoo.org/?product=gentoo-2005.1-install-universal&os=amd64> Il s'agit du CD Gentoo 2005.1 Universal

Liens utiles :

- ▷ <http://www.gentoo.org/>
- ▷ <http://www.gentoo.fr/>
- ▷ <http://fr.gentoo-wiki.com/Accueil>
- ▷ <http://forums.gentoo.org/>
- ▷ http://gentoo-wiki.com/Main_Page

2 Démarrage

Une fois que vous avez gravé votre CD d'installation, vous devez amorcer votre système avec celui-ci. Enlevez tous les CD de vos lecteurs, redémarrez votre système et entrez dans la configuration du BIOS. Cela se fait habituellement en tapant la touche Suppr, F1 ou Échap, selon le BIOS. Dans la configuration du BIOS, changez l'ordre de démarrage afin que le CD-ROM soit testé avant le disque dur. Vous pouvez habituellement accéder à ce paramètre dans la section CMOS Setup. Si le CD-ROM n'est pas prioritaire pour l'amorçage, le système va simplement démarrer à partir du disque dur et ignorer le CD-ROM.

Maintenant, insérez le CD d'installation dans le lecteur de CD-ROM (vous l'aviez deviné ?) puis redémarrez. Une invite devrait apparaître à l'écran. Vous pouvez alors taper Entrée pour débiter le processus d'installation avec les options d'amorçage par défaut. Si vous le souhaitez, vous pouvez plutôt spécifier des options et un noyau particuliers avant de taper Entrée.

Plusieurs noyaux sont disponibles sur nos CD d'installation. Le noyau par défaut est gentoo. Les autres sont utilisés avec du matériel particulier et des variantes -nofb existent pour désactiver le «framebuffer». **gentoo** Le noyau par défaut avec support des processeurs K8 avec NUMA. Voici les options disponibles.

```
agpgart - Active le support AGP.
acpi=on - Active le support ACPI.
ide=nodma - Désactive le support DMA.
doscsi - Active le support pour le matériel SCSI.
dopcmcia - Active le support pour le matériel PCMCIA.
nofirewire - Désactive le support pour le matériel FireWire.
nokeymap - Désactive le support des claviers internationaux.
docache - Charge les fichiers du CD en mémoire pour pouvoir
éjecter le CD et en utiliser un autre.
nodetect - Désactive la détection automatique.
nousb - Désactive le support du matériel USB.
nodhcp - Ne pas utiliser DHCP pour obtenir une adresse IP.
nohotplug - Désactive hotplug (un programme de chargement dans le noyau).
noapic - Désactive l'apic (p.ex. en cas de problèmes avec le matériel.)
noevms Désactive le chargement des modules EVMS2
nolvm2 Désactive le chargement des modules LVM2
hdx=stroke - Permet de partitionner tout le disque dur même si le
BIOS ne reconnaît pas les disques de grande capacité.
noload=module1[,module2[,...]] - Désactive le chargement des modules spécifiés
```

Vous verrez alors un autre écran de chargement avec une barre de progression. Si vous installez Gentoo sur un système équipé d'un clavier international (non US), utilisez immédiatement

Alt+F1 pour basculer vers un écran plus détaillé et suivez les indications. Si vous ne sélectionnez pas votre type de clavier dans les dix premières secondes, le clavier par défaut (US) sera sélectionné. Lorsque le processus de chargement est terminé, vous serez automatiquement connecté à Gentoo Linux Live en tant qu'utilisateur `root` (qui est le superutilisateur). Vous devriez avoir une invite de commande `root` («#») sur la console active. Vous pouvez aussi passer aux autres consoles en tapant Alt+F2, Alt+F3 et Alt+F4. Retournez à la première console en tapant Alt+F1.

Si par hasard vous n'avez pas eu le temps de saisir votre type de clavier, lorsque vous aurez accès à votre console `root`, chargez manuellement le clavier français :

```
# loadkeys fr
```

Lorsque le CD d'installation démarre, il tente de détecter votre matériel et charge les modules du noyau appropriés pour supporter ce matériel. Dans la plupart des cas cette auto-détection donne d'excellents résultats. Toutefois, il se peut que les modules du noyau dont vous avez besoin ne soient pas tous chargés. Si l'auto-détection PCI a ignoré une partie de votre matériel, vous devrez charger les modules appropriés manuellement.

3 Préparation du disque

A la fin du démarrage du CDRom, vous avez accès à une console `root`. C'est à partir de là que l'on va installer la distribution. Avant l'installation, nous allons préparer le disque dur. Nous allons créer plusieurs partitions à l'aide de l'outil `fdisk`.

Les partitions sont divisées en trois types : primaire, étendue et logique. Une partition primaire est une partition dont l'information est stockée dans le MBR (« master boot record », secteur d'amorçage principal). Comme un MBR est très petit (512 bytes), seules 4 partitions primaires peuvent y être définies (par exemple, de `/dev/hda1` à `/dev/hda4`). Une partition étendue est une partition primaire spéciale (ce qui signifie que la partition étendue doit être une des quatre partitions primaires possibles) qui contient d'autres partitions. Une telle partition n'existait pas à l'origine, mais étant donné que quatre partitions ne suffisaient plus, on a étendu le schéma de formatage sans perdre la compatibilité ascendante. Une partition logique est une partition contenue dans la partition étendue. Sa définition n'est pas placée dans le MBR, mais est déclarée dans la partition étendue.

Nous allons concevoir un plan de partitionnement. Le nombre de partitions dépend beaucoup de votre environnement. Dans la cas d'une station de travail, nous opterons pour 4 partitions :

- ▷ une partition d'une centaine de Mo, pour le répertoire `/boot`
- ▷ une partition de swap, d'une taille équivalente à la mémoire
- ▷ une partition qui recevra la racine `/` du système de fichier
- ▷ une partition qui recevra les données des utilisateurs `/home`

`fdisk` est un outil populaire et puissant pour diviser votre disque en partitions. Lancez `fdisk` sur votre disque (dans notre exemple, nous utilisons `/dev/hda`) :

```
# fdisk /dev/hda/
```

```
Disque /dev/hda: 20.0 Go, 20003880960 octets
255 têtes, 63 secteurs/piste, 2432 cylindres
Unités = cylindres de 16065 * 512 = 8225280 octets
```

Périphérique	Amorce	Début	Fin	Blocs	Id	Système
/dev/hda1	*	1	1657	13309821	c	Win95 FAT32 (LBA)
/dev/hda2		1658	2432	6225187+	5	Extended
/dev/hda5		1658	1682	200781	82	Linux swap
/dev/hda6		1683	2191	4088511	83	Linux
/dev/hda7		2192	2432	1935801	83	Linux

Commande (m pour l'aide):

Si comme moi vous avez un premier disque avec un autre système d'exploitation vous aurez le partitionnement suivant :

```
biboo etc # fdisk -l /dev/hdb
```

```
Disque /dev/hdb: 60.0 Go, 60022480896 octets
255 têtes, 63 secteurs/piste, 7297 cylindres
Unités = cylindres de 16065 * 512 = 8225280 octets
```

Périphérique	Amorce	Début	Fin	Blocs	Id	Système
/dev/hdb1	*	1	13	104391	83	Linux
/dev/hdb2		14	138	1004062+	82	Linux swap / Solaris
/dev/hdb3		139	3178	24418800	83	Linux
/dev/hdb4		3179	7297	33085867+	83	Linux

```
biboo etc #
```

Nous allons ensuite formater les différentes partitions à l'aide des commandes correspondantes suivantes :

Système de fichiers	Commande de création
ext2	mke2fs
ext3	mke2fs -j
reiserfs	mkreiserfs
xf	mkfs.xfs
jfs	mkfs.jfs

Par exemple, pour formater la partition de démarrage (/dev/hda1 dans notre exemple) en **ext2**, la partition de swap (/dev/hda2) et les partitions principales (/dev/hda3 et /dev/hda4 dans notre exemple) en **ext3**, nous utiliserons les commandes suivantes :

```
# mke2fs /dev/hda1
# mke2fs -j /dev/hda3
# mke2fs -j /dev/hda4
```

mkswap est la commande utilisée pour initialiser la partition de mémoire virtuelle :

```
# mkswap /dev/hda2/
```

Pour activer la partition de mémoire virtuelle, utilisez **swapon**

```
# swapon /dev/hda2/
```

Attention : Si comme moi vous avez un premier disque avec un autre système d'exploitation il faut remplacer **hda** par **hdb**.

Maintenant que nos partitions sont initialisées et contiennent un système de fichiers, il est temps de les monter avec la commande **mount**. N'oubliez pas de créer les points de montage nécessaires

pour toutes les partitions que vous avez créées. Par exemple, pour monter les partitions de démarrage et racine :

```
# mount /dev/hda3 /mnt/gentoo
# mkdir /mnt/gentoo/boot
# mount /dev/hda1 /mnt/gentoo/boot
```

4 Installation des fichiers d'installation Gentoo

Avant de poursuivre, vous devez régler l'heure et la date de votre système. Si l'horloge de votre machine n'est pas à l'heure et surtout à la bonne date, des effets indésirables se produiront. Pour changer la date et l'heure de votre système, utilisez date MMJJhhmmAAAA (Mois, Jour, heure, minute, Année). Par exemple, pour le 29 mars 2005 à 16 :21, utilisez :

```
# date
Fri Mar 29 16:21:18 CEST 2005
# date 032916212005
```

Les archives étape incluses sur le CD se trouvent dans le répertoire `/mnt/cdrom/stages`. Pour afficher la liste des étapes disponibles, utilisez `ls` :

```
# ls /mnt/cdrom/stages
ls: /mnt/cdrom/stages: No such file or directory
# mount /dev/cdroms/cdrom0 /mnt/cdrom
# ls /mnt/cdrom/stages
# cd /mnt/gentoo
```

Nous allons maintenant extraire l'archive étape de votre choix. Nous le ferons avec l'outil `tar`. Faites bien attention à utiliser les mêmes options (`xjvpf`). Le `x` signifie extraire, le `v` verbeux et affiche la liste de tous les fichiers extraits (celui-ci est facultatif), le `j` décompresser avec `bzip2`, le `p` préserver les permissions et le `f` veut dire que nous désarchivons un fichier d'archive, pas l'entrée standard. Dans l'exemple suivant, nous décompressons l'archive étape `stage3-<sous-architecture>-2005.1-r1.tar.bz2`. N'oubliez pas de substituer le nom de l'étape par celle que vous aurez choisie.

```
# tar xvpf /mnt/cdrom/stages/stage3-<sous-architecture>-2005.1-r1.tar.bz2
```

Vous devez maintenant installer un instantané de l'arbre Portage. Celui-ci contient l'ensemble des fichiers qui permettent à Gentoo d'installer des paquets, les différents profils, etc. Installer l'instantané de Portage du CD d'installation Pour installer l'instantané du CD d'installation, regardez dans `/mnt/cdrom/snapshots/` pour voir quel instantané est disponible :

```
# ls /mnt/cdrom/snapshots
```

Décompressez ensuite l'archive avec la commande suivante. Assurez-vous de passer les bonnes options à `tar`. De plus, le `-C` est un `C` majuscule, pas un `c`. Dans l'exemple suivant, nous utilisons `portage-<date>.tar.bz2` comme nom d'instantané. Utilisez le nom de votre archive.

```
# tar xvjf /mnt/cdrom/snapshots/portage-<date>.tar.bz2 -C /mnt/gentoo/usr
```

Vous devez aussi copier les sources qui se trouvent sur le CD d'installation sur votre système.

```
# mkdir /mnt/gentoo/usr/portage/distfiles
# cp /mnt/cdrom/distfiles/* /mnt/gentoo/usr/portage/distfiles/
```

Pour optimiser Gentoo, vous pouvez définir quelques variables qui influencent le comportement de Portage. Toutes ces variables peuvent être définies comme des variables d'environnement (en utilisant `export`), mais elles ne sont dans ce cas pas permanentes. Pour conserver votre configuration, vous pouvez utiliser `/etc/make.conf`, un fichier de configuration de Portage. C'est ce fichier que nous allons éditer maintenant.

Note : Une liste commentée de toutes les variables de Portage se trouve dans le fichier `/etc/make.conf(.example)`. Pour installer Gentoo avec succès, seules celles mentionnées ci-dessous sont indispensables.

```
# nano -w /mnt/gentoo/etc/make.conf
```

Comme vous l'avez sans doute remarqué, le fichier `make.conf.example` est structuré de manière générique : les lignes de commentaires commencent par un `#`, les autres définissent des variables en utilisant la syntaxe `VARIABLE="contenu"`. Le fichier `make.conf` utilise la même syntaxe. Certaines variables sont décrites ci-dessous.

CHOST La variable `CHOST` définit l'architecture de votre machine. Sa valeur devrait déjà être définie. Ne modifiez pas cette variable, car cela rendrait votre système inutilisable. Si la valeur de la variable `CHOST` ne vous semble pas correcte, il est possible que vous ayez utilisé une mauvaise archive `stage3`.

CFLAGS et CXXFLAGS Les variables `CFLAGS` et `CXXFLAGS` définissent les options d'optimisation pour le compilateur `gcc`, respectivement en C et C++. Bien que nous les définissions de manière générale ici, vous n'obtiendrez des performances maximales qu'en fixant les optimisations individuellement pour chaque programme. La raison en est que chaque programme est différent.

Un premier paramètre est l'option `-march=` qui spécifie le nom de l'architecture cible. Les options possibles sont décrites dans le fichier `make.conf` (en commentaires). Par exemple pour l'architecture x86 Athlon XP :

```
# Les utilisateurs d'AMD64 qui veulent un système 64 bits natif
#doivent utiliser -march=k8.
# Pour une machine EM64T, il faut utiliser -march=nocona.
-march=athlon-xp
```

Un deuxième paramètre est l'option `-O` (la lettre O majuscule) qui spécifie la classe d'optimisation de `gcc`. Les classes possibles sont `s` (pour optimiser en taille), `0` (zéro, pour ne pas optimiser), `1`, `2`, `3` pour plus d'optimisation de la vitesse d'exécution (chacune de ces classes a les mêmes options que celle qui la précède plus quelques autres). Par exemple, pour une optimisation de classe 2 :

```
-O2
```

Une autre option d'optimisation populaire est `-pipe` (utilise des tubes plutôt que des fichiers temporaires pour la communication entre les différentes étapes de la compilation). Lorsque vous définissez les variables `CFLAGS` et `CXXFLAGS`, vous devez combiner plusieurs options d'optimisation, comme dans l'exemple suivant :

```
CFLAGS="-march=athlon-xp -pipe -O2" # Utilisez -march=k8 pour une machine AMD64.
# Utilisez -march=nocona pour une machine EM64T.
CXXFLAGS="${CFLAGS}" # Utilise les mêmes paramètres pour les deux variables.
```

Note : `-march=k8`, `-march=athlon64` et `-march=opteron` sont équivalents.

Attention : N'utilisez jamais `-fPIC`, `-m32` ou `-m64` dans vos `CFLAGS`. Ils sont ajoutés automatiquement quand ils sont requis. De plus, `-Os` n'est pas supporté.

Vous devriez utiliser des `CFLAGS` simples pour éviter des problèmes ultérieurs. Par exemple, `CFLAGS="-march=k8 -O2 -pipe"` fonctionne très bien sur un Athlon64 ou Opteron, et `CFLAGS="-march=no`

marche tout aussi bien pour les processeurs EM64T. Pour un environnement chroot 32 bits, utilisez `CFLAGS="-march=athlon-xp -O2 -pipe -msse2"` sur des processeurs AMD64 ou `CFLAGS="-march=pe` pour des EM64T à l'intérieur du chroot.

MAKEOPTS Avec **MAKEOPTS**, vous pouvez définir le nombre de compilations à lancer en parallèle. Une valeur souvent utilisée est le nombre de processeurs dans votre système plus un, mais une autre valeur peut parfois mieux fonctionner.

```
MAKEOPTS="-j2"
```

Mettez à jour votre `/mnt/gentoo/etc/make.conf` comme vous le souhaitez, et sauvez (Ctrl-X avec nano). Vous êtes maintenant prêt à continuer avec l'installation d'un système de base Gentoo.

Voici en exemple mon fichier `/mnt/gentoo/etc/make.conf`

```
# These settings were set by the catalyst build script that automatically built
this stage
# Please consult /etc/make.conf.example for a more detailed example
LINGUAS="fr"
CFLAGS="-march=k8 -pipe -O2"
CHOST="x86_64-pc-linux-gnu"
CXXFLAGS="${CFLAGS}"
MAKEOPTS="-j2"

GENTOO_MIRRORS="http://ftp.belnet.be/mirror/rsync.gentoo.org/gentoo/"
SYNC="rsync://rsync.europe.gentoo.org/gentoo-portage"

USE="mmx mmxext sse 3dnow 3dnowext apache2 esd ffmpeg network nsplugin v4l
win32codecs xine gtk gtk2 gnome cdparanoia bash-completion cups encode
foomaticdb glut ftp imagemagick lesstif mysql ogg oggvorbis threads win32codecs
dvd alsa cdr -arts -kde -ldap -qt"

PORTDIR_OVERLAY="/usr/local/portage"
```

Après moultes essais et modifications de mon fichier `make.conf`, voici la version final (pour l'instant) :

```
# These settings were set by the catalyst build script that automatically built this st
# Please consult /etc/make.conf.example for a more detailed example
LINGUAS="fr"
CFLAGS="-march=k8 -pipe -O2"
CHOST="x86_64-pc-linux-gnu"
CXXFLAGS="${CFLAGS}"
MAKEOPTS="-j2"

#GENTOO_MIRRORS="http://ftp.belnet.be/mirror/rsync.gentoo.org/gentoo"
SYNC="rsync://rsync.europe.gentoo.org/gentoo-portage"

PORTAGE_ELOG_SYSTEM="save mail"
PORTAGE_ELOG_CLASSES="info warn error log"
PORTAGE_ELOG_MAILURI="morere@univ-metz.fr /usr/sbin/sendmail"
PORTAGE_ELOG_MAILFROM="yann@athome.org"

USE="X pdf mmx mmxext sse 3dnowext apache2 esd ffmpeg network nsplugin v4l
win32codecs xine gtk gtk2 gnome cdparanoia bash-completion cups encode
foomaticdb glut ftp imagemagick ipv6 lesstif mysql mysqli nls spell eds opengl
pam xv type1-fonts bitmap-fonts truetype-fonts ogg oggvorbis ssl threads
readline foomaticdb gif png jpeg quicktime xml sdl java crypt pdflib
berkdb ncurses tcpd tiff mp3 audiofil doc curl gd bzip2 lcms perl
bash-completion quicktime mad vorbis openal dvdread xvid svg
mpeg truetype dvd dvdr alsa cdr gnome hal howl mikmod zlib
v4l2 mjpeg lzo fame dv -extrafilters mplayer subtitles rar
-arts -kde -ldap -qt nptl nptlonly kernel_linux mono pcre session unicode"

ALSA_CARDS="intel8x0"
INPUT_DEVICES="keyboard mouse"
VIDEO_CARDS="nv nvidia vesa fbdev"

FEATURES="-strict" #Ã©vitez les problÃ©mes de security violation

#PORTDIR_OVERLAY="/usr/local/portage"
```

5 Installation d'un système de base Gentoo

Monter le système de fichiers `/proc` dans `/mnt/gentoo/proc` permet à l'installation d'utiliser les informations fournies par le noyau, même lorsqu'on se trouve dans l'environnement `chroot`.

```
# mount -t proc none /mnt/gentoo/proc
```

Si vous avez configuré une connexion réseau, il reste une dernière chose à faire avant d'entrer dans le nouvel environnement. Il s'agit de copier l'information DNS de `/etc/resolv.conf`. Vous devez le faire afin d'assurer le bon fonctionnement du réseau dans le nouvel environnement. `/etc/resolv.conf` contient les serveurs de noms pour votre réseau.

```
# cp -L /etc/resolv.conf /mnt/gentoo/etc/resolv.conf
```

Maintenant que toutes les partitions sont initialisées et que l'environnement de base est installé, il est temps d'entrer dans notre nouvel environnement d'installation. Cela signifie que l'on passe

de l'environnement d'installation actuel (CD d'installation ou autre environnement d'installation) à l'environnement de votre système (soit les partitions initialisées). L'entrée se fait en trois étapes. D'abord, on change la racine de / (sur l'environnement d'installation) en `/mnt/gentoo` (sur vos partitions) en utilisant `chroot`. Ensuite, on crée un nouvel environnement en utilisant `env-update` dont l'effet est essentiellement de créer les variables d'environnement. Finalement, ces variables sont chargées en mémoire en utilisant `source`.

```
# chroot /mnt/gentoo /bin/bash
# env-update
* Caching service dependencies...
# source /etc/profile
```

L'arbre Portage a déjà été installé, mais vous devez maintenant créer le cache de Portage pour améliorer les temps de réponse quand vous installerez des paquets.

```
# emerge --metadata
```

La variable USE `USE` est une des plus puissantes variables mises à la disposition des utilisateurs de Gentoo. Plusieurs programmes peuvent être compilés avec ou sans le support optionnel disponible pour certaines fonctionnalités. Par exemple, certains programmes peuvent être compilés avec un support pour `gtk` ou pour `Qt`. D'autres peuvent être compilés avec ou sans support pour `SSL`. Certains programmes peuvent même être compilés avec un support pour le «frame-buffer» (`svglib`) plutôt que pour `X11` (serveur `X`). La plupart des distributions compilent leurs paquets avec un support aussi complet que possible, augmentant ainsi la taille des programmes et le temps de chargement, sans mentionner le nombre énorme de dépendances qui en résulte. Avec Gentoo, vous pouvez définir les options à utiliser lors de la compilation d'un paquet. C'est ici que la variable `USE` entre en jeu. La variable `USE` contient des mots-clés que vous choisissez et qui correspondent à des options de compilation. Par exemple, `ssl` compilera le support `ssl` dans les programmes qui le supportent. `-X` retirera le support pour le serveur `X` (remarquez le signe moins devant le mot-clé). `gnome gtk -kde -qt` compilera vos programmes avec le support pour `GNOME` (et `gtk`), mais sans le support pour `KDE` (et `qt`). Le résultat est un système complètement réglé pour `GNOME`.

Les options par défaut pour `USE` se trouvent dans les fichiers `make.defaults` de votre profil. Vous trouverez ces fichiers `make.defaults` dans le répertoire sur lequel le lien `/etc/make.profile` pointe, ainsi que dans tous ses répertoires parents. Les options par défaut de `USE` sont donc la somme de toutes les options `USE` de ces fichiers. Vos modifications à `/etc/make.conf` sont jugées en fonction de ces options par défaut. Si vous ajoutez quelque chose aux options `USE`, c'est ajouté à la liste par défaut. Si vous retirez quelque chose des options `USE` (en le précédant du signe moins), c'est retiré de la liste par défaut (en supposant que cela s'y trouvait). Ne modifiez jamais quoi que ce soit dans le répertoire `/etc/make.profile` car ces fichiers sont écrasés lors des mises à jour de Portage !

Une description complète des options disponibles se trouve dans le fichier `/usr/portage/profiles/use.desc` qui devrait déjà être sur votre système.

```
# less /usr/portage/profiles/use.desc
(Utilisez les flèches de votre clavier pour faire défiler le texte et
tapez 'q' pour quitter.)
# nano -w /etc/make.conf
USE="-gtk -gnome qt kde dvd alsa cdr"
```

Voici ma variable `USE`

```
USE="mmx mmxext sse 3dnow 3dnowext apache2 esd ffmpeg network nsplugin v4l  
win32codecs xine gtk gtk2 gnome cdparanoia bash-completion cups encode  
foomaticdb glut ftp imagemagick lesstif mysql ogg oggvorbis threads win32codecs  
dvd alsa cdr -arts -kde -ldap -qt"
```

6 Configurer et compiler le noyau

Vous devez maintenant choisir votre fuseau horaire afin que votre système sache où il se trouve. Cherchez votre fuseau horaire dans `/usr/share/zoneinfo`, puis copiez-le sur `/etc/localtime`. Évitez les zones `/usr/share/zoneinfo/Etc/GMT*`, car leur nom prête à confusion. En effet, GMT-8 signifie en fait GMT+8.

```
# ls /usr/share/zoneinfo  
(En supposant que vous utilisez l'heure de Paris.)  
# cp /usr/share/zoneinfo/Europe/Paris /etc/localtime
```

Pour les systèmes AMD64, nous offrons les `gentoo-sources` (les sources du noyau 2.6 auxquelles sont appliqués des correctifs spécifiques pour AMD64 qui améliorent la stabilité, les performances et le support matériel). Choisissez les sources à utiliser pour votre noyau et installez-les avec `emerge`. On utilise `USE="-doc"` pour éviter d'éventuelles dépendances telles que `xorg-x11`.

```
# USE="-doc" emerge gentoo-sources
```

Si vous examinez le contenu de `/usr/src`, vous devriez voir un lien symbolique nommé `linux` pointant vers les sources de votre noyau :

```
# ls -l /usr/src/linux  
lrwxrwxrwx 1 root root 12 Oct 13 11:04 /usr/src/linux -> linux-2.6.12-gentoo-r10
```

Si ce n'est pas le cas (le lien symbolique pointe vers un noyau différent), changez le lien avant de continuer :

```
# rm /usr/src/linux  
# cd /usr/src  
# ln -s linux-2.6.12-gentoo-r10 linux
```

Il est maintenant temps de configurer et de compiler votre noyau. Vous pouvez utiliser **genkernel**. Cette commande construira un noyau générique tel que celui utilisé par le CD d'installation.

Maintenant que l'arbre des sources de votre noyau est installé, il est temps de compiler ce noyau à l'aide du script `genkernel` qui construira automatiquement un noyau dont la configuration sera presque identique à celle du noyau du CD d'installation. Cela signifie que si vous utilisez `genkernel` pour construire votre noyau, votre système détectera généralement tout votre matériel au moment de l'amorçage, à la manière du CD d'installation. Puisque **genkernel** ne requiert aucune configuration manuelle, il s'agit d'une solution idéale pour l'utilisateur rebuté par l'idée de compiler son propre noyau.

Maintenant, voyons comment utiliser `genkernel`. D'abord, installez **genkernel** comme suit :

```
# emerge genkernel
```

Ensuite, compilez les sources du noyau en exécutant `genkernel all`. Puisque `genkernel` compile un noyau qui supporte presque n'importe quel matériel, ne soyez pas surpris que la compilation demande un temps considérable. Veuillez noter que si votre partition de démarrage utilise un autre système de fichiers que `ext2` ou `ext3`, vous devez compiler le support pour ce système de fichiers dans le noyau (donc pas comme module) avec la commande `genkernel --menuconfig all`.

```
genkernel --menuconfig all
```

Lorsque `genkernel` aura fini son travail, un noyau, un ensemble complet de modules et un «initial root disk» (`initrd`) auront été créés. Le noyau et le `initrd` seront utilisés plus tard lors de la

configuration du chargeur de démarrage. Notez bien les noms du noyau et du initrd puisque vous devrez les spécifier lors de l'écriture du fichier de configuration du chargeur de démarrage. Le initrd sera démarré immédiatement après l'amorçage afin de réaliser l'autodétection du matériel (tout comme pour le CD d'installation) avant que votre «véritable» système ne démarre.

Afin d'obtenir un système plus semblable encore à celui du CD d'installation, effectuons une dernière étape : l'installation de `coldplug`. Alors que initrd autodétecte le matériel nécessaire au démarrage du système, `coldplug` autodétecte tout le reste. Pour l'installer et l'activer, utilisez les commandes suivantes :

```
# emerge coldplug
# rc-update add coldplug default
```

Vous devriez indiquer la liste des modules que vous souhaitez charger automatiquement dans `/etc/modules.autoload.d/kernel-2.6`. Vous pouvez également ajouter des options aux modules si vous le souhaitez.

Pour dresser la liste des modules disponibles, exécutez la commande `find` tel qu'indiqué ci-dessous. N'oubliez pas de substituer `<kernel version>` par la version du noyau que vous venez juste de compiler :

```
# find /lib/modules/<kernel version>/ -type f -iname '*.o' -or -iname '*.ko'
```

Par exemple, pour charger automatiquement le module `3c59x.o`, spécifiez-le dans le fichier `kernel-2.6`.

```
# nano -w /etc/modules.autoload.d/kernel-2.6
3c59x
```

7 Configurer le système

Sous Linux, toutes les partitions utilisées par le système doivent être listées dans `/etc/fstab`. Ce fichier contient l'information relative aux points de montage de ces partitions (où elles se situent dans le système de fichiers de Linux), à la façon dont elles sont montées (décrite par des options spéciales) et aux circonstances de leur montage (qui peut être automatique ou non, sous le contrôle des utilisateurs ou non, etc.).

`/etc/fstab` emploie une syntaxe particulière. Chaque ligne contient six champs séparés par des blancs (un ou plusieurs espaces ou tabulations, ou encore un mélange d'espaces et de tabulations). Chaque champ a une signification particulière :

- ▷ Le premier champ indique la partition (il s'agit du chemin d'accès vers le fichier matériel).
- ▷ Le second champ indique le point de montage où la partition sera montée.
- ▷ Le troisième champ indique le type de système de fichiers présent sur la partition.
- ▷ Le quatrième champ indique les options de montage utilisées par `mount` lorsque cette commande tente de monter la partition. Puisque chaque type de système de fichiers a ses propres options de montage, vous êtes encouragé à lire la page `man` de la commande `mount` (`man mount`) pour obtenir une liste complète de ces options. Des options de montage multiples doivent être séparées par des virgules.
- ▷ Le cinquième champ est utilisé par `dump` pour déterminer si la partition doit être «dumpée» ou non. Vous pouvez généralement laisser cette valeur à 0 (zéro).
- ▷ Le sixième champ est utilisé par `fsck` afin de déterminer l'ordre dans lequel les systèmes de fichiers doivent être vérifiés si le système n'a pas été mis hors tension de façon appropriée. Pour le système de fichiers racine, la valeur devrait être 1. Pour les autres systèmes de fichiers, la valeur devrait être 2 (ou 0 s'il n'est pas nécessaire de vérifier le système de fichiers).

Vous devez modifier le fichier `/etc/fstab` qui a été installé par Gentoo, car celui-ci n'est qu'un exemple et votre système ne démarrera pas si vous le laissez tel quel. Ouvrez `nano` (ou votre éditeur favori) pour créer votre `/etc/fstab` :

```
# nano -w /etc/fstab/
```

un exemple de fichier `fstab`

```
/dev/hda1      /boot          ext2           noauto,noatime      1 2
/dev/hda2      none           swap          sw                   0 0
/dev/hda3      /              ext3           noatime              0 1
/dev/hda4      /home          ext3           noatime              0 1
none /proc        proc           defaults          0 0
none /dev/shm tmpfs nodev,nosuid,noexec 0 0
/dev/cdroms/cdrom0 /mnt/cdrom auto noauto,user 0 0
```

Si comme moi, votre premier disque contient windows, et le second la distribution Gentoo vous avez le fichier suivant :

```
/dev/hdb1      /boot          ext2           noauto,noatime      1 2
/dev/hdb2      none           swap          sw                   0 0
/dev/hdb3      /              ext3           noatime              0 1
/dev/hdb4      /home          ext3           noatime              0 1
none /proc        proc           defaults          0 0
none /dev/shm tmpfs nodev,nosuid,noexec 0 0
/dev/cdroms/cdrom0 /mnt/cdrom auto noauto,user 0 0
```

Information réseau Une des choses que chaque utilisateur doit faire est nommer son PC. Cela peut sembler aisé, mais de nombreux utilisateurs ont bien du mal à trouver un nom approprié pour leur PC-Linux. Afin d'accélérer les choses, dites-vous que le nom que vous choisirez maintenant pourra être changé plus tard. Si vous êtes embêté, nommez temporairement votre système tux et choisissez homenetwork comme nom de domaine. Nous utiliserons ces valeurs dans les exemples suivants. Premièrement, définissons le nom d'hôte, nom de domaine :

```
# nano -w /etc/conf.d/hostname
(Définissez le nom de votre machine :)
HOSTNAME="tux"
# nano -w /etc/conf.d/domainname
(Définissez le nom de votre domaine :)
DNSDOMAIN="homenetwork"
```

Maintenant, ajoutez le script `domainname` au niveau d'exécution «default» :

```
# rc-update add domainname default
```

Configurer le réseau Toute l'information réseau est rassemblée dans `/etc/conf.d/net`. Ce fichier utilise une syntaxe simple mais pas nécessairement intuitive si vous ne savez pas comment paramétrer manuellement un réseau. Pas d'inquiétude, tout vous sera expliqué. Un exemple commenté complet se trouve dans le fichier `/etc/conf.d/net.example`. DHCP est utilisé par défaut et ne demande aucune intervention de votre part. Si vous devez configurer votre réseau soit pour spécifier des options particulières pour DHCP, soit parce que vous n'utilisez pas DHCP, ouvrez le fichier `/etc/conf.d/net` avec votre éditeur favori :

```
# nano -w /etc/conf.d/net
# This blank configuration will automatically use DHCP for any net.*
# scripts in /etc/init.d. To create a more complete configuration,
# please review /etc/conf.d/net.example and save your configuration
# in /etc/conf.d/net (this file :!)).
```

Pour entrer une adresse fixe, un masque de réseau et une adresse de passerelle, vous devez définir `config_eth0` et `routes_eth0` :

```
config_eth0=( "192.168.0.2 netmask 255.255.255.0 brd 192.168.0.255" )
routes_eth0=( "default gw 192.168.0.1" )
```

Pour utiliser DHCP et lui passer des options, définissez les variables `config_eth0` et `dhcp_eth0` :

```
config_eth0=( "dhcp" )
dhcp_eth0="nodns nontp nonis"
```

Le fichier `/etc/conf.d/net.example` contient une série d'exemples commentés pour vous aider à configurer votre réseau.

Sauvegardez votre configuration, puis quittez l'éditeur afin de poursuivre. Activer les connexions réseau automatiquement au démarrage Pour que vos interfaces réseau soient activées automatiquement lors du démarrage, vous devez les ajouter au niveau d'exécution «default».

```
# rc-update add net.eth0 default
```

Modifions maintenant le fichier `/etc/hosts` :

```
# nano -w /etc/hosts/
127.0.0.1 localhost tux
```

Information système Pour commencer, définissons le mot de passe root en tapant :

```
# passwd
```

Si vous voulez pouvoir vous identifier en tant que root en passant par la console série, ajoutez `tts/0` à `/etc/securetty` :

Gentoo utilise `/etc/rc.conf` pour la configuration générale qui s'applique à l'ensemble du système. Ouvrez `/etc/rc.conf` et appréciez les commentaires qui s'y trouvent :)

```
# nano -w /etc/rc.conf
```

Ensuite, sauvez votre fichier et quittez votre éditeur. Comme vous pouvez le voir, ce fichier est généreusement commenté afin de vous aider à paramétrer les différentes variables relatives à la configuration. Vous pouvez configurer votre système pour qu'il utilise unicode. Vous pouvez aussi définir votre éditeur par défaut et votre gestionnaire d'affichage préféré (gdm ou kdm). Le fichier `/etc/conf.d/keymaps` permet de spécifier le type de clavier que vous utilisez.

```
# nano -w /etc/conf.d/keymaps
```

La valeur que vous attribuez à la variable `KEYMAP` détermine la disposition des touches de votre clavier. Si vous choisissez une valeur incorrecte, vous serez surpris quand vous taperez sur votre clavier.

8 Installation des outils système

Pour installer le système de journalisation de votre choix, utilisez `emerge` puis ajoutez le au niveau d'exécution «default» avec la commande `rc-update`. L'exemple suivant installe `syslog-ng`. Bien sûr, n'oubliez pas d'y substituer le nom de votre système de journalisation.

```
# emerge syslog-ng
# rc-update add syslog-ng defaultd
```

Seul le paquet `vixie-cron` est disponible lors d'une installation sans réseau. Si vous préférez en installer un autre, vous pouvez attendre et l'installer quand vous le pourrez.

```
# emerge vixie-cron
# rc-update add vixie-cron default
(Seulement si vous avez choisi dcron ou fcron.) # crontab /etc/crontab
```

Si vous voulez indexer vos fichiers pour pouvoir les retrouver rapidement grâce à l'outil `locate`, vous devez installer le paquet `sys-apps/slocate`.

```
# emerge slocate
```

Si vous voulez que votre système acquière une adresse IP automatiquement, vous devez installer `dhcpcd` (ou tout autre client DHCP). Si vous ne le faites pas, vous risquez de ne pas pouvoir vous connecter à internet après avoir fini l'installation.

```
# emerge dhcpcd
```

9 Configurer le chargeur de démarrage

Maintenant que votre noyau est configuré et compilé et que les fichiers de configuration du système sont correctement paramétrés, il est temps d'installer le programme qui charge le noyau au démarrage de votre machine. Un tel programme est appelé un chargeur de démarrage («bootloader» en anglais).

Pour installer GRUB, il suffit d'une simple commande `emerge`.

```
# emerge grub
```

Bien que GRUB soit maintenant installé, vous devez encore écrire son fichier de configuration pour qu'il puisse démarrer votre noyau et installer grub dans votre secteur de démarrage (le MBR). Créez le fichier `/boot/grub/grub.conf` avec `nano`, ou, si disponible, avec un autre éditeur :

```
# nano -w /boot/grub/grub.conf
default 0
timeout 30
splashimage=(hd0,0)/boot/grub/splash.xpm.gz
title=Gentoo Linux
root (hd0,0)
kernel /boot/kernel-genkernel-amd64-2.6.12-gentoo-r10 root=/dev/ram0
  init=/linuxrc ramdisk=8192 real_root=/dev/hda3 udev
initrd /boot/initramfs-genkernel-amd64-2.6.12-gentoo-r10
# Uniquement pour démarrer un autre système.
title=Windows XP
root (hd0,5)
makeactive
chainloader +1
```

Si comme moi, votre premier disque contient windows, et le second la distribution Gentoo vous avez le fichier suivant :

```
# nano -w /boot/grub/grub.conf
default 0
timeout 30
splashimage=(hd1,0)/boot/grub/splash.xpm.gz
title=Gentoo Linux
root (hd1,0)
kernel /boot/kernel-genkernel-amd64-2.6.12-gentoo-r10 root=/dev/ram0
  init=/linuxrc ramdisk=8192 real_root=/dev/hdb3 udev
initrd /boot/initramfs-genkernel-amd64-2.6.12-gentoo-r10
# Uniquement pour démarrer un autre système.
title=Windows XP
root (hd0,0)
makeactive
chainloader +1
```

Si vous devez passer des options supplémentaires à votre noyau, ajoutez-les simplement à la fin de la commande kernel. Nous lui passons déjà une option (`root=/dev/hda3` ou `real_root=/dev/hda3`), mais vous pouvez en ajouter (par exemple, l'option `video` pour le framebuffer mentionnée plus haut).

```
title=Gentoo Linux
root (hd0,0)
kernel /kernel-2.6.12-gentoo-r10 root=/dev/hda3 video=vesafb
```

Si vous avez configuré votre noyau avec le support pour le framebuffer (ou si vous avez utilisé la configuration de noyau par défaut de genkernel) vous pouvez l'activer en ajoutant `vga` et/ou `video` au fichier de configuration de votre chargeur de démarrage. Avec un système 64 bits, vous devez utiliser le pilote `vesafb` et devez ajouter la mention `vga` à votre configuration. Le paramètre `vga` contrôle la résolution et la profondeur des couleurs de votre écran en framebuffer pour `vesafb`. Comme indiqué dans le fichier `/usr/src/linux/Documentation/fb/vesafb.txt` (qui est installé lors de l'installation d'un paquet de source du noyau), vous devez lui donner un numéro VESA correspondant à la résolution et la profondeur de couleur voulues. Le tableau suivant vous donne la liste des résolutions et profondeurs des couleurs disponibles et vous donne la valeur du nombre que vous devez indiquer pour le paramètre `vga` :

	640x480	800x600	1024x768	1280x1024
256	0x301	0x303	0x305	0x307
32k	0x310	0x313	0x316	0x319
64k	0x311	0x314	0x317	0x31A
16M	0x312	0x315	0x318	0x31B

Le paramètre `video` contrôle les options d'affichage du framebuffer. Il faut lui indiquer le pilote du framebuffer (`vesafb` pour un noyau 2.6) suivi des paramètres de contrôle que vous voulez activer. Toutes les variables sont listées dans le fichier `/usr/src/linux/Documentation/fb/vesafb.txt`. Nous vous présentons donc seulement les trois options les plus utilisées :

ywrap Indique que la carte graphique peut réutiliser sa mémoire (c'est-à-dire continuer au début quand elle approche de la fin).

mtrr Configure les registres MTRR

Un exemple de résultat pour ces deux paramètres pourrait être `vga=0x318 video=vesafb:mtrr,ywrap` ou `video=vesafb:mtrr,ywrap,1024x768-32@85`.

Le script `grub-install` installe GRUB automatiquement, mais, puisque vous êtes dans un environnement «chroot», vous créer le fichier `/etc/mtab` à jour. Ce dernier contient la liste des systèmes de fichiers qui sont montés. Heureusement, il suffit d'utiliser une copie de `/proc/mounts` en évitant la ligne `rootfs` (nécessaire si vous n'avez pas de partition `/boot` séparée). La commande suivante fonctionne avec ou sans partition `/boot` séparée :

```
# grep -v rootfs /proc/mounts > /etc/mtab
# grub-install /dev/hda
```

Sortez de l'environnement «chroot» et démontez toutes les partitions montées. Ensuite, tapez la commande magique tant attendue : **reboot**.

```
# exit
# cd
cdimage ~# umount /mnt/gentoo/boot /mnt/gentoo/proc /mnt/gentoo
cdimage ~# reboot
```

Bien entendu, n'oubliez pas de retirer le CD du lecteur car sinon le système amorcera le CD au lieu de démarrer sur votre nouveau système Gentoo.

10 Finaliser l'installation

Travailler en temps que root sur un système Unix/Linux est dangereux et devrait être évité aussi souvent que possible. Par conséquent, il est vivement recommandé d'ajouter un utilisateur pour une utilisation quotidienne. Les actions qu'un utilisateur a le droit de faire sont définies par les groupes dont l'utilisateur est membre. Le tableau ci-dessous liste quelques groupes importants :

Groupe	Description
audio	accès autorisé aux périphériques audio
cdrom	accès direct autorisé aux lecteurs optiques
floppy	accès direct autorisé au lecteur de disquettes
games	accès aux jeux
portage	permet d'utiliser emerge –pretend
usb	accès autorisé aux périphériques USB
video	accès autorisé au matériel de capture vidéo et à l'accélération matérielle
wheel	commande su utilisable

Par exemple, pour créer un utilisateur nommé yann qui est membre des groupes wheel, users et audio, identifiez-vous en tant qu'utilisateur root (seul root peut créer des comptes) et faites :

```
Login: root
Password: (votre mot de passe root)
# useradd -m -G users,wheel,audio -s /bin/bash yann
# passwd yann
Password: (tapez le mot de passe pour john)
Re-enter password: (retapez-le pour vérifier)
```

Si cet utilisateur a besoin d'utiliser le compte root, il peut utiliser `su -` pour obtenir les privilèges root. Un autre moyen est d'utiliser le paquet `sudo` qui est, s'il est configuré correctement, très sécurisé.

11 Utiliser la commande emerge

Portage est probablement l'innovation de Gentoo la plus remarquable en ce qui concerne la gestion des logiciels. Sa grande flexibilité et ses nombreuses fonctionnalités font parfois dire de Portage qu'il est le meilleur outil de gestion des logiciels pour Linux. Portage a été écrit en Python et en Bash qui sont tous les deux des langages scriptés, c'est-à-dire que 100 % du code source est installé et consultable sur tous les systèmes Gentoo. La plupart des utilisateurs interagiront avec Portage via la commande `emerge`. Ce chapitre n'a pas pour vocation de dupliquer toute l'information disponible dans la page man de `emerge`. Pour consulter la page man, faites :

```
$ man emerge
```

Quand nous parlons de paquets, nous parlons des logiciels qui sont disponibles dans Gentoo grâce à l'arbre de Portage. Celui-ci est un ensemble d'ebuilds qui sont en fait des fichiers qui donnent toutes les informations nécessaires à Portage pour installer un logiciel. Par défaut, ces ebuilds se trouvent dans `/usr/portage`. Dès que vous employez Portage pour une action relative aux paquets, il utilisera les ebuilds de votre système. Il est donc important de maintenir les ebuilds de votre système à jour pour que Portage puisse installer des nouvelles versions des logiciels que vous utilisez ou des correctifs de failles de sécurité.

L'arbre Portage est généralement mis à jour avec `rsync` qui est un outil de transfert de fichiers incrémental. La mise à jour se fait simplement avec la commande `emerge`. L'utilisation de `rsync` est tout à fait transparente :

```
# emerge --sync
```

Pour rechercher un logiciel dans l'arbre Portage, vous pouvez utiliser `emerge`. En effet, la commande `emerge --search` affiche la liste des paquets dont le titre correspond plus ou moins au terme recherché.

Par exemple, pour trouver tous les paquets dont le nom contient «pdf», vous utiliseriez :

```
$ emerge --search pdf
```

Si vous voulez aussi chercher dans les descriptions, utilisez l'option `--searchdesc` (ou `-S`) :

```
$ emerge --searchdesc pdf
```

La liste des paquets affichés contient quelques informations utiles pour chaque paquet.

```
* net-print/cups-pdf
Latest version available: 1.5.2
Latest version installed: [ Not Installed ]
Size of downloaded files: 15 kB
Homepage: http://cip.physik.uni-wuerzburg.de/~vrbehr/cups-pdf/
Description: Provides a virtual printer for CUPS to produce PDF files.
License: GPL-29
```

Une fois que vous avez identifié un paquet que vous voulez installer, il vous suffit d'utiliser la commande `emerge` suivie du nom du paquet pour l'installer. Par exemple, pour installer `gnumeric` :

```
# emerge gnumeric
```

De nombreuses applications dépendent d'autres paquets. Par conséquent, quand vous installez un logiciel, il se peut que Portage en installe d'autres qui sont nécessaires au bon fonctionnement du paquet que vous installez. Si vous voulez connaître la liste des paquets que Portage installerait si vous installiez un paquet donné, vous pouvez utiliser l'option `--pretend`. Un exemple :

```
# emerge --pretend gnumeric
```

Quand vous installez un paquet avec Portage, il télécharge les sources nécessaires et les sauve dans le répertoire `/usr/portage/distfiles`. Ensuite, Portage décompresse l'archive, compile son contenu et installe le logiciel. Si vous voulez télécharger les sources sans installer le paquet, utilisez l'option `--fetchonly`. Par exemple, pour télécharger les sources de `gnumeric` :

```
# emerge --fetchonly gnumeric
```

Trouver la documentation d'un paquet installé De nombreux paquets installent leur propre documentation. Parfois l'option `USE doc` indique si la documentation d'un paquet doit être installée ou non. Vous pouvez vérifier l'existence de l'option `USE doc` avec la commande `emerge -vp <nom du paquet>`.

```
# emerge -vp alsa-lib
[ebuild N ] media-libs/alsa-lib-1.0.9_rc3 +doc -jack 674 kB
```

Pour désinstaller un paquet de votre système, utilisez `emerge --unmerge`. Cette commande supprime les fichiers qui avaient été installés par Portage, mais ne supprime pas les fichiers de configuration si vous les avez modifiés après l'installation. Cela vous permet de réutiliser vos fichiers de configuration si vous réinstallez le paquet plus tard. Cependant, un avertissement est de mise : Portage ne vérifie pas que le paquet que vous supprimez est nécessaire au bon fonctionnement d'un autre paquet. Toutefois, un message s'affichera si vous essayez de supprimer un paquet important dont la disparition causerait de graves problèmes.

```
# emerge --unmerge gnumeric
```

Quand vous supprimez un paquet, les paquets dont il dépend qui avaient été installés initialement ne seront pas désinstallés automatiquement. Pour que Portage recherche les dépendances qui peuvent être supprimées, utilisez l'option `depclean`.

Pour maintenir votre système en bon état et disposer des correctifs de failles de sécurité, vous devriez le mettre à jour régulièrement. Puisque Portage ne se base que sur les ebuilds de votre

machine, vous devez vous assurer que votre arbre Portage est à jour. Une fois votre arbre Portage à jour, vous pouvez mettre votre système à jour avec la commande `emerge --update world` :

```
# emerge --update world
```

Portage recherche alors des versions plus récentes des logiciels que vous avez installés explicitement et uniquement ceux-là. Portage ignorera les paquets qui ont été installés automatiquement pour qu'un paquet que vous avez demandé puisse être installé. Si vous voulez que Portage prenne ces paquets en considération, utilisez l'option `--deep` :

```
# emerge --update --deep world
```

Étant donné que des mises à jour qui corrigent des failles de sécurité sont apportées à des paquets que vous n'avez pas explicitement installés, mais qui ont été installés parce que d'autres paquets en dépendent, il est recommandé d'exécuter la commande ci-dessus de temps en temps. Si vous avez modifié les options de la variable `USE`, vous devriez également ajouter l'option `--newuse` pour que Portage vérifie si certains paquets ne doivent pas être recompilés. Par exemple :

```
# emerge --update --deep --newuse world
```

Prenons l'exemple de mozilla : à quels paramètres `USE` est-il sensible ? Pour le savoir, nous utilisons `emerge` avec les options `--pretend` et `--verbose` :

```
# emerge --pretend --verbose mozilla
These are the packages that I would merge, in order:
Calculating dependencies ...done!
[ebuild N ] net-www/mozilla-1.5-r1 +java +crypt -ipv6 -gtk2 +ssl +ldap
+gnome -debug +mozcalendar -mozaccess -mozxmlterm -moznoirc -moznomail
-moznocompose -moznoxf
```

La variable `ACCEPT_KEYWORDS` définit quelle branche vous voulez utiliser. La valeur par défaut est la branche stable pour votre architecture, par exemple `amd64`. Il est recommandé de n'utiliser que la branche stable. Cependant, si la stabilité des logiciels n'est pas votre première préoccupation ou si vous souhaitez aider Gentoo et envoyer des rapports de bogues sur <http://bugs.gentoo.org>, alors lisez ce qui suit.

Si vous désirez utiliser les versions les plus récentes des logiciels, vous pouvez envisager de passer à la branche de test. Pour cela, ajoutez un tilde devant le nom de votre architecture. La branche de test désigne exactement ce que son nom indique - Test. Si un paquet appartient à cette branche, cela signifie que les développeurs pensent qu'il est fonctionnel mais qu'il n'a pas été suffisamment testé. Vous pouvez très bien être le premier à découvrir un bogue sur le paquet, auquel cas vous devriez remplir un rapport de bogue pour que les développeurs soient au courant du problème. Si vous décidez d'utiliser cette branche de test, attendez-vous à rencontrer des problèmes de stabilité, des paquets imparfaits, notamment en ce qui concerne les dépendances, des mises à jour fréquentes et donc beaucoup de compilations, voire des paquets qui cessent de fonctionner. Si vous ne maîtrisez pas Gentoo ou si vous ne savez pas comment résoudre les problèmes éventuels, il est fortement recommandé de vous en tenir à la branche stable. Par exemple, pour utiliser la branche de test sur une machine x86, modifiez le fichier `/etc/make.conf` comme suit :

```
ACCEPT_KEYWORDS="~amd64"
```

Si vous mettez votre système à jour maintenant, vous constaterez que beaucoup de paquets vont être mis à jour. Veuillez noter qu'une fois passé à la branche de test, il est pratiquement impossible de revenir à la branche stable.

Afin de mettre à jour que certains paquets en version de test, on passe l'option lors de l'installation :

```
# ACCEPT_KEYWORDS="~amd64" emerge gnumeric
```

Cela aura pour effet d'installer la version de test de gnumeric. Il est possible d'indiquer à Portage d'utiliser les versions de test pour certains paquets tout en restant dans la branche stable. Pour cela, ajoutez le nom du paquet dont vous voulez la version instable et sa catégorie dans le fichier `/etc/portage/package.keywords`. Par exemple, pour utiliser la version instable de gnumeric, ajoutez :

```
app-office/gnumeric ~amd64
```

Afin d'installer un paquet de la meilleure manière possible, commencer par vérifier quels sont les paquets nécessaires à l'installation :

```
# emerge -pv mozilla
These are the packages that I would merge, in order:
Calculating dependencies ...done!
[ebuild N ] net-www/mozilla-1.5-r1 +java +crypt -ipv6 -gtk2 +ssl +ldap
+gnome -debug +mozcalendar -mozaccess -mozxmlterm -moznoirc -moznomail
-moznocompose -moznoxf
```

Vous pouvez ensuite ajouter les mots-clés nécessaires à la variable USE pour une configuration optimale du paquet choisi.

12 Mise à jour de /etc

L'outil `etc-update` vous aide à intégrer les fichiers `._cfg0000_<nom>`. Il permet de comparer et de copier les nouveaux fichiers que Portage a installés sans modifier les fichiers qui existaient déjà. Il peut aussi intégrer des modifications simples qui n'ont aucune incidence sur votre système. Les fichiers `._cfg0000_<nom>` sont créés par Portage quand un nouveau fichier devrait en remplacer un autre dans un répertoire protégé par la variable `CONFIG_PROTECT`. Utiliser `etc-update` est plutôt trivial :

```
# etc-update
```

Après avoir intégré les modifications triviales, le programme affiche une liste des fichiers protégés qui n'ont pas été remplacés et pour lesquels une mise à jour est peut-être souhaitable. Les choix suivants sont indiqués au bas de la liste :

```
Exemple de code 2 : Menu etc-update
Please select a file to edit by entering the corresponding number.
(-1 to exit) (-3 to auto merge all remaining files)
(-5 to auto-merge AND not use 'mv -i'):
```

Si vous entrez -1, `etc-update` quitte et ne change rien aux fichiers qui restent dans la liste. Si vous entrez -3 ou -5, tous les fichiers seront remplacés par les nouvelles versions. Il est donc très important de sélectionner les fichiers qui ne doivent pas être remplacés automatiquement avant de choisir cette option. Il suffit d'entrer le numéro du fichier dans la liste.

13 La commande rc-update

Gentoo construit un arbre de dépendances pour déterminer l'ordre d'exécution des services. Cela est loin d'être trivial et nous avons donc créé des outils qui facilitent l'administration des niveaux d'exécution et des scripts d'initialisation.

La commande `rc-update` permet d'ajouter ou d'enlever un script d'un niveau d'exécution. Cette commande utilise automatiquement le script `depscan.sh` qui reconstruit l'arbre des dépendances.

Vous avez déjà ajouté des scripts d'initialisation au niveau d'exécution «default» pendant l'installation de Gentoo. Vous ignoriez alors la signification de «default», mais maintenant, vous la connaissez. Le script `rc-update` a besoin d'un second argument qui spécifie l'action à effectuer : `add`, `del` ou `show` pour respectivement ajouter, supprimer ou afficher.

Pour ajouter ou supprimer un service, ajoutez simplement `add` ou `del` à la commande `rc-update` et spécifiez ensuite le nom du script d'initialisation et le niveau d'exécution. Par exemple :

```
# rc-update del postfix default
```

La commande `rc-update show` affiche la liste des scripts d'initialisation disponibles et les niveaux d'exécution dans lesquels ils ont été ajoutés :

```
# rc-update show
```

14 Mettre à jour votre noyau

La mise à jour des sources d'un noyau se fait de la même façon que la mise à jour de n'importe quel paquet logiciel : avec le programme `emerge`. Vous déciderez probablement de mettre votre noyau à jour lorsque ses sources apparaîtront dans la liste des mises à jour «world». Par exemple, :

```
# emerge -Dup world
Calculating dependencies ...done!
[ebuild NS ] sys-kernel/gentoo-sources-2.6.9-r2 [2.6.8-r5]
```

Note : L'indication NS dans le message ci-dessus signifie que le nouveau noyau sera installé dans un nouveau SLOT, c'est-à-dire que l'ancien noyau sera laissé tel quel jusqu'à ce que vous le supprimiez explicitement.

Vous pouvez alors aller de l'avant et installer la mise à jour :

```
# emerge -u gentoo-sources
```

Les sources du noyau seront installées dans un sous-répertoire de `/usr/src`. Dans l'exemple ci-dessus, les sources seraient installées dans `/usr/src/linux-2.6.9-gentoo-r2`.

Il est nécessaire d'avoir un lien symbolique nommé `/usr/src/linux` pointant vers les sources du noyau que vous utilisez. Portage peut modifier ce lien automatiquement lorsque vous installez un nouveau noyau. Pour cela, ajoutez le mot-clé `symlink` à votre variable `USE` dans le fichier `/etc/make.conf`.

```
USE="symlink x86 3dnow 3dnowex X aac aalib adns alsa apache2"
```

Si vous tenez vraiment à le modifier vous-même, l'exemple suivant vous montre comment faire pointer le lien vers `linux-2.6.9-gentoo-r2` :

```
# cd /usr/src
# ln -sf linux-2.6.9-gentoo-r2 linux
```

Générer le noyau Vous pouvez aussi utiliser des paramètres supplémentaires pour activer certaines fonctionnalités de Genkernel. Par exemple, si vous souhaitez préciser quelques options du noyau avec `menuconfig` et si vous voulez que Genkernel mette à jour la configuration de votre chargeur de démarrage GRUB automatiquement, exécutez Genkernel avec la commande suivante :

```
# genkernel --menuconfig --bootloader=grub all
```

Supprimer les anciens noyaux Il est fort possible que vous soyez tout à fait satisfait de votre nouveau noyau et que vous n'ayez pas besoin de conserver les anciennes versions sur votre système. Pour supprimer facilement toutes les sources sauf la plus récente pour un noyau donné, utilisez l'option prune de la commande emerge. Poursuivons l'exemple précédent utilisant gentoo-sources :

```
# emerge -P gentoo-sources
```

Vous pouvez également supprimer les modules utilisés par les anciens noyaux sans problème. Pour ce faire, il vous suffit de supprimer les sous-répertoires appropriés dans `/lib/modules/` (ceux qui correspondent aux versions des noyaux dont vous avez supprimé les sources). Faites attention à ne pas supprimer des modules appartenant à des noyaux que vous utilisez encore ! Finalement, vous pouvez monter votre partition `/boot` et supprimer les fichiers bzImage des noyaux que vous avez supprimés. Vous devriez aussi configurer votre chargeur de démarrage pour retirer les entrées qui font référence aux noyaux supprimés.

Incompatibilité des Cartes-Mères Asustech Possédant une carte asus K8N4-E Deluxe et un Athlon 64 3000+ en socket 754, lors de la mise à jour au noyau supérieure au `linux-2.6.12-gentoo-r10`, soit le noyau ne bootait pas du tout, soit il bootait et aucun des périphériques PCI n'était reconnu. J'obtenais des messages d'erreurs du type :

```
biboo ~ # more dmesg-kernel-2.6.15
Bootdata ok (command line is root=/dev/ram0 init=/linuxrc ramdisk=8192 real_root=/dev/hd
Linux version 2.6.15-gentoo-r5 (root@biboo) (version gcc 3.4.4 (Gentoo 3.4.4-r1, ssp-3.4
[snip...]
[snip...]
ACPI: PCI interrupt for device 0000:00:02.1 disabled
ehci_hcd 0000:00:02.1: init 0000:00:02.1 fail, -16
ehci_hcd: probe of 0000:00:02.1 failed with error -16
[snip...]
[snip...]
ACPI: PCI interrupt for device 0000:00:02.0 disabled
ohci_hcd 0000:00:02.0: init 0000:00:02.0 fail, -16
ohci_hcd: probe of 0000:00:02.0 failed with error -16
[snip...]
[snip...]
ohci1394: fw-host0: Unexpected PCI resource length of 0!
ohci1394: MMIO resource (0x0 - 0x800) unavailable
ohci1394: probe of 0000:05:0b.0 failed with error -12
[snip...]
[snip...]
ACPI: PCI Interrupt 0000:05:0a.0[A] -> Link [LNK4] -> GSI 3 (level, low) -> IRQ
3
ACPI: PCI interrupt for device 0000:05:0a.0 disabled
sata_sil: probe of 0000:05:0a.0 failed with error -12
[snip...]
[snip...]
forcedeth: Couldn't find register window for device 0000:00:0a.0.
ACPI: PCI interrupt for device 0000:00:0a.0 disabled
forcedeth: probe of 0000:00:0a.0 failed with error -22
[snip...]
[snip...]
```

```
ACPI: PCI Interrupt 0000:05:07.0[A] -> Link [LNK2] -> GSI 5 (level, low) -> IRQ
5
btv0: can't request iomem (0x0).
btv: probe of 0000:05:07.0 failed with error -16
[snip...]
[snip...]
```

Du coup, pas de carte réseau, pas de son, par de mode graphique... En cherchant un peu je suis tombé sur ce message <http://forums.gentoo.org/viewtopic-t-418790-highlight-ehcihcd+0000+00+02+1+init+0000+00+02+1+fail+16.html> qui décrivait les mêmes symptômes K8N4-E Deluxe : almost everything fails with 2.6.14 [Solved]. Des infos aussi ici <http://forums.gentoo.org/viewtopic-t-417237.html>.

Et la réponse était la suivante :

Remove support for mmconfig config space access in the kernel configuration, it's broken
Code:

```
Bus options (PCI etc.) --->
[ ] Support mmconfig PCI config space access
```

You can also boot with "pci=nommconf" added to the kernel parameters, it should disable

Il s'agit d'un bug entre les carte Asus et le noyau lorsque l'on active mmconfig dans les options du bus PCI.

J'ai donc recompilé mon noyau en enlevant cette option est tout est rentré dans l'ordre.

Il m'a bien sur fallu réinstaller les pilotes alsa pour la carte son. Puis réinstaller les pilotes Nvidia pour profiter de l'accélération graphique.

```
biboo ~ # emerge alsa-driver
biboo ~ # ./NVIDIA-Linux-x86_64-1.0-8178-pkg2.run
```

Lors du redémarrage, la carte son ne fonctionnait plus avec le message d'erreur suivant :

```
snd: Unknown parameter 'device_mode'
snd_timer: Unknown symbol snd_info_register
snd_timer: Unknown symbol snd_info_create_module_entry
snd_timer: Unknown symbol snd_info_free_entry
snd_timer: Unknown symbol snd_verbose_printk
...
```

d'après l'article suivant <http://dev.gentoo.org/~dsd/genpatches/issues-current.htm>, cela vient de la configuration d'alsa.

```
"Unknown parameter device_mode" error
ALSA broken; lots of "Unknown symbol" messages appear on load
```

Solution. ALSA has recently dropped the device_mode parameter. If you still have this pa

```
snd: Unknown parameter 'device_mode'
```

Immediately after this message, you'll see *many* messages like these:

```
snd_timer: Unknown symbol snd_info_register
```

```
snd_timer: Unknown symbol snd_info_create_module_entry
snd_timer: Unknown symbol snd_info_free_entry
snd_timer: Unknown symbol snd_verbose_printk
```

Note that the "Unknown symbol" messages aren't actually correct - the real problem is th

This is usually caused by a file in /etc/modules.d supplying a device_mode parameter.

To find which file is supplying this, you can do:

```
# grep device_mode /etc/modules.d/*
```

Usually, it is a file called alsa containing the line:

```
options snd device_mode=0666
```

Remove this line, save the file, run modules-update, and the problem should be solved.

j'ai donc commenté la ligne dans le fichier concerné :

```
biboo ~ # grep device_mode /etc/modules.d/*
/etc/modules.d/alsa:#options snd device_mode=0666
biboo ~ # modules-update
biboo ~ # biboo ~ # /etc/init.d/alsasound restart
```

et tout est rentré dans l'ordre.

Carte firewire en eth0 et carte réseau en eth1 Avec ce nouveau noyau (2.6.15-gentoo-r5), le réseau ne démarrait plus. En effet la carte réseau initialement détectée en **eth0** avec le noyau 2.6.12, est détectée en **eth1** avec le noyau 2.6.15.

```
biboo ~ # dmesg | grep eth
eth1394: $Rev: 1312 $ Ben Collins <bcollins@debian.org>
eth1394: eth0: IEEE-1394 IPv4 over 1394 Ethernet (fw-host0)
forcedeth.c: Reverse Engineered nForce ethernet driver. Version 0.48.
eth1: forcedeth.c: subsystem: 01043:8141 bound to 0000:00:0a.0
```

Du coup, impossible d'accéder au réseau. J'ai donc "installé" la seconde carte réseau et désactivé la première. Voici la marche à suivre :

```

biboo ~ # more /etc/conf.d/net*
::::::::::::
/etc/conf.d/net
::::::::::::
# This blank configuration will automatically use DHCP for any net.*
# scripts in /etc/init.d. To create a more complete configuration,
# please review /etc/conf.d/net.example and save your configuration
# in /etc/conf.d/net (this file :!)).
config_eth0=("dhcp")
dhcp_eth0="nodns nontp nonis"

config_eth1=("dhcp")
dhcp_eth1="nodns nontp nonis"
biboo ~ # ln -s /etc/init.d/net.eth0 /etc/init.d/net.eth1
biboo ~ # ls -al /etc/init.d/net*
lrwxrwxrwx 1 root root      6 fév 17 18:15 /etc/init.d/net.eth0 -> net.lo
lrwxrwxrwx 1 root root      8 fév 17 23:08 /etc/init.d/net.eth1 -> net.eth0
-rwxr-xr-x 1 root root 24324 fév 17 18:15 /etc/init.d/net.lo
-rwxr-xr-x 1 root root 3055 fév 17 18:15 /etc/init.d/netmount
biboo ~ #
biboo ~ # rc-update add net.eth1 default
* net.eth1 added to runlevel default
* rc-update complete.
biboo ~ #
biboo ~ # rc-update delete net.eth0
* net.eth0 removed from the following runlevels: default
* rc-update complete.
biboo ~ # rc-update show | grep eth
          net.eth0 |
          net.eth1 |      default
biboo ~ #

```

Tout est alors rentré dans l'ordre.

Installation WIFI Tout d'abord un petit `lspci | less` afin de trouver le chipset qui s'occupe de la connexion sans fil. Pour ma part dans mon Dell latitude D800 il s'agit d'une Dell TrueMobile 1300 WLAN Mini-PCI Card, qui est en fait une carte Broadcom avec un chip `bcmwl5`.

J'ai tout d'abord essayé le couple `ndiswrapper` et les pilotes windows du constructeurs sans succès. Un problème au chargement du module empêche la carte de fonctionner.

Je me suis alors tourné vers le module `bcm43xx` inclus dans les derniers noyaux. Celui-ci semble bien fonctionner.

J'ai recompilé pour l'occasion un 2.6.18 en vérifiant bien que ce module était dans la liste à générer.

J'ai ensuite ajouté la ligne :

```
bcm43xx
```

dans le fichier `/etc/modules.autoload/kernel2.6` pour un chargement automatique au démarrage.

Afin de tester sans redémarrage, un petit `modprobe bcm43xx` pour installer le module en mémoire puis

```
iwconfig
```

pour vérifier la présence de la carte wifi. Si à tout hasard il n'y avait rien, vérifier la présence de `net.wlan0` dans `/etc/init.d`. S'il n'y a rien, fait un lien symbolique sur `net.lo`

```
ln -s /etc/init.d/net.lo /etc/init.d/net.wlan0
```

Ensuite vérifier le fichier `/etc/conf.d/net`

```
config_wlan0="dhcp"  
dhcp_wlan0="nontp nonis"
```

Pour ma part j'ai du enlever l'option `nodns` dans `dhcp_wlan0` afin que le fichier `resolv.conf` soit modifié par le serveur dhcp distant.

Normalement, c'est OK et bon surf.

15 Installation d'une station de travail

15.1 Installation de Xorg

Assez bavardé, installons Xorg ! Lancez simplement la commande `emerge xorg-x11`. L'installation prend un certain temps, profitez-en pour manger un morceau.

Avant de faire :

```
# emerge xorg-x11
```

comme l'indique la méthode officielle, taper donc la commande suivante afin de voir quelle sont les mots clés nécessaires pour la compilation de `xorg`.

```
# emerge -pv xorg-x11
```

En effet si vous omettez le mot-clé `opengl` vous aurez du mal à faire fonctionner l'accélération graphique. En effet le module `GLcore` n'est pas généré et le module `glx` ne peut alors se charger correctement.

Ajouter donc le mot-clé `opengl` dans la variable `USE` du fichier `make.conf`.

Après l'installation, vous devez réinitialiser votre environnement avant de poursuivre. Lancez les commandes `env-update` et `source /etc/profile`. Ceci n'a pas d'effets négatifs sur votre système.

```
# env-update  
# source /etc/profile
```

Xorg contient un outil de configuration appelé `xorgconfig`. Celui-ci vous pose une série de questions (carte graphique, type de clavier, etc.) et construit un fichier de configuration `xorg.conf` en fonction de vos réponses.

```
# xorgconfig
```

`xorg` fournit un autre outil de configuration appelé `xorgcfg` qui essaie de lancer `Xorg -configure` et lance ensuite le serveur X pour permettre des ajustements supplémentaires.

```
# xorgcfg  
(Si X plante ou si la configuration échoue, essayez plutôt)  
# xorgcfg -textmode
```

Recopions le fichier `xorg.conf.new` vers `/etc/X11/xorg.conf` pour éviter de devoir lancer `Xorg -config` ; taper `X` ou `startx` est bien plus facile.

```
# cp /root/xorg.conf.new /etc/X11/xorg.conf
```

Voici un exemple de fichier de configuration `xorg.conf` :

```
Section "Module"
    Load      "glx"
    Load      "dbe"
    SubSection "extmod"
        Option "omit xfree86-dga"
    EndSubSection
    Load      "type1"
    Load      "freetype"
EndSection

Section "Files"
    RgbPath    "/usr/X11R6/lib/X11/rgb"
    FontPath   "/usr/share/fonts/TTF/"
    FontPath   "/usr/share/fonts/ttf-bitstream-vera/"
    FontPath   "/usr/share/fonts/terminus/"
    FontPath   "/usr/share/fonts/corefonts/"
    FontPath   "/usr/share/fonts/freefont/"
    FontPath   "/usr/share/fonts/artwiz/"
    FontPath   "/usr/share/fonts/local/"
    FontPath   "/usr/share/fonts/75dpi:unscaled"
    FontPath   "/usr/share/fonts/100dpi:unscaled"
    FontPath   "/usr/share/fonts/Speedo/"
    FontPath   "/usr/share/fonts/CID/"
    FontPath   "/usr/share/fonts/encodings/"
    FontPath   "/usr/share/fonts/util/"
    FontPath   "/usr/share/fonts/misc/"
    FontPath   "/usr/share/fonts/Type1/"
    FontPath   "/usr/share/fonts/sharefont/"
    FontPath   "/usr/share/fonts/75dpi/"
    FontPath   "/usr/share/fonts/100dpi/"
    FontPath   "/usr/share/fonts/lfp-fix/"
    FontPath   "/usr/share/fonts/unifont/"
    FontPath   "/usr/share/fonts/local/"
EndSection

Section "ServerFlags"
EndSection
```

```
Section "InputDevice"
    Identifier "Keyboard1"
    Driver     "kbd"
    Option     "XkbRules"    "xorg"
    Option     "XkbModel"    "pc105"
    Option     "XkbLayout"   "fr"
EndSection

Section "InputDevice"
    Identifier "Mouse1"
    Driver     "mouse"
    Option     "Protocol"     "IMPS/2"
    Option     "ZAxisMapping" "6 7"
    Option     "ButtonNumber" "7"
    Option     "Buttons"      "7"
    Option     "Device"       "/dev/mouse"
EndSection

Section "Monitor"
    Identifier "My Monitor"
    HorizSync  31.5 - 57.0
    VertRefresh 50-90
EndSection

Section "Device"
    Identifier "Standard VGA"
    VendorName "Unknown"
    BoardName  "Unknown"
    Driver     "vga"
EndSection
```

```

Section "Device"
    Identifier   "Chaintech nVidia GeForce 6800GT"
    Driver       "nvidia"
    VideoRam     131072
EndSection

Section "Screen"
    Identifier   "Screen 1"
    Device       "Chaintech nVidia GeForce 6800GT"
    Monitor      "My Monitor"
    DefaultDepth 24
    Subsection   "Display"
        Depth     8
        Modes      "1280x1024" "1024x768" "800x600" "640x480"
        ViewPort   0 0
    EndSubsection
    Subsection   "Display"
        Depth     16
        Modes      "1280x1024" "1024x768" "800x600" "640x480"
        ViewPort   0 0
    EndSubsection
    Subsection   "Display"
        Depth     24
        Modes      "1280x1024" "1024x768" "800x600" "640x480"
        ViewPort   0 0
    EndSubsection
EndSection

Section "ServerLayout"
    Identifier   "Simple Layout"
    Screen       "Screen 1"
    InputDevice  "Mouse1" "CorePointer"
    InputDevice  "Keyboard1" "CoreKeyboard"
EndSection

```

Ensuite, essayez de démarrer le serveur X avec la commande **startx**. Celle-ci exécute un script qui lance une session X, c'est-à-dire qu'elle démarre le serveur X et des applications graphiques. Les applications sont démarrées selon la logique suivante :

- ▷ Si un fichier **.xinitrc** existe dans le répertoire **home**, les commandes qu'il contient seront exécutées.
- ▷ Sinon, la variable **XSESSION** indique le nom de la session à exécuter. Cette session correspond à un fichier dans le répertoire **/etc/X11/Sessions/**. Vous pouvez définir la variable **XSESSION** dans le fichier **/etc/rc.conf** pour en faire le défaut pour tous les utilisateurs.
- ▷ En cas d'échec, le défaut est de démarrer un simple gestionnaire de fenêtres, généralement **twm**.

```
# startx
```

Si un affreux gestionnaire de fenêtres s'affiche, c'est **twm**. Pour quitter, tapez **exit** ou **Ctrl-D** dans chaque terminal X. Vous pouvez aussi tuer la session avec la combinaison de touches **Ctrl-Alt-Backspace**. Cette combinaison vous permet de quitter X brusquement et n'est pas recommandée, sauf cas de force majeure.

15.2 Installation des pilotes Nvidia

15.2.1 Installation des pilotes Gentoo

Afin de bénéficier de l'accélération graphique des toutes nouvelles cartes 3D il est possibles d'installer un pilote différent que le pilote générique NV des cartes à base de GPU Nvidia.

La compagnie nVidia propose ses propres pilotes graphiques. Ceux-ci offrent de bonnes performances et une accélération 3D complète. Les distributions des pilotes comprennent deux composants : `nvidia-kernel` et `nvidia-glx`. `nvidia-kernel` est un pilote pour le noyau Linux qui prend en charge la communication de bas niveau avec votre matériel vidéo. Il s'agit simplement d'un module du noyau, nommé `nvidia`, installé en fonction de votre arbre du code source du noyau, et qui doit être chargé à chaque fois que vous souhaitez utiliser les pilotes nVidia.

En plus du pilote du noyau, vous devez installer le composant GLX pour X11 (`nvidiaglx`). Celui-ci est utilisé par X pour le rendu des images qui se fait par le biais du pilote `nvidia-kernel`, ce dernier servant d'interface vers le matériel.

Tel que mentionné ci-dessus, les pilotes nVidia sont installés et exécutés en fonction de votre noyau courant. Puisque le pilote consiste en un module, votre noyau doit, logiquement, supporter le chargement de modules. Si vous utilisez `genkernel` pour la configuration de votre noyau, les paramètres appropriés ont déjà été choisis pour vous.

Autrement, vérifiez vous-même votre configuration pour vous assurer que les options appropriées sont sélectionnées :

```
Loadable module support --->
[*] Enable loadable module support
```

Vous devez aussi activer l'option Memory Type Range Register :

```
Processor and Features --->
[*] MTRR (Memory Type Range Register) support
```

Les bibliothèques et les modules de nVidia sont disponibles dans deux paquets séparés : `nvidia-glx` et `nvidia-kernel`. Le premier contient les bibliothèques GLX pour X11 tandis que le second contient les modules du noyau.

L'`ebuild nvidia-kernel` trouve automatiquement la version de votre noyau grâce au lien symbolique `/usr/src/linux`. Assurez-vous que ce lien pointe vers les bonnes sources et que le noyau a été configuré correctement (le noyau que vous utilisez couramment).

Il est maintenant temps d'installer `nvidia-kernel` et `nvidia-glx`. Puisque `nvidia-glx` dépend de `nvidia-kernel`, installer le premier est suffisant :

```
# emerge nvidia-glx
```

Important : Chaque fois que vous compilez un nouveau noyau ou que vous recompilez votre noyau courant, vous devez exécuter `emerge nvidia-kernel` pour réinstaller les modules nVidia. `nvidia-glx` n'est pas affecté par les changements du noyau ; il n'est même pas nécessaire de le réinstaller lorsque vous mettez à jour (ou recompilez) X.

Si comme pour moi les `ebuilds` de la branche stable ne fonctionnent pas vous pouvez tester les `ebuilds` de la branche `testing` par la commande :

```
# ACCEPT_KEYWORDS="~amd64" emerge nvidia-glx
```

ou encore mettre dans le fichier `/etc/portage/package.keywords` les lignes suivantes :

```
media-video/nvidia-kernel ~amd64
media-video/nvidia-glx ~amd64
```

Lorsque l'installation est finie, lancez `modprobe nvidia` pour charger le module du noyau en mémoire.

```
# modprobe nvidia
```

Pour éviter d'avoir à charger le module à chaque démarrage, vous souhaitez sans doute automatiser ce processus. Pour ce faire, éditez le fichier `/etc/modules.autoload.d/kernel-2.6` (ou `kernel-2.4` en fonction de la version de votre noyau) et ajoutez-y `nvidia`. N'oubliez pas d'exécuter `modules-update` par après :

```
# modules-update
```

Et voici ce que contient le fichier `/etc/modules.d/nvidia`

```
biboo etc # more ./modules.d/nvidia
# Nvidia drivers support
alias char-major-195 nvidia
alias /dev/nvidiactl char-major-195

# To tweak the driver the following options can be used, note that
# you should be careful, as it could cause instability!! For more
# option see /usr/share/doc/nvidia-kernel-1.0.6629-r4/README.gz
#
#   To enable Side Band Addressing: NVreg_EnableAGPSBA=1
#
#   To enable Fast Writes: NVreg_EnableAGPFW=1
#
# To enable both for instance, uncomment following line:
#
#options nvidia NVreg_EnableAGPSBA=1 NVreg_EnableAGPFW=1

biboo etc #
```

Une fois que les pilotes sont installés, vous devez configurer votre serveur X (Xorg ou XFree86) pour qu'il utilise le pilote `nvidia` au lieu du pilote `nv` par défaut. Ouvrez `/etc/X11/xorg.conf` (ou `/etc/X11/XF86Config` si vous utilisez encore l'ancien emplacement pour le fichier de configuration) avec votre éditeur favori (nano ou vim par exemple) et allez à la section **Device**. Dans cette section, modifiez la ligne **Driver** :

```
Section "Device"
Identifier "nVidia Inc. GeForce6200"
Driver "nvidia"
VideoRam 131072
EndSection
```

Ensuite, allez à la section **Module** et assurez vous que le module `glx` se charge tandis que le module `dri` ne se charge pas :

```
Section "Module"
(...)
# Load "dri"
Load "glx"
(...)
EndSection
```

Puis, dans la section **Screen**, vérifiez que la directive `DefaultDepth` indique 16 ou 24, ou qu'il n'y a pas de sous-section **Display** avec des directives `Depth` qui valent autre chose que 16 ou 24. Si vous spécifiez d'autres profondeurs de couleurs que 16 ou 24, les extensions `nvidia-glx` ne fonctionneront pas.

```
Section "Screen"
(...)
DefaultDepth 16
Subsection "Display"
(...)
EndSection
```

Lancez `opengl-update` afin que le serveur X utilise les bibliothèques GLX de nVidia :

```
# opengl-update nvidia
```

Ajouter des utilisateurs dans le groupe `video`. Afin que vos utilisateurs puissent accéder aux fichiers de périphérique nVidia, il faut les ajouter au groupe `video` :

```
# gpasswd -a nom_utilisateur video
```

Cela n'est pas forcément obligatoire si vous n'utilisez pas `udev` mais cela ne vous fera pas de mal. De plus, votre système sera préparé pour l'avenir :).

Pour tester votre carte nVidia, démarrez X et lancez la commande `glxinfo | grep direct`. Cela devrait vous répondre que le rendu direct est activé :

```
$ glxinfo | grep direct
direct rendering: Yes
```

Pour obtenir votre taux de FPS (images par seconde), lancez `glxgears`.

Certains paquets comme `mplayer` ou `xine-lib` ont une option `USE locale nvidia` qui permet d'activer le support `XvMCNVIDIA`, utile notamment pour visionner des films en haute résolution. Vous pouvez définir cette option dans la variable `USE` de votre fichier `/etc/make.conf` ou bien l'ajouter pour les paquets `media-video/mplayer` et/ou `medialibs/xine-lib` dans le fichier `/etc/portage/package.use`. Ensuite, exécutez `emerge -uD --newuse world` pour recompiler les applications qui utilisent cette option.

Depuis la version 1.0.6106 des pilotes nVidia, vous pouvez disposer d'un outil configuration qui vous permet de modifier les réglages graphiques sans redémarrer le serveur X. Cet outil est disponible dans Portage sous le nom de `media-video/nvidia-settings`.

Faire fonctionner la carte en 2D sur des machines avec 4 Go de mémoire ou plus Si vous avez des problèmes avec l'accélération 2D, il est probable que l'origine soit un problème d'écriture avec `MTRR`. Veuillez vérifier le contenu du fichier `/proc/mtrr` :

```
# cat /proc/mtrr
```

Chaque ligne doit contenir soit «write-back», soit «write-combining». Si une ligne avec «uncachable» apparaît, vous devez modifier une option de votre BIOS pour résoudre le problème.

Redémarrez votre machine et cherchez l'option `MTRR`, en général, dans un menu «CPU Settings». Changez cette option de «continuous» en «discrete» et redémarrez Linux. Vous ne devriez plus voir «uncachable» dans `/proc/mtrr` et l'accélération 2D devrait mieux fonctionner.

15.2.2 Configuration avancée

Afin d'utiliser toutes les possibilités de votre carte graphique (sortie TV, Bi-écran,...). Nous verrons un peu plus loin comment configurer `xorg.conf` afin d'avoir le bi-écran sur notre carte Nvidia Geforce 6200.

Le paquet logiciel du pilote nVidia contient une documentation exhaustive. Elle est installée dans le répertoire `/usr/share/doc` et peut être consultée avec la commande suivante :

```
# less /usr/share/doc/nvidia-glx-*/README.txt.gz
```

Paramètres du module du noyau

Le module du noyau nvidia accepte un éventail de paramètres (options) qui permettent d'ajuster le comportement du pilote. La plupart sont mentionnés dans la documentation. Pour ajouter ou modifier ces paramètres, éditez le fichier `/etc/modules.d/nvidia`. Souvenez-vous d'exécuter `modules-update` après ces modifications et gardez à l'esprit que vous devrez recharger le module nvidia avant que ces modifications ne soient prises en compte.

```
# nano -w /etc/modules.d/nvidia
(Mettez à jour les fichiers des modules.)
# modules-update
Déchargez le module nvidia...)
# modprobe -r nvidia
(... et rechargez-le.)
# modprobe nvidia
```

La couche logicielle GLX offre aussi une pléthore d'options de configuration. Celles-ci contrôlent la sortie TV, l'utilisation simultanée de deux écrans, la détection de la fréquence des moniteurs, etc. Ces options sont elles aussi couvertes dans la documentation.

Si vous voulez utiliser certaines de ces options, vous devez les indiquer dans la section Device appropriée de votre fichier de configuration de X (habituellement `/etc/X11/xorg.conf`). Par exemple, supposons que vous souhaitez désactiver le logo de démarrage :

```
Section "Device"
Identifier "nVidia Inc. GeForce6200"
Driver "nvidia"
Option "NoLogo" "true"
VideoRam 131072
EndSection
```

15.2.3 Installation des Pilotes Nvidia

Dans mon cas, avec une Geforce6200 Leadtek, en suivant toutes les instructions précédente, au lancement de `startx`, j'obtenais une beau plantage avec écran noir.

J'ai donc commencé par désinstaller les pilotes nvidia de la distribution gentoo

```
# emerge unmerge nvidia-kernel
# emerge unmerge nvidia-glx
```

ensuite on enlève le module `nvidia` de la mémoire et on met à jour la liste des modules.

```
# rmmod nvidia
# modules-update
```

Ensuite il faut télécharger les pilotes propriétaires Nvidia à l'adresse suivante http://www.nvidia.com/object/linux_display_amd64_1.0-8178.html. Si ce lien ne fonctionne plus, c'est que la version a changé et rendez-vous sur la page <http://www.nvidia.com/content/drivers/drivers.asp> et sélectionnez le pilote désiré.

Il suffit alors de changer les droits du fichier `NVIDIA-Linux-x86_64-1.0-<version>-pkg2.run` et de l'exécuter :

```
# chmod 755 NVIDIA-Linux-x86_64-1.0-<version>-pkg2.run
# ./NVIDIA-Linux-x86_64-1.0-<version>-pkg2.run
```

Suivez les indications proposées par l'installateur. Normalement tout se passe bien.

Bien sur dans le cas d'un changement de noyau, vous serez obligé de réitérer l'installation des pilotes Nvidia.

Normalement, notre fichier `xorg.conf` est prêt et on peut lancer `startx`. Dans mon cas tout a fonctionner et j'ai eu droit à la très belle interface de `twm`.

Si vous ne désirez pas le logo Nvidia au démarrage utilisez l'option `nologo` dans la section `Device`

```
Option "NoLogo" "true"
```

15.2.3.1 Remarque : Si comme moi, votre carte graphique est équipée d'un connecteur DVI et d'un connecteur VGA, utilisez de préférence le connecteur DVI (auquel vous ajoutez ou non le convertisseur VGA), car au départ, j'avais branché l'écran sur la VGA et la qualité de l'image en mode graphique était assez médiocre (très floue dans les hautes résolutions). En passant par la suite sur la connecteur DVI (avec le convertisseur), la qualité s'est nettement améliorée.

15.2.4 Configuration Bi-ecran

Voici un exemple de configuration bi-écran avec une carte Nvidia Geforce 6200 `/etc/X11/xorg.conf`. J'ai utilisé deux écrans identiques. Il faut remarqué que la définition n'est pas la même sur les deux écrans. L'écran branché sur la prose SVI (avec adaptateur) possède une image très nette. Par contre celui branché sur la sortie VGA de la carte, possède une image floue. On dirait que la sortie VGA est un clone amélioré de la sortie TV.

La partie la plus importante du fichier est la section `Device`

```
Section "Device"
#nv"
# Insert Clocks lines here if appropriate
Identifier      "Nvidia Geforce 6200"
Driver          "nvidia"
#option twinview
Option          "TwinView"
Option          "NoTwinViewXineramaInfo"    "true"
Option          "SecondMonitorHorizSync"    "27.0-96.0"
Option          "SecondMonitorVertRefresh"  "50.0-160.0"
Option          "TwinViewOrientation"       "RightOf"
Option          "MetaModes"                  "1280x1024,1280x1024"
Option          "ConnectedMonitor"          "crt,crt"
#fin option twinview
EndSection
```

Toute cette section contient la configuration du bi-écran. La ligne principale est `ption "TwinView"` qui active le bi-ecran sur ma Geforce. Il faut bien sur avoir le pilote Nvidia chargé. Les deux lignes suivantes :

```
Option          "SecondMonitorHorizSync"    "27.0-96.0"
Option          "SecondMonitorVertRefresh"  "50.0-160.0"
```

Permettent de définir les caractéristiques du moniteur supplémentaire. Ensuite on indique où se trouve le moniteur secondaire par rapport au moniteur principal

```
Option          "TwinViewOrientation"       "RightOf"
```

puis on indique les résolution de chaque moniteur et enfin on indique le mode de connexion (`crt` ou `dvi`).

```
Option          "MetaModes"                  "1280x1024,1280x1024"
Option          "ConnectedMonitor"          "crt,crt"
```

Voici le fichier complet pour ma configuration.

```
# nvidia-xconfig: X configuration file generated by nvidia-xconfig
# nvidia-xconfig:  version 1.0  (buildmeister@builder26)  Wed Dec 14 17:16:10 PST 2005

# File generated by xorgconfig.
#
# Copyright 2004 The X.Org Foundation
#
# Permission is hereby granted, free of charge, to any person obtaining a
# copy of this software and associated documentation files (the "Software"),
# to deal in the Software without restriction, including without limitation
# the rights to use, copy, modify, merge, publish, distribute, sublicense,
# and/or sell copies of the Software, and to permit persons to whom the
# Software is furnished to do so, subject to the following conditions:
#
# The above copyright notice and this permission notice shall be included in
# all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
# IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
# FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.  IN NO EVENT SHALL
# The X.Org Foundation BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
# WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF
# OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
# SOFTWARE.
#
# Except as contained in this notice, the name of The X.Org Foundation shall
# not be used in advertising or otherwise to promote the sale, use or other
# dealings in this Software without prior written authorization from
# The X.Org Foundation.
#
# *****
# Refer to the xorg.conf(5x) man page for details about the format of
# this file.
# *****
# *****
# Module section -- this section is used to specify
# which dynamically loadable modules to load.
# *****
#
# *****
# Files section.  This allows default font and rgb paths to be set
# *****
# *****
# Server flags section.
# *****
# *****
# Input devices
# *****
# *****
# Core keyboard's InputDevice section
# *****
```

```

# *****
# Core Pointer's InputDevice section
# *****
# *****
# Other input device sections
# this is optional and is required only if you
# are using extended input devices. This is for example only. Refer
# to the xorg.conf man page for a description of the options.
# *****
#
# Section "InputDevice"
#   Identifier "Mouse2"
#   Driver      "mouse"
#   Option      "Protocol"          "MouseMan"
#   Option      "Device"            "/dev/mouse2"
# EndSection
#
# Section "InputDevice"
#   Identifier "spaceball"
#   Driver      "magellan"
#   Option      "Device"            "/dev/cua0"
# EndSection
#
# Section "InputDevice"
#   Identifier "spaceball2"
#   Driver      "spaceorb"
#   Option      "Device"            "/dev/cua0"
# EndSection
#
# Section "InputDevice"
#   Identifier "touchscreen0"
#   Driver      "microtouch"
#   Option      "Device"            "/dev/ttyS0"
#   Option      "MinX"              "1412"
#   Option      "MaxX"              "15184"
#   Option      "MinY"              "15372"
#   Option      "MaxY"              "1230"
#   Option      "ScreenNumber"      "0"
#   Option      "ReportingMode"     "Scaled"
#   Option      "ButtonNumber"      "1"
#   Option      "SendCoreEvents"
# EndSection
#

```

```

# Section "InputDevice"
#   Identifier "touchscreen1"
#   Driver      "elo2300"
#   Option      "Device"          "/dev/ttyS0"
#   Option      "MinX"            "231"
#   Option      "MaxX"            "3868"
#   Option      "MinY"            "3858"
#   Option      "MaxY"            "272"
#   Option      "ScreenNumber"    "0"
#   Option      "ReportingMode"   "Scaled"
#   Option      "ButtonThreshold"  "17"
#   Option      "ButtonNumber"    "1"
#   Option      "SendCoreEvents"
# EndSection
# *****
# Monitor section
# *****
# Any number of monitor sections may be present
# *****
# Graphics device section
# *****
# Any number of graphics device sections may be present
# Standard VGA Device:
# Device configured by xorgconfig:
# *****
# Screen sections
# *****
# Any number of screen sections may be present.  Each describes
# the configuration of a single screen.  A single specific screen section
# may be specified from the X server command line with the "-screen"
# option.
# *****
# ServerLayout sections.
# *****
# Any number of ServerLayout sections may be present.  Each describes
# the way multiple screens are organised.  A specific ServerLayout
# section may be specified from the X server command line with the
# "-layout" option.  In the absence of this, the first section is used.
# When now ServerLayout section is present, the first Screen section
# is used alone.
# Section "DRI"
#   Mode 0666
# EndSection

```

Section "ServerLayout"

```
# The Identifier line must be present
# Each Screen line specifies a Screen section name, and optionally
# the relative position of other screens. The four names after
# primary screen name are the screens to the top, bottom, left and right
# of the primary screen. In this example, screen 2 is located to the
# right of screen 1.
# Each InputDevice line specifies an InputDevice section name and
# optionally some options to specify the way the device is to be
# used. Those options include "CorePointer", "CoreKeyboard" and
# "SendCoreEvents".
    Identifier      "Simple Layout"
    Screen          "Screen 1" 0 0
    InputDevice     "Mouse1" "CorePointer"
    InputDevice     "Keyboard1" "CoreKeyboard"
EndSection
```

Section "Files"

```
# The location of the RGB database. Note, this is the name of the
# file minus the extension (like ".txt" or ".db"). There is normally
# no need to change the default.
# Multiple FontPath entries are allowed (which are concatenated together),
# as well as specifying multiple comma-separated entries in one FontPath
# command (or a combination of both methods)
#
#
#   FontPath      "/usr/share/fonts/CID/"
#   FontPath      "/usr/share/fonts/Speedo/"
#   FontPath      "/usr/share/fonts/TrueType/"
#   FontPath      "/usr/share/fonts/freefont/"
# The module search path. The default path is shown here.
#   ModulePath    "/usr/X11R6/lib/modules"
    RgbPath        "/usr/X11R6/lib/X11/rgb"
    FontPath        "/usr/share/fonts/misc/"
    FontPath        "/usr/share/fonts/TTF/"
    FontPath        "/usr/share/fonts/Type1/"
    FontPath        "/usr/share/fonts/75dpi/"
    FontPath        "/usr/share/fonts/100dpi/"
    FontPath        "/usr/share/fonts/local/"
EndSection
```

```
Section "Module"

# This loads the DBE extension module.
# initialisation of the XFree86-DGA extension within that module.
# This loads the font modules
# This loads the GLX module
#   Load      "dri"
#       Load      "dbe"          # Double buffer extension
#       Load      "bitmap"
#       Load      "ddc"
#       Load      "extmod"
#       Load      "int10"
#       Load      "record"
#   Load      "speedo"
#       Load      "vbe"
# This loads the miscellaneous extensions module, and disables
#   SubSection  "extmod"
#       Option      "omit xfree86-dga"    # don't initialise the DGA extension
#   EndSubSection
#   Load      "type1"
#   Load      "speedo"
#   Load      "freetype"
#   Load      "xtt"
#       Load      "glx"
# This loads the DRI module
EndSection
```

Section "InputDevice"

```
# For most OSs the protocol can be omitted (it defaults to "Standard").
# When using XQUEUE (only for SVR3 and SVR4, but not Solaris),
# uncomment the following line.
# Option      "Protocol"      "Xqueue"
# Option      "Xleds"         "1 2 3"
# Option "LeftAlt"      "Meta"
# Option "RightAlt"     "ModeShift"
# To customise the XKB settings to suit your keyboard, modify the
# lines below (which are the defaults).  For example, for a non-U.S.
# keyboard, you will probably want to use:
# Option "XkbModel"       "pc105"
# If you have a US Microsoft Natural keyboard, you can use:
# Option "XkbModel"       "microsoft"
#
# Then to change the language, change the Layout setting.
# For example, a german layout can be obtained with:
# Option "XkbLayout"      "de"
# or:
# Option "XkbLayout"      "de"
# Option "XkbVariant"     "nodeadkeys"
#
# If you'd like to switch the positions of your capslock and
# control keys, use:
# Option "XkbOptions"      "ctrl:swapcaps"
# These are the default XKB settings for Xorg
# Option "XkbRules"        "xorg"
# Option "XkbModel"        "pc105"
# Option "XkbLayout"       "us"
# Option "XkbVariant"      ""
# Option "XkbOptions"      ""
# Option "XkbDisable"
Identifier      "Keyboard1"
Driver          "kbd"
Option          "AutoRepeat" "500 30"
# Specify which keyboard LEDs can be user-controlled (eg, with xset(1))
Option          "XkbRules" "xorg"
Option          "XkbModel" "pc105"
Option          "XkbLayout" "fr"
EndSection
```

Section "InputDevice"

```
# Identifier and driver
#   Option "Resolution"          "256"
# When using XQUEUE, comment out the above two lines, and uncomment
# the following line.
#   Option "Protocol"            "Xqueue"
# Baudrate and SampleRate are only for some Logitech mice. In
# almost every case these lines should be omitted.
#   Option "BaudRate"            "9600"
#   Option "SampleRate"          "150"
# Emulate3Buttons is an option for 2-button Microsoft mice
# Emulate3Timeout is the timeout in milliseconds (default is 50ms)
#   Option "Emulate3Buttons"
#   Option "Emulate3Timeout"     "50"
# ChordMiddle is an option for some 3-button Logitech mice
#   Option "ChordMiddle"
#       Identifier      "Mouse1"
#       Driver          "mouse"
#       Option          "Protocol" "IMPS/2"
#       Option          "Device"  "/dev/input/mice"
#       Option          "Emulate3Buttons" "true"
#       Option          "ZAxisMapping" "4 5"
# Mouse-speed setting for PS/2 mouse.
EndSection
```

Section "Monitor"

```
# HorizSync is in kHz unless units are specified.
# HorizSync may be a comma separated list of discrete values, or a
# comma separated list of ranges of values.
# NOTE: THE VALUES HERE ARE EXAMPLES ONLY. REFER TO YOUR MONITOR'S
# USER MANUAL FOR THE CORRECT NUMBERS.
#   HorizSync      30-64          # multisync
#   HorizSync      31.5, 35.2     # multiple fixed sync frequencies
#   HorizSync      15-25, 30-50   # multiple ranges of sync frequencies
# VertRefresh is in Hz unless units are specified.
# VertRefresh may be a comma separated list of discrete values, or a
# comma separated list of ranges of values.
# NOTE: THE VALUES HERE ARE EXAMPLES ONLY. REFER TO YOUR MONITOR'S
# USER MANUAL FOR THE CORRECT NUMBERS.
#       Identifier      "Iiyama Vision Master Pro 410"
#       HorizSync       27.0 - 96.0
#       VertRefresh     50.0 - 160.0
#       ModeLine "1280x1024" 135.00 1280 1328 1472 1712 1024 1033 1036 1070 +hsync +vsync
EndSection
```

Section "Device"

```
# The chipset line is optional in most cases. It can be used to override
# the driver's chipset detection, and should not normally be specified.
#   Chipset           "generic"
# The Driver line must be present. When using run-time loadable driver
# modules, this line instructs the server to load the specified driver
# module. Even when not using loadable driver modules, this line
# indicates which driver should interpret the information in this section.
# The BusID line is used to specify which of possibly multiple devices
# this section is intended for. When this line isn't present, a device
# section can only match up with the primary video device. For PCI
# devices a line like the following could be used. This line should not
# normally be included unless there is more than one video device
# intalled.
#   BusID             "PCI:0:10:0"
#   VideoRam          256
#   Clocks             25.2 28.3
#   Identifier         "Standard VGA"
#   Driver             "vga"
#   VendorName         "Unknown"
#   BoardName          "Unknown"
EndSection
```

Section "Device"

```
#nv"
# Insert Clocks lines here if appropriate
Identifier           "Nvidia Geforce 6200"
Driver               "nvidia"
#oprion twinview
Option               "TwinView"
Option               "NoTwinViewXineramaInfo"    "true"
Option               "SecondMonitorHorizSync"    "27.0-96.0"
Option               "SecondMonitorVertRefresh"  "50.0-160.0"
Option               "TwinViewOrientation"        "RightOf"
Option               "MetaModes"                  "1280x1024,1280x1024"
Option               "ConnectedMonitor"           "crt,crt"
#fin option twinview
EndSection
```

```

Section "Screen"
    Identifier      "Screen 1"
    Device          "Nvidia Geforce 6200"
    Monitor         "Iiyama Vision Master Pro 410"
    DefaultDepth    24
    Option          "NoLogo" "true"
    SubSection      "Display"
        Viewport    0 0
        Depth        8
        Modes        "1280x1024" "1024x768" "800x600" "640x480"
    EndSubSection
    SubSection      "Display"
        Viewport    0 0
        Depth        16
        Modes        "1280x1024" "1024x768" "800x600" "640x480"
    EndSubSection
    SubSection      "Display"
        Viewport    0 0
        Depth        24
        Modes        "1280x1024" "1024x768" "800x600" "640x480"
    EndSubSection
EndSection

```

15.3 Installation du Son via Alsa

ALSA (Advanced Linux Sound Architecture ou architecture avancée pour le son sous Linux) fournit les fonctionnalités audio et MIDI (Musical Instrument Digital Interface ou interface numérique pour les instruments de musique) dans le système d'exploitation Linux. ALSA est le système sonore par défaut dans les noyaux de la série 2.6. Il a remplacé OSS (Open Sound System) qui était utilisé dans les noyaux 2.4. Les fonctionnalités principales de ALSA comprennent le support pour une variété d'interfaces audio allant des cartes de son entrée de gamme jusqu'à l'équipement professionnel, des pilotes entièrement modulaires, l'intégration fiable des fils d'exécution et le SMP, une compatibilité rétroactive avec OSS et une bibliothèque pour l'espace utilisateur (alsa-lib) qui rend le développement d'applications on ne peut plus facile.

Une des principales forces de Gentoo est que l'utilisateur a le plein contrôle sur l'installation et la configuration de son système. L'utilisation de ALSA dans Gentoo suit ce principe. Il existe deux méthodes différentes pour mettre en place le support ALSA sur votre système.

Les deux méthodes sont :

1. Utiliser l'implémentation de ALSA fournie par votre noyau. Il s'agit de la méthode recommandée.
2. Utiliser le paquet logiciel `media-sound/alsa-driver` de Gentoo.

Les avantages et les inconvénients liés à l'utilisation de l'implémentation de ALSA présente dans le noyau sont les suivants :

ALSA dans le noyau	Avantages et inconvénients
+	Très stable puisque les pilotes sont intégrés avec le noyau.
+	Solution définitive qui ne requiert pas de réinstallations périodiques.
-	Parfois légèrement moins à jour que alsa-driver.

Voici ceux liés à l'utilisation de alsa-driver :

alsadriver	Avantages et inconvénients
+	Ce sont les pilotes les plus récents du projet ALSA.
-	Vous devez réinstaller alsa-driver chaque fois que vous recompilez votre noyau.
-	Nécessite que certaines options du noyau soient désactivées pour fonctionner correctement.

La principale différence entre alsa-driver et l'implémentation de ALSA présente dans le noyau est que alsa-driver contient des pilotes plus récents. Puisque cela fait généralement très peu de différence, vous êtes encouragé à utiliser l'implémentation du noyau, car elle est plus simple à gérer.

15.4 Installation d'Alsa

Dans la plupart des cas, les cartes de son (intégrées à la carte-mère ou non) sont de type PCI ; `lspci` vous aidera à trouver l'information nécessaire. Si `lspci` n'est pas déjà installé sur votre système, il vous faudra d'abord l'installer avec la commande : `emerge sys-apps/pciutils`. Si vous avez une carte USB, le programme `lsusb` du paquet `sys-apps/usbutils` vous sera peut-être utile. Pour les cartes ISA, essayez les utilitaires `sys-apps/isapnptools`.

Voici comment obtenir des informations relatives à la carte de son :

```
# lspci -v | grep -i audio
0000:00:0a.0 Multimedia audio controller: Creative Labs SB Live! EMU10k1 (rev 06)
```

Nous savons maintenant que la carte de son de l'ordinateur est une Sound Blaster Live! et que le fabricant de la carte est Creative Labs. Reportez-vous à la page ALSA Soundcard Matrix sur <http://www.alsa-project.org/alsa-doc/> et choisissez Creative Labs dans le menu déroulant. Vous serez renvoyé sur la page du matériel de Creative Labs où vous verrez que la carte SB Live! utilise le module `emu10k1`. C'est toute l'information dont nous avons besoin pour l'instant. Toutefois, si vous êtes curieux, vous pouvez suivre le lien qui vous amènera sur la page spécifique à `emu10k1` pour y lire de l'information détaillée.

Dans notre cas nous allons utiliser les `alsa-driver`.

Si vous ne savez pas quels pilotes sont nécessaires à votre carte de son, jetez un oeil à la section de ce guide portant sur `lspci`. Une fois identifié le nom du pilote (`emu10k1` dans notre exemple), éditez `/etc/make.conf` et ajoutez-y la variable `ALSA_CARDS`.

```
(Pour une seule carte de son :)
ALSA_CARDS="emu10k1"
(Pour spécifier plus d'une carte, séparez les noms avec des espaces :)
ALSA_CARDS="emu10k1 via82xx"
```

Dans mon cas j'utilise la carte son intégrée au chip Nforce4 et il se comporte comme un chip `intel8x0` d'où :

```
ALSA_CARDS="intel8x0"
```

dans le fichier `/etc/make.conf`

Si vous avez déjà compilé votre noyau et que vous voulez utiliser `alsa-driver`, vous devez procéder à un test pour vous assurer que votre système est correctement configuré. Si ce n'est pas le cas, vous ferez probablement face à des erreurs lorsque vous tenterez d'installer `alsa-driver`. L'exemple de code ci-bas illustre comment réaliser ces vérifications.

Note : Si vous n'utilisez pas `genkernel`, vérifiez les points ci-dessous :

1. `CONFIG_SOUND` doit être activé (fonctionnalités de base pour le son).
2. `CONFIG_SOUND_PRIME` doit être désactivé (support intégré de OSS).
3. `CONFIG_SND` doit être désactivé (support intégré de ALSA).

4. `/usr/src/linux` doit pointer vers le noyau que vous voulez utiliser avec ALSA.

(Nous assumons que le lien symbolique `linux` pointe vers le bon noyau.)

```
# cd /usr/src/linux
# grep SOUND .config
(1. Ok)
CONFIG_SOUND=y
(2. Ok)
CONFIG_SOUND_PRIME is not set
# grep SND .config
(Et 3. Ok)
CONFIG_SND is not set
```

Il ne reste qu'à installer `alsa-driver`.

```
# emerge alsa-driver
```

Important : Notez que vous devrez exécuter `emerge alsa-driver` à chaque fois que vous (re)compilerez le noyau, puisque les anciens pilotes sont supprimés.

15.4.1 Configurer Alsa

`alsa-utils` fait partie intégrale de ALSA et propose une variété de programmes fort utiles, incluant les scripts d'initialisation de ALSA. C'est pourquoi nous vous recommandons de l'installer :

```
# emerge alsa-utils
```

Note : Si vous avez activé ALSA dans le noyau et que vous n'avez pas compilé ALSA en temps que module, passez tout de suite à la section script d'initialisation de ALSA. Autrement, vous devez configurer ALSA, ce qui est rendu très facile par l'existence de l'outil `alsaconf`. (Ce dernier fait partie de `alsa-utils`.)

Note : Veuillez fermer tout programme qui pourrait utiliser la carte de son avant d'exécuter `alsaconf`. La façon la plus simple de configurer ALSA est d'exécuter `alsaconf`. Tapez simplement `alsaconf` dans un shell en tant qu'utilisateur root.

```
# alsaconf
```

Vous serez accueilli par une interface agréable basée sur un système de menus qui tentera de détecter automatiquement votre carte de son en interrogeant votre matériel.

On vous offrira alors une liste où vous devrez choisir votre carte. Une fois cela fait, on vous demandera si vous désirez que le programme modifie automatiquement `/etc/modules.d/alsa` pour y incorporer les changements nécessaires. Ensuite, le programme ajustera vos paramètres de volume à des niveaux optimum, exécutera `modules-update` et démarrera le service `/etc/init.d/alsasound`. Une fois que `alsaconf` a terminé son travail, vous pouvez continuer avec la mise en place du script d'initialisation de ALSA.

Nous avons presque terminé. Peu importe la méthode que vous avez choisie pour installer ALSA, vous aurez besoin de charger vos modules, d'initialiser ALSA et de restaurer vos paramètres de volume lors du démarrage. Le script d'initialisation de ALSA (nommé `alsasound`) s'occupe de tout cela. Ajoutez-le au niveau d'exécution du démarrage.

```
# rc-update add alsasound boot
* alsasound added to runlevel boot
* rc-update complete.
```

Il ne vous reste qu'à vérifier le fichier `/etc/conf.d/alsasound` pour vous assurer que la variable `SAVE_ON_STOP` est réglée à "yes". Ainsi, vos paramètres de volume seront sauvegardés lorsque vous redémarrerez ou éteindrez l'ordinateur.

Avant de tester votre système, il faut s'occuper d'un dernier détail important. Règle générale pour les systèmes UNIX : N'utilisez pas le compte root à moins que cela ne soit absolument nécessaire. Cela s'applique au cas présent.

La plupart du temps, vous êtes connecté en tant qu'utilisateur et voulez écouter de la musique ou utiliser votre carte de son. Pour cela, vous devez faire partie du groupe audio. C'est pourquoi nous allons maintenant ajouter les utilisateurs au groupe audio afin qu'ils n'aient pas de problème pour accéder aux périphériques sonores. Nous utiliserons l'utilitaire `gpasswd`. Vous devez être connecté avec le compte root.

(Remplacez `<nom_d'utilisateur>` par le nom approprié.)

```
# gpasswd -a <nom_d'utilisateur> audio
Adding user <nom_d'utilisateur> to group audio
```

Nous en avons fini avec les prérequis et la configuration. Lançons ALSA ! Si vous avez utilisé `alsaconf`, vous pouvez sauter cette étape puisque `alsaconf` s'en est occupé pour vous.

(Si ALSA a été compilé en modules :)

```
# /etc/init.d/alsasound start
* Loading ALSA modules ...
* Loading: snd-card-0 ... [ ok ]
* Loading: snd-pcm-oss ... [ ok ]
* Loading: snd-seq ... [ ok ]
* Loading: snd-emul0k1-synth ... [ ok ]
* Loading: snd-seq-midi ... [ ok ]
* Restoring Mixer Levels ... [ ok ]
(Si ALSA a été compilé dans le noyau :)
# /etc/init.d/alsasound start
* Loading ALSA modules ...
* Restoring Mixer Levels ... [ ok ]
```

Maintenant que nous nous sommes occupé du nécessaire, vérifions les niveaux du volume ; dans certains cas, les canaux sont muets. Pour ce faire, nous utiliserons `alsamixer`.

(Ouvre le programme en mode console. Seuls les paramètres requis sont affichés.)

```
# alsamixer
```

Important : Si vous éprouvez des difficultés avec `alsamixer` et que des erreurs se produisent telles que : `alsamixer : function snd_ctl_open failed for default : No such file or directory`, vous avez sans doute un problème avec la gestion du matériel par `udev`. Exécutez `killall udevd; udevstart` pour recharger les entrées du répertoire `/dev`, puis lancez `alsamixer`. Cela devrait résoudre vos problèmes.

Portez attention aux canaux *Master* et *PCM* sous lesquels on voit la mention **MM**, ce qui signifie qu'ils sont muets. Si vous essayez de jouer un son avec `alsamixer` dans ces circonstances, vous n'entendrez rien.

Attention : Vous devez activer à la fois les canaux *Master* et *PCM* (et régler leur volume à un niveau audible) pour pouvoir entendre du son.

Une façon désagréable de vérifier si votre carte fonctionne est de faire jouer du grésillement dans vos haut-parleurs. C'est plutôt bête, mais cela vous dira tout de suite si votre carte fonctionne.

```
# cat /dev/urandom > /dev/dsp
```

Note : `/dev/dsp` est un lien symbolique pointant vers `/dev/sound/dsp` ; il devrait être automatiquement créé. Essayez de rediriger la sortie vers `/dev/sound/dsp` si vous obtenez une erreur du type : `No such file or directory`. Vous devriez entendre du bruit de fond. Tapez `Ctrl + C` pour y mettre fin. Si vous n'entendez rien, c'est le moment de revenir en arrière pour identifier ce qui peut causer problème et apporter les corrections nécessaires.

Si tout ce qui précède s'est bien déroulé, vous devriez être capable d'écouter de la musique. Utiliser une application en mode ligne de commande telle que `media-sound/madplay` est une manière facile de faire ce test. Bien sûr, vous pourriez utiliser un logiciel un peu plus connu tel que `mpg123` ou `xmms`. Si vous êtes un fanatique du format ogg, vous pouvez utiliser `ogg123`, fourni par le paquet `mediasound/vorbis-tools`. Utilisez n'importe quel lecteur avec lequel vous êtes confortable. Comme toujours, il suffit d'utiliser `emerge` pour l'installer.

```
(Installer l'application que vous désirez :)
```

```
# emerge madplay mpg123 xmms
```

```
(Pour jouer des fichiers .ogg :)
```

```
# emerge vorbis-tools
```

Puis faites jouer votre chanson favorite.

```
# madplay -v /mnt/shyam/Music/Paul\ Oakenfold\ -\ Dread\ Rock.mp3
MPEG Audio Decoder 0.15.2 (beta) - Copyright (C) 2000-2004 Robert Leslie et al.
Title: Dread Rock
Artist: Paul Oakenfold
Album: Matrix Reloaded
Year: 2003
Genre: Soundtrack
Soundtrack
00:04:19 Layer III, 160 kbps, 44100 Hz, joint stereo (MS), no CRC
# ogg123 Paul\ Oakenfold\ -\ Dread\ Rock.ogg
Audio Device: Advanced Linux Sound Architecture (ALSA) output
Playing: Paul Oakenfold - Dread Rock.ogg
Ogg Vorbis stream: 2 channel, 44100 Hz
Genre: Soundtrack
Transcoded: mp3;160
Title: Dread Rock
Artist: Paul Oakenfold
Date: 2003
Album: Matrix Reloaded
Time: 00:11.31 [04:28.75] of 04:40.06 (200.6 kbps) Output Buffer 96.9%u
```

Vous pouvez maintenant ajouter `alsa` aux options de votre variable `USE` dans `/etc/make.conf` afin de vous assurer que le support `ALSA` est présent dans vos applications. Sur certaines architectures, telles que `x86` et `amd64`, cette option est activée par défaut.

15.4.2 Résolution de problèmes

Si vous êtes incapable d'entendre du son, la première chose à vérifier est le réglage de `alsamixer`. 80 % des problèmes impliquent un canal muet ou dont le volume est trop bas. Consultez aussi l'applet de configuration du son de votre gestionnaire de fenêtres pour vous assurer que le volume est réglé à un niveau audible.

`/proc` est votre ami. Dans le cas présent, `/proc/asound` se révélera votre meilleur ami.

Jetons un coup d'oeil à l'information que vous pouvez y consulter.

```
(D'abord et avant tout, si /proc/asound/cards contient votre carte, ALSA
a correctement reconnu celle-ci.)
# cat /proc/asound/cards
0 [Live ]: EMU10K1 - Sound Blaster Live!
Sound Blaster Live! (rev.6, serial:0x80271102) at 0xb800, irq 11
(Si vous utilisez l'implémentation de ALSA du noyau et vous demandez,
comme moi, jusqu'à quel point votre version date par rapport à alsa-driver,
voici comment afficher la version actuelle de ALSA :)
# cat /proc/asound/version
Advanced Linux Sound Architecture Driver Version 1.0.8 (Thu Jan 13 09:39:32 2005 UTC).
(Détails de l'émulation OSS par ALSA :)
# cat /proc/asound/oss/sndstat
Sound Driver:3.8.1a-980706 (ALSA v1.0.8 emulation code)
Kernel: Linux airwolf.zion 2.6.11ac1 #2 Wed May 4 00:35:08 IST 2005 i686
Config options: 0
Installed drivers:
Type 10: ALSA emulation
Card config:
Sound Blaster Live! (rev.6, serial:0x80271102) at 0xb800, irq 11
Audio devices:
0: EMU10K1 (DUPLEX)
Synth devices: NOT ENABLED IN CONFIG
Midi devices:
0: EMU10K1 MPU-401 (UART)
Timers:
7: system timer
Mixers:
0: SigmaTel STAC9721/23
```

Un autre problème très souvent rencontré par les utilisateurs est la fameuse erreur *Unknown symbol in module*. En voici un exemple :

```
# /etc/init.d/alsasound start
* Loading ALSA modules ...
* Loading: snd-card-0 ... [ ok ]
* Loading: snd-pcm-oss ...
WARNING: Error inserting snd_mixer_oss
(/lib/modules/2.6.12-gentoo-r6/kernel/sound/core/oss/snd-mixer-oss.ko): Unknown
symbol in module, or unknown parameter (see dmesg) FATAL: Error inserting
snd_pcm_oss
(/lib/modules/2.6.12-gentoo-r6/kernel/sound/core/oss/snd-pcm-oss.ko): Unknown
symbol in module, or unknown parameter (see dmesg)
[ !! ]
* Loading: snd-mixer-oss ...
FATAL: Error inserting snd_mixer_oss
(/lib/modules/2.6.12-gentoo-r6/kernel/sound/core/oss/snd-mixer-oss.ko): Unknown
symbol in module, or unknown parameter (see dmesg)
[ !! ]
* Loading: snd-seq ... [ ok ]
* Loading: snd-emu10k1-synth ... [ ok ]
* Loading: snd-seq-midi ... [ ok ]
* Restoring Mixer Levels ... [ ok ]
```

Et lorsque vous regardez dans le dmesg comme suggéré :

```
(Seules les lignes intéressantes ont été conservées.)
# dmesg | less
ACPI: PCI Interrupt 0000:02:06.0[A] -> Link [APC3] -> GSI 18 (level, low) -> IRQ 209
snd_mixer_oss: Unknown symbol snd_unregister_oss_device
snd_mixer_oss: Unknown symbol snd_register_oss_device
snd_mixer_oss: Unknown symbol snd_mixer_oss_notify_callback
snd_mixer_oss: Unknown symbol snd_oss_info_register
snd_pcm_oss: Unknown symbol snd_unregister_oss_device
snd_pcm_oss: Unknown symbol snd_register_oss_device
snd_pcm_oss: Unknown symbol snd_mixer_oss_ioctl_card
snd_pcm_oss: Unknown symbol snd_oss_info_register
snd_mixer_oss: Unknown symbol snd_unregister_oss_device
snd_mixer_oss: Unknown symbol snd_register_oss_device
snd_mixer_oss: Unknown symbol snd_mixer_oss_notify_callback
snd_mixer_oss: Unknown symbol snd_oss_info_register
```

Le problème ci-dessus apparaît lorsque vous passez des pilotes du paquet **alsa-driver** à ceux intégrés au noyau. Lorsque vous désinstallez le paquet **alsa-driver**, les fichiers des modules sont protégés et ne sont pas effacés, ce qui crée un mélange des deux versions des pilotes et ne fonctionne pas. La solution est très simple. Vous n'avez qu'à supprimer manuellement les fichiers qui causent problème, après avoir désinstallé le paquet **alsa-driver**. Assurez-vous de retirer la bonne version des pilotes et de ne pas effacer les pilotes actuels !

```
(Remplacez VERSION par la version de votre noyau.)
# rm -rf /lib/modules/VERSION/alsa-driver
```

15.5 Installation de Gnome

15.5.1 Introduction

GNOME est un environnement de bureau et une plate-forme de développement. Ce logiciel libre est le bureau de choix pour de nombreux chefs de file de l'industrie. Il est intéressant aussi bien pour les entreprises que pour les individus et les développeurs.

15.5.2 Installer GNOME

Avant de débiter l'installation de GNOME, vous souhaitez peut-être modifier votre variable **USE**. Assurez-vous que les options **gtk** et **gnome** sont présentes dans votre variable **USE** du fichier **/etc/make.conf**. Si vous voulez utiliser **hald**, le démon qui permet l'abstraction vis-à-vis du matériel ou **howl** qui offre un système de détection DNS similaire à Rendezvous pour Mac OS X, veuillez ajouter les options correspondantes à votre variable **USE**. Si vous ne voulez pas du support KDE (l'autre environnement de bureau populaire), enlevez **qt** et **kde**.

```
USE="-qt -kde gtk gnome hal howl"
```

Une fois cela fait, installez GNOME avec la commande **emerge** et les paquets **gnome** et **xscreensaver** :

```
# emerge gnome xscreensaver
```

Maintenant, mettez à jour vos variables d'environnement :

```
# env-update && source /etc/profile
```

Si vous avez prêté attention à la sortie de la dernière exécution de **emerge**, vous avez sans doute remarqué que l'on vous suggère d'ajouter **famd** au niveau d'exécution par défaut afin que **nautilus** et **gnome-vfs** surveillent les modifications apportées aux fichiers :

```
# /etc/init.d/famd start
# rc-update add famd default
# /etc/init.d/hald start
# rc-update add hald default
# /etc/init.d/mDNSResponder start
# rc-update add mDNSResponder default
```

Remarques : si vous avez un erreur du style suivant lors de la compilation :

```
Calculating dependencies ...done!
>>> emerge (1 of 6) media-libs/smpeg-0.4.4-r5 to /
!!! Security Violation: A file exists that is not in the manifest.
!!! File: files/digest-smpeg-0.4.4-r4
```

Comme décrit à l'adresse suivante http://gentoo-wiki.com/TIP_Downloading_distfiles_on_another_machine, vous pouvez désactiver l'option `strict` afin d'éviter certains tests. Modifiez votre fichier `/etc/make.conf` comme suit :

```
FEATURES--strict
```

15.5.3 Connexion graphique GNOME

Si vous souhaitez que l'invite de connexion graphique GNOME (GDM) soit exécutée lors du démarrage (afin de pouvoir ouvrir une session graphiquement), vous devez ajouter le script d'initialisation `xdm` au niveau d'exécution par défaut :

```
# rc-update add xdm default
```

Ensuite, éditez le fichier `/etc/rc.conf` et définissez la variable `DISPLAYMANAGER`.

```
DISPLAYMANAGER="gdm"
```

À partir de maintenant, si vous redémarrez l'ordinateur, l'invite de connexion graphique GNOME vous demandera votre nom d'utilisateur et votre mot de passe et, par défaut, sélectionnera GNOME comme environnement de bureau (mais, bien sûr, vous pourrez en choisir un autre parmi `/usr/share/xsessions/`). Si vous utilisez GDM, il n'est pas nécessaire de modifier votre fichier `~/.xinitrc`. Pour utiliser les fonctionnalités de `hald`, démarrez simplement `gnome-volume-manager` et modifiez ses préférences.

15.6 Mise à l'heure automatique

Il est possible de synchroniser l'horloge système par l'intermédiaire d'internet et le protocole NTP (Network Time Protocol). Pour ça il faut installer le paquet `ntp-client` et non `ntpd`.

```
# emerge ntp-client
```

Ensuite vous configurez le fichier `/etc/ntp.conf` en lui spécifiant quel serveur de temps vous voulez connecter (j'ai choisi le serveur `chronos.cru.fr`).

```
# NOTES:
# - you should only have to update the server line below
# - if you start getting lines like 'restrict' and 'fudge'
#   and you didnt add them, AND you run dhcpcd on your
#   network interfaces, be sure to add '-Y -N' to the
#   dhcpcd_ethX variables in /etc/conf.d/net

# Name of the servers ntpd should sync with
# Please respect the access policy as stated by the responsible person.
#server          ntp.example.tld          iburst

#server pool.ntp.org
server chronos.cru.fr
##
# A list of available servers can be found here:
# http://www.pool.ntp.org/
# http://www.pool.ntp.org/#use
# A good way to get servers for your machine is:
# netselect -s 3 pool.ntp.org
##

# you should not need to modify the following paths
driftfile        /var/lib/ntp/ntp.drift
#server ntplocal.example.com prefer
#server timeserver.example.org

# Warning: Using default NTP settings will leave your NTP
# server accessible to all hosts on the Internet.

# If you want to deny all machines (including your own)
# from accessing the NTP server, uncomment:
#restrict default ignore

# To deny other machines from changing the
# configuration but allow localhost:
restrict default nomodify nopeer
restrict 127.0.0.1

# To allow machines within your network to synchronize
# their clocks with your server, but ensure they are
# not allowed to configure the server or used as peers
# to synchronize against, uncomment this line.
#
#restrict 192.168.0.0 mask 255.255.255.0 nomodify nopeer notrap
```

Puis on va lancer tout cela automatiquement grâce à la commande `rc-update` :

```
# rc-update add ntp-client default
```

15.7 Petits trucs en vrac

15.7.1 Flash et Firefox

Si vous avez recompilé les sources de firefox, il vous sera impossible d'installer et d'utiliser Flash. En effet il n'existe pas de version 64 bits de Flash. Vous devrez donc installer les binaires 32 bits de Firefox et le paquet `netscape-flash` pour pouvoir bénéficier de toutes les applications et sites web en flash.

15.7.2 Imprimante HP en réseau et Cups

Afin de bénéficier de l'impression via Cups, j'ai installé les paquets suivants : `cups`, `foomatic-db`, `gimp-print`, `app-text/ghostscript-esp`, `hpijs`.

15.7.3 Utilisateur lambda et groupes

Si vous désirez que votre utilisateur lambda puisse accéder aux ressources de la machine (carte son, cdrom etc...), il est nécessaire de l'ajouter à différents groupes. La commande est la suivante :

```
# gpasswd -a nom_utilisateur cdrom
# gpasswd -a nom_utilisateur audio
```

Voici mon fichier `/etc/group`

```
/etc/group
root::0:root
bin::1:root,bin,daemon
daemon::2:root,bin,daemon
sys::3:root,bin,adm
adm::4:root,adm,daemon
tty::5:
disk::6:root,adm
lp:!:7:lp,yann
mem::8:
kmem::9:
wheel::10:root,yann
floppy::11:root
mail::12:mail
news::13:news
uucp::14:uucp
man::15:man
cron::16:cron
console::17:
audio:!:18:yann
cdrom:!:19:yann
dialout::20:root
ftp::21:
sshd::22:
at::25:at
```

```
tape::26:root
video:!:27:root,yann
squid::31:squid
gdm::32:gdm
xfs::33:xfs
games:!:35:yann
named::40:named
mysql:x:60:
postgres::70:
cdrw::80:
nut::84:
usb::85:
vpopmail:x:89:
users::100:games,yann
nofiles:x:200:
qmail:x:201:
postfix:x:207:
postdrop:x:208:
smmsp:x:209:smmsp
slocate::245:
portage::250:portage
utmp:x:406:
nogroup::65533:
nobody::65534:
messagebus:x:407:
haldaemon:x:408:
rpc:x:111:
ntp:x:123:
```

15.7.4 Tout ce qui est lancé au démarrage

La commande `rc-update show` permet de voir tout ce qui est lancé au démarrage. En voici un exemple :

```

biboo etc # rc-update show
      alsasound | boot
      bootmisc | boot
      checkfs  | boot
      checkroot | boot
      clock    | boot
      coldplug |      default
consolefont | boot
crypto-loop |
      cupsd   |      default
      dbus    |
domainname | boot default
      esound  |
      famd    |      default
      gpm     |      default
      hald    |      default
      hdparm  |
      hostname | boot
      hotplug |      default
      keymaps | boot
      local   |      default nonetwork
localmount | boot
mDNSResponder |      default
      modules | boot
      mysql   |
      net.eth0 |      default
      net.lo  | boot
      netmount |      default
      nifd    |
      nscd    |
ntp-client |      default
      ntpd    |
      numlock |      default
      portmap |
      rmnologin | boot
      rsyncd  |
      spamd   |
      sshd    |      default
      syslog-ng |      default
      urandom | boot
      vixie-cron |      default
      xdm     |      default
biboo etc #

```

15.7.5 Mise à jour vers php5

Lors de la mise à jour vers php5, j'ai eu quelques soucis avec un site web, qui ne fonctionnait plus. En effet par défaut, l'option `--disable-session` est configurée. De ce fait tous les sites en PHP qui utilisent les fonctions `sessionstart()` ne fonctionne plus et pprovoque l'erreur suivante `Call to undefined function session_start()`. Pour que tout rentre dans l'ordre

il suffit de passer quelques options lors de la compilation du paquet. Pour cela éditer le fichier `/etc/portage/package.use` et ajouter la ligne suivante. Cela devrait convenir à la plupart des utilisateurs.

```
vim /etc/portage/package.use
=dev-lang/php-5* dba -berkdb apache2 ctype gd jpeg mysql pcre png session truetype xml
```

16 Installation rapide

gentoo Le noyau par défaut avec support des processeurs K8 avec NUMA. Voici les options disponibles.

```
agpgart - Active le support AGP.
acpi=on - Active le support ACPI.
ide=nodma - Désactive le support DMA.
doscsi - Active le support pour le matériel SCSI.
dopcmcia - Active le support pour le matériel PCMCIA.
nofirewire - Désactive le support pour le matériel FireWire.
nokeymap - Désactive le support des claviers internationaux.
docache - Charge les fichiers du CD en mémoire pour pouvoir
éjecter le CD et en utiliser un autre.
nodetect - Désactive la détection automatique.
nusb - Désactive le support du matériel USB.
nodhcp - Ne pas utiliser DHCP pour obtenir une adresse IP.
nohotplug - Désactive hotplug (un programme de chargement dans le noyau).
noapic - Désactive l'apic (p.ex. en cas de problèmes avec le matériel.)
noevms Désactive le chargement des modules EVMS2
nolvm2 Désactive le chargement des modules LVM2
hdx=stroke - Permet de partitionner tout le disque dur même si le
BIOS ne reconnaît pas les disques de grande capacité.
noload=module1[,module2[,...]] - Désactive le chargement des modules spécifiés
```

```
# loadkeys fr
```

Nous allons concevoir un plan de partitionnement. Le nombre de partitions dépend beaucoup de votre environnement. Dans la cas d'une station de travail, nous opterons pour 4 partitions :

- ▷ une partition d'une centaine de Mo, pour le répertoire `/boot`
- ▷ une partition de swap, d'une taille équivalente à la mémoire
- ▷ une partition qui recevra la racine `/` du système de fichier
- ▷ une partition qui recevra les données des utilisateurs `/home`

`fdisk` est un outil populaire et puissant pour diviser votre disque en partitions. Lancez `fdisk` sur votre disque (dans notre exemple, nous utilisons `/dev/hda`) :

```
# fdisk /dev/hda/
```

Nous allons ensuite formater les différentes partitions à l'aide des commandes correspondantes suivantes :

Système de fichiers	Commande de création
ext2	<code>mke2fs</code>
ext3	<code>mke2fs -j</code>
reiserfs	<code>mkreiserfs</code>
xfs	<code>mkfs.xfs</code>
jfs	<code>mkfs.jfs</code>

```
# mke2fs /dev/hda1
# mke2fs -j /dev/hda3
# mke2fs -j /dev/hda4

# mkswap /dev/hda2/

# swapon /dev/hda2/

# mount /dev/hda3 /mnt/gentoo
# mkdir /mnt/gentoo/boot
# mount /dev/hda1 /mnt/gentoo/boot

# date
Fri Mar 29 16:21:18 CEST 2005
# date 032916212005

# ls /mnt/cdrom/stages
ls: /mnt/cdrom/stages: No such file or directory
# mount /dev/cdroms/cdrom0 /mnt/cdrom
# ls /mnt/cdrom/stages
# cd /mnt/gentoo

# tar xvjpf /mnt/cdrom/stages/stage3-<sous-architecture>-2005.1-r1.tar.bz2
# ls /mnt/cdrom/snapshots
# tar xvjf /mnt/cdrom/snapshots/portage-<date>.tar.bz2 -C /mnt/gentoo/usr

# mkdir /mnt/gentoo/usr/portage/distfiles
# cp /mnt/cdrom/distfiles/* /mnt/gentoo/usr/portage/distfiles/

# nano -w /mnt/gentoo/etc/make.conf

# These settings were set by the catalyst build script that automatically built
this stage
# Please consult /etc/make.conf.example for a more detailed example
LINGUAS="fr"
CFLAGS="-march=k8 -pipe -O2"
CHOST="x86_64-pc-linux-gnu"
CXXFLAGS="${CFLAGS}"
MAKEOPTS="-j2"

GENTOO_MIRRORS="http://ftp.belnet.be/mirror/rsync.gentoo.org/gentoo"
SYNC="rsync://rsync.europe.gentoo.org/gentoo-portage"

USE="mmx mmxext sse 3dnowext apache2 esd ffmpeg network nsplugin v4l
win32codecs xine gtk gtk2 gnome cdparanoia bash-completion cups encode
foomaticdb glut ftp imagemagick ipv6 lesstif mysql nls spell eds opengl
pam xv type1-fonts bitmap-fonts truetype-fonts ogg oggvorbis ssl threads
readline foomaticdb gif png jpeg quicktime xml2 sdl java crypt pdflib berkbd
mpeg truetype dvd alsa cdr gnome hal howl mikmod -arts -kde -ldap -qt "

ALSA_CARDS="intel8x0"

FEATURES="-strict" #évite les problème de "security violation"

# mount -t proc none /mnt/gentoo/proc
```

```
# cp -L /etc/resolv.conf /mnt/gentoo/etc/resolv.conf

# chroot /mnt/gentoo /bin/bash
# env-update
* Caching service dependencies...
# source /etc/profile

# emerge --metadata

# less /usr/portage/profiles/use.desc
(Utilisez les flèches de votre clavier pour faire défiler le texte et
tapez 'q' pour quitter.)
# nano -w /etc/make.conf
USE="-gtk -gnome qt kde dvd alsa cdr"

USE="mmx mmxext sse 3dnow 3dnowext apache2 esd ffmpeg network nsplugin v4l
win32codecs xine gtk gtk2 gnome cdparanoia bash-completion cups encode
foomaticdb glut ftp imagemagick lesstif mysql ogg oggvorbis threads win32codecs
dvd alsa cdr -arts -kde -ldap -qt"

# ls /usr/share/zoneinfo
(En supposant que vous utilisez l'heure de Paris.)
# cp /usr/share/zoneinfo/Europe/Paris /etc/localtime

# USE="-doc" emerge gentoo-sources

# ls -l /usr/src/linux
lrwxrwxrwx 1 root root 12 Oct 13 11:04 /usr/src/linux -> linux-2.6.12-gentoo-r10

# rm /usr/src/linux
# cd /usr/src
# ln -s linux-2.6.12-gentoo-r10 linux

# emerge genkernel

genkernel --menuconfig all

# emerge coldplug
# rc-update add coldplug default

# find /lib/modules/<kernel version>/ -type f -iname '*.o' -or -iname '*.ko'

# nano -w /etc/modules.autoload.d/kernel-2.6
3c59x

# nano -w /etc/fstab/
```

voici un exemple de fichier fstab

/dev/hda1	/boot	ext2	noauto,noatime	1 2	
/dev/hda2	none	swap	sw		0 0
/dev/hda3	/	ext3	noatime		0 1
/dev/hda4	/home	ext3	noatime	0 1	
none /proc	proc	defaults			0 0
none /dev/shm	tmpfs	nodev,nosuid,noexec		0 0	
/dev/cdroms/cdrom0	/mnt/cdrom	auto	noauto,user	0 0	

```
# nano -w /etc/conf.d/hostname
(Définissez le nom de votre machine :)
HOSTNAME="tux"
# nano -w /etc/conf.d/domainname
(Définissez le nom de votre domaine :)
DNSDOMAIN="homenetwork"

# rc-update add domainname default

# nano -w /etc/conf.d/net
# This blank configuration will automatically use DHCP for any net.*
# scripts in /etc/init.d. To create a more complete configuration,
# please review /etc/conf.d/net.example and save your configuration
# in /etc/conf.d/net (this file :]).
```

Pour entrer une adresse fixe, un masque de réseau et une adresse de passerelle, vous devez définir `config_eth0` et `routes_eth0` :

```
config_eth0=( "192.168.0.2 netmask 255.255.255.0 brd 192.168.0.255" )
routes_eth0=( "default gw 192.168.0.1" )
```

Pour

utiliser DHCP et lui passer des options, définissez les variables `config_eth0` et `dhcp_eth0` :

```
config_eth0=( "dhcp" )
dhcp_eth0="nodns nontp nonis"

# rc-update add net.eth0 default
```

Modifions maintenant le fichier `/etc/hosts` :

```
# nano -w /etc/hosts/
127.0.0.1 localhost tux

# passwd
```

Gentoo utilise `/etc/rc.conf` pour la configuration générale qui s'applique à l'ensemble du système. Ouvrez `/etc/rc.conf` et appréciez les commentaires qui s'y trouvent :)

```
# nano -w /etc/rc.conf
```

Le fichier `/etc/conf.d/keymaps` permet de spécifier le type de clavier que vous utilisez.

```
# nano -w /etc/conf.d/keymaps

# emerge syslog-ng
# rc-update add syslog-ng defaultd

# emerge vixie-cron
# rc-update add vixie-cron default
(Seulement si vous avez choisi dcron ou fcron.) # crontab /etc/crontab

# emerge slocate

# emerge dhcpcd

# emerge grub
```

```
# nano -w /boot/grub/grub.conf
default 0
timeout 30
splashimage=(hd0,0)/boot/grub/splash.xpm.gz
title=Gentoo Linux
root (hd0,0)
kernel /boot/kernel-genkernel-amd64-2.6.12-gentoo-r10 root=/dev/ram0
  init=/linuxrc ramdisk=8192 real_root=/dev/hda3 udev
initrd /boot/initramfs-genkernel-amd64-2.6.12-gentoo-r10
# Uniquement pour démarrer un autre système.
title=Windows XP
root (hd0,5)
makeactive
chainloader +1
```

Si comme moi, votre premier disque contient windows, et le second la distribution Gentoo vous avez le fichier suivant :

```
# nano -w /boot/grub/grub.conf
default 0
timeout 30
splashimage=(hd1,0)/boot/grub/splash.xpm.gz
title=Gentoo Linux
root (hd1,0)
kernel /boot/kernel-genkernel-amd64-2.6.12-gentoo-r10 root=/dev/ram0
  init=/linuxrc ramdisk=8192 real_root=/dev/hdb3 udev
initrd /boot/initramfs-genkernel-amd64-2.6.12-gentoo-r10
# Uniquement pour démarrer un autre système.
title=Windows XP
root (hd0,0)
makeactive
chainloader +1
```

```
title=Gentoo Linux
root (hd0,0)
kernel /kernel-2.6.12-gentoo-r10 root=/dev/hda3 video=vesafb
```

```
# grep -v rootfs /proc/mounts > /etc/mtab
# grub-install /dev/hda
```

```
# exit
# cd
cdimage ~# umount /mnt/gentoo/boot /mnt/gentoo/proc /mnt/gentoo
cdimage ~# reboot
```

```
Login: root
Password: (votre mot de passe root)
# useradd -m -G users,wheel,audio -s /bin/bash yann
# passwd yann
Password: (tapez le mot de passe pour john)
Re-enter password: (retapez-le pour vérifier)
```

```
# emerge --sync
```

```
$ emerge --search pdf
```

Si vous voulez aussi chercher dans les descriptions, utilisez l'option `--searchdesc` (ou `-S`) :

```
$ emerge --searchdesc pdf
```

La liste des paquets affichés contient quelques informations utiles pour chaque paquet.

```
* net-print/cups-pdf
Latest version available: 1.5.2
Latest version installed: [ Not Installed ]
Size of downloaded files: 15 kB
Homepage: http://cip.physik.uni-wuerzburg.de/~vrbehr/cups-pdf/
Description: Provides a virtual printer for CUPS to produce PDF files.
License: GPL-29
```

```
# emerge gnumeric
```

```
# emerge --pretend gnumeric
```

```
# emerge --fetchonly gnumeric
```

```
# emerge -vp alsa-lib
```

```
[ebuild N ] media-libs/alsa-lib-1.0.9_rc3 +doc -jack 674 kB
```

```
# emerge --unmerge gnumeric
```

```
# emerge --update world
```

```
# emerge --update --deep world
```

```
# emerge --update --deep --newuse world
```

```
# emerge --pretend --verbose mozilla
```

These are the packages that I would merge, in order:

Calculating dependencies ...done!

```
[ebuild N ] net-www/mozilla-1.5-r1 +java +crypt -ipv6 -gtk2 +ssl +ldap
+gnome -debug +mozcalendar -mozaccess -mozxmlterm -moznoirc -moznomail
-moznocompose -moznoxft
```

```
# ACCEPT_KEYWORDS="~amd64" emerge gnumeric
```

```
# emerge -pv mozilla
```

These are the packages that I would merge, in order:

Calculating dependencies ...done!

```
[ebuild N ] net-www/mozilla-1.5-r1 +java +crypt -ipv6 -gtk2 +ssl +ldap
+gnome -debug +mozcalendar -mozaccess -mozxmlterm -moznoirc -moznomail
-moznocompose -moznoxft
```

```
# etc-update
```

```
# rc-update show
```

```
# rc-update del postfix default
```

```
# emerge -Dup world
```

Calculating dependencies ...done!

```
[ebuild NS ] sys-kernel/gentoo-sources-2.6.9-r2 [2.6.8-r5]
```

```
# emerge -u gentoo-sources
```

```
# genkernel --menuconfig --bootloader=grub all
```

```
# emerge -P gentoo-sources
```

```
# emerge -pv xorg-x11
```

```
# emerge xorg-x11
```

```
# env-update
# source /etc/profiler

# xorgconfig

# xorgcfg
(Si X plante ou si la configuration échoue, essayez plutôt)
# xorgcfg -textmode

# cp /root/xorg.conf.new /etc/X11/xorg.conf

# startx

# emerge nvidia-glx

# ACCEPT_KEYWORDS="~amd64" emerge nvidia-glx
```

mettre dans le fichier `/etc/portage/package.keywords` les lignes suivantes :

```
media-video/nvidia-kernel ~amd64
media-video/nvidia-glx ~amd64
```

```
# modprobe nvidia
```

```
# modules-update
```

```
biboo etc # more ./modules.d/nvidia
```

```
# Nvidia drivers support
```

```
alias char-major-195 nvidia
```

```
alias /dev/nvidiactl char-major-195
```

```
# To tweak the driver the following options can be used, note that
# you should be careful, as it could cause instability!! For more
# option see /usr/share/doc/nvidia-kernel-1.0.6629-r4/README.gz
```

```
#
```

```
# To enable Side Band Addressing: NVreg_EnableAGPSBA=1
```

```
#
```

```
# To enable Fast Writes: NVreg_EnableAGPFW=1
```

```
#
```

```
# To enable both for instance, uncomment following line:
```

```
#
```

```
#options nvidia NVreg_EnableAGPSBA=1 NVreg_EnableAGPFW=1
```

```
biboo etc #
```

```
Section "Device"
```

```
Identifier "nVidia Inc. GeForce6200"
```

```
Driver "nvidia"
```

```
VideoRam 131072
```

```
EndSection
```

```
Section "Module"
```

```
(...)
```

```
# Load "dri"
```

```
Load "glx"
```

```
(...)
```

```
EndSection
```

```
Section "Screen"
(...)
DefaultDepth 16
Subsection "Display"
(...)
EndSection
```

Lancez `opengl-update` afin que le serveur X utilise les bibliothèques GLX de nVidia :

```
# opengl-update nvidia
```

Ajouter des utilisateurs dans le groupe `video`. Afin que vos utilisateurs puissent accéder aux fichiers de périphérique nVidia, il faut les ajouter au groupe `video` :

```
# gpasswd -a nom_utilisateur video
```

Pour tester votre carte nVidia, démarrez X et lancez la commande `glxinfo | grep direct`. Cela devrait vous répondre que le rendu direct est activé :

```
$ glxinfo | grep direct
direct rendering: Yes
```

Pour obtenir votre taux de FPS (images par seconde), lancez `glxgears`.

```
# less /usr/share/doc/nvidia-glx-*/README.txt.gz
```

```
# nano -w /etc/modules.d/nvidia
(Mettez à jour les fichiers des modules.)
# modules-update
Déchargez le module nvidia...)
# modprobe -r nvidia
(... et rechargez-le.)
# modprobe nvidia
```

```
Section "Device"
Identifiant "nVidia Inc. GeForce6200"
Driver "nvidia"
Option "NoLogo" "true"
VideoRam 131072
EndSection
```

```
# emerge unmerge nvidia-kernel
# emerge unmerge nvidia-glx
```

ensuite on enlève le module `nvidia` de la mémoire et on met à jour la liste des modules.

```
# rmmod nvidia
# modules-update
```

Ensuite il faut télécharger les pilotes propriétaires Nvidia à l'adresse suivante http://www.nvidia.com/object/linux_display_amd64_1.0-8178.html. Si ce lien ne fonctionne plus, c'est que la version a changé et rendez-vous sur la page <http://www.nvidia.com/content/drivers/drivers.asp> et sélectionnez le pilote désiré.

Il suffit alors de changer les droits du fichier `NVIDIA-Linux-x86_64-1.0-<version>-pkg2.run` et de l'exécuter :

```
# chmod 755 NVIDIA-Linux-x86_64-1.0-<version>-pkg2.run
# ./NVIDIA-Linux-x86_64-1.0-<version>-pkg2.run
```

```
Section "Device"
#nv"
# Insert Clocks lines here if appropriate
Identifier      "Nvidia Geforce 6200"
Driver          "nvidia"
#option twinview
Option          "TwinView"
Option          "NoTwinViewXineramaInfo"    "true"
Option          "SecondMonitorHorizSync"    "27.0-96.0"
Option          "SecondMonitorVertRefresh"  "50.0-160.0"
Option          "TwinViewOrientation"       "RightOf"
Option          "MetaModes"                 "1280x1024,1280x1024"
Option          "ConnectedMonitor"          "crt,crt"
#fin option twinview
EndSection
```

```
# nvidia-xconfig: X configuration file generated by nvidia-xconfig
# nvidia-xconfig:  version 1.0  (buildmeister@builder26)  Wed Dec 14 17:16:10 PST 2005

# File generated by xorgconfig.
#
# Copyright 2004 The X.Org Foundation
#
# Permission is hereby granted, free of charge, to any person obtaining a
# copy of this software and associated documentation files (the "Software"),
# to deal in the Software without restriction, including without limitation
# the rights to use, copy, modify, merge, publish, distribute, sublicense,
# and/or sell copies of the Software, and to permit persons to whom the
# Software is furnished to do so, subject to the following conditions:
#
# The above copyright notice and this permission notice shall be included in
# all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
# IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
# FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.  IN NO EVENT SHALL
# The X.Org Foundation BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
# WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF
# OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
# SOFTWARE.
#
# Except as contained in this notice, the name of The X.Org Foundation shall
# not be used in advertising or otherwise to promote the sale, use or other
# dealings in this Software without prior written authorization from
# The X.Org Foundation.
#
# *****
# Refer to the xorg.conf(5x) man page for details about the format of
# this file.
# *****
# *****
# Module section -- this section is used to specify
# which dynamically loadable modules to load.
# *****
#
# *****
# Files section.  This allows default font and rgb paths to be set
# *****
# *****
# Server flags section.
# *****
# *****
# Input devices
# *****
# *****
# Core keyboard's InputDevice section
# *****
```

```

# *****
# Core Pointer's InputDevice section
# *****
# *****
# Other input device sections
# this is optional and is required only if you
# are using extended input devices. This is for example only. Refer
# to the xorg.conf man page for a description of the options.
# *****
#
# Section "InputDevice"
#   Identifier "Mouse2"
#   Driver      "mouse"
#   Option      "Protocol"          "MouseMan"
#   Option      "Device"            "/dev/mouse2"
# EndSection
#
# Section "InputDevice"
#   Identifier "spaceball"
#   Driver      "magellan"
#   Option      "Device"            "/dev/cua0"
# EndSection
#
# Section "InputDevice"
#   Identifier "spaceball2"
#   Driver      "spaceorb"
#   Option      "Device"            "/dev/cua0"
# EndSection
#
# Section "InputDevice"
#   Identifier "touchscreen0"
#   Driver      "microtouch"
#   Option      "Device"            "/dev/ttyS0"
#   Option      "MinX"              "1412"
#   Option      "MaxX"              "15184"
#   Option      "MinY"              "15372"
#   Option      "MaxY"              "1230"
#   Option      "ScreenNumber"      "0"
#   Option      "ReportingMode"     "Scaled"
#   Option      "ButtonNumber"      "1"
#   Option      "SendCoreEvents"
# EndSection
#

```

```

# Section "InputDevice"
#   Identifier "touchscreen1"
#   Driver      "elo2300"
#   Option      "Device"      "/dev/ttyS0"
#   Option      "MinX"        "231"
#   Option      "MaxX"        "3868"
#   Option      "MinY"        "3858"
#   Option      "MaxY"        "272"
#   Option      "ScreenNumber" "0"
#   Option      "ReportingMode" "Scaled"
#   Option      "ButtonThreshold" "17"
#   Option      "ButtonNumber" "1"
#   Option      "SendCoreEvents"
# EndSection
# *****
# Monitor section
# *****
# Any number of monitor sections may be present
# *****
# Graphics device section
# *****
# Any number of graphics device sections may be present
# Standard VGA Device:
# Device configured by xorgconfig:
# *****
# Screen sections
# *****
# Any number of screen sections may be present.  Each describes
# the configuration of a single screen.  A single specific screen section
# may be specified from the X server command line with the "-screen"
# option.
# *****
# ServerLayout sections.
# *****
# Any number of ServerLayout sections may be present.  Each describes
# the way multiple screens are organised.  A specific ServerLayout
# section may be specified from the X server command line with the
# "-layout" option.  In the absence of this, the first section is used.
# When now ServerLayout section is present, the first Screen section
# is used alone.
# Section "DRI"
#   Mode 0666
# EndSection

```

Section "ServerLayout"

```
# The Identifier line must be present
# Each Screen line specifies a Screen section name, and optionally
# the relative position of other screens. The four names after
# primary screen name are the screens to the top, bottom, left and right
# of the primary screen. In this example, screen 2 is located to the
# right of screen 1.
# Each InputDevice line specifies an InputDevice section name and
# optionally some options to specify the way the device is to be
# used. Those options include "CorePointer", "CoreKeyboard" and
# "SendCoreEvents".
    Identifier      "Simple Layout"
    Screen          "Screen 1" 0 0
    InputDevice     "Mouse1" "CorePointer"
    InputDevice     "Keyboard1" "CoreKeyboard"
EndSection
```

Section "Files"

```
# The location of the RGB database. Note, this is the name of the
# file minus the extension (like ".txt" or ".db"). There is normally
# no need to change the default.
# Multiple FontPath entries are allowed (which are concatenated together),
# as well as specifying multiple comma-separated entries in one FontPath
# command (or a combination of both methods)
#
#
#   FontPath      "/usr/share/fonts/CID/"
#   FontPath      "/usr/share/fonts/Speedo/"
#   FontPath      "/usr/share/fonts/TrueType/"
#   FontPath      "/usr/share/fonts/freefont/"
# The module search path. The default path is shown here.
#   ModulePath    "/usr/X11R6/lib/modules"
    RgbPath        "/usr/X11R6/lib/X11/rgb"
    FontPath        "/usr/share/fonts/misc/"
    FontPath        "/usr/share/fonts/TTF/"
    FontPath        "/usr/share/fonts/Type1/"
    FontPath        "/usr/share/fonts/75dpi/"
    FontPath        "/usr/share/fonts/100dpi/"
    FontPath        "/usr/share/fonts/local/"
EndSection
```

```
Section "Module"

# This loads the DBE extension module.
# initialisation of the XFree86-DGA extension within that module.
# This loads the font modules
# This loads the GLX module
#   Load       "dri"
#       Load     "dbe"           # Double buffer extension
#       Load     "bitmap"
#       Load     "ddc"
#       Load     "extmod"
#       Load     "int10"
#       Load     "record"
#   Load       "speedo"
#       Load     "vbe"
# This loads the miscellaneous extensions module, and disables
#   SubSection  "extmod"
#       Option  "omit xfree86-dga"  # don't initialise the DGA extension
#   EndSubSection
#   Load       "type1"
#   Load       "speedo"
#   Load       "freetype"
#   Load       "xtt"
#       Load     "glx"
# This loads the DRI module
EndSection
```

Section "InputDevice"

```
# For most OSs the protocol can be omitted (it defaults to "Standard").
# When using XQUEUE (only for SVR3 and SVR4, but not Solaris),
# uncomment the following line.
# Option      "Protocol"      "Xqueue"
# Option      "Xleds"         "1 2 3"
# Option "LeftAlt"      "Meta"
# Option "RightAlt"     "ModeShift"
# To customise the XKB settings to suit your keyboard, modify the
# lines below (which are the defaults).  For example, for a non-U.S.
# keyboard, you will probably want to use:
# Option "XkbModel"      "pc105"
# If you have a US Microsoft Natural keyboard, you can use:
# Option "XkbModel"      "microsoft"
#
# Then to change the language, change the Layout setting.
# For example, a german layout can be obtained with:
# Option "XkbLayout"     "de"
# or:
# Option "XkbLayout"     "de"
# Option "XkbVariant"    "nodeadkeys"
#
# If you'd like to switch the positions of your capslock and
# control keys, use:
# Option "XkbOptions"     "ctrl:swapcaps"
# These are the default XKB settings for Xorg
# Option "XkbRules"       "xorg"
# Option "XkbModel"       "pc105"
# Option "XkbLayout"      "us"
# Option "XkbVariant"     ""
# Option "XkbOptions"     ""
# Option "XkbDisable"
Identifier      "Keyboard1"
Driver          "kbd"
Option         "AutoRepeat" "500 30"
# Specify which keyboard LEDs can be user-controlled (eg, with xset(1))
Option         "XkbRules" "xorg"
Option         "XkbModel" "pc105"
Option         "XkbLayout" "fr"
EndSection
```

Section "InputDevice"

```
# Identifier and driver
#   Option "Resolution"          "256"
# When using XQUEUE, comment out the above two lines, and uncomment
# the following line.
#   Option "Protocol"            "Xqueue"
# Baudrate and SampleRate are only for some Logitech mice. In
# almost every case these lines should be omitted.
#   Option "BaudRate"            "9600"
#   Option "SampleRate"          "150"
# Emulate3Buttons is an option for 2-button Microsoft mice
# Emulate3Timeout is the timeout in milliseconds (default is 50ms)
#   Option "Emulate3Buttons"
#   Option "Emulate3Timeout"     "50"
# ChordMiddle is an option for some 3-button Logitech mice
#   Option "ChordMiddle"
#       Identifier      "Mouse1"
#       Driver          "mouse"
#       Option          "Protocol" "IMPS/2"
#       Option          "Device"  "/dev/input/mice"
#       Option          "Emulate3Buttons" "true"
#       Option          "ZAxisMapping" "4 5"
# Mouse-speed setting for PS/2 mouse.
EndSection
```

Section "Monitor"

```
# HorizSync is in kHz unless units are specified.
# HorizSync may be a comma separated list of discrete values, or a
# comma separated list of ranges of values.
# NOTE: THE VALUES HERE ARE EXAMPLES ONLY. REFER TO YOUR MONITOR'S
# USER MANUAL FOR THE CORRECT NUMBERS.
#   HorizSync      30-64          # multisync
#   HorizSync      31.5, 35.2     # multiple fixed sync frequencies
#   HorizSync      15-25, 30-50   # multiple ranges of sync frequencies
# VertRefresh is in Hz unless units are specified.
# VertRefresh may be a comma separated list of discrete values, or a
# comma separated list of ranges of values.
# NOTE: THE VALUES HERE ARE EXAMPLES ONLY. REFER TO YOUR MONITOR'S
# USER MANUAL FOR THE CORRECT NUMBERS.
#       Identifier      "Iiyama Vision Master Pro 410"
#       HorizSync        27.0 - 96.0
#       VertRefresh       50.0 - 160.0
#       ModeLine "1280x1024" 135.00 1280 1328 1472 1712 1024 1033 1036 1070 +hsync +vsync
EndSection
```

Section "Device"

```
# The chipset line is optional in most cases. It can be used to override
# the driver's chipset detection, and should not normally be specified.
#   Chipset           "generic"
# The Driver line must be present. When using run-time loadable driver
# modules, this line instructs the server to load the specified driver
# module. Even when not using loadable driver modules, this line
# indicates which driver should interpret the information in this section.
# The BusID line is used to specify which of possibly multiple devices
# this section is intended for. When this line isn't present, a device
# section can only match up with the primary video device. For PCI
# devices a line like the following could be used. This line should not
# normally be included unless there is more than one video device
# intalled.
#   BusID             "PCI:0:10:0"
#   VideoRam          256
#   Clocks             25.2 28.3
#   Identifier        "Standard VGA"
#   Driver             "vga"
#   VendorName        "Unknown"
#   BoardName         "Unknown"
EndSection
```

Section "Device"

```
#nv"
# Insert Clocks lines here if appropriate
Identifier          "Nvidia Geforce 6200"
Driver              "nvidia"
#oprion twinview
Option              "TwinView"
Option              "NoTwinViewXineramaInfo"    "true"
Option              "SecondMonitorHorizSync"    "27.0-96.0"
Option              "SecondMonitorVertRefresh"  "50.0-160.0"
Option              "TwinViewOrientation"        "RightOf"
Option              "MetaModes"                  "1280x1024,1280x1024"
Option              "ConnectedMonitor"          "crt,crt"
#fin option twinview
EndSection
```

```

Section "Screen"
    Identifier      "Screen 1"
    Device          "Nvidia Geforce 6200"
    Monitor         "Iiyama Vision Master Pro 410"
    DefaultDepth    24
    Option          "NoLogo" "true"
    SubSection      "Display"
        Viewport    0 0
        Depth       8
        Modes       "1280x1024" "1024x768" "800x600" "640x480"
    EndSubSection
    SubSection      "Display"
        Viewport    0 0
        Depth       16
        Modes       "1280x1024" "1024x768" "800x600" "640x480"
    EndSubSection
    SubSection      "Display"
        Viewport    0 0
        Depth       24
        Modes       "1280x1024" "1024x768" "800x600" "640x480"
    EndSubSection
EndSection

```

```

# lspci -v | grep -i audio
0000:00:0a.0 Multimedia audio controller: Creative Labs SB Live! EMU10k1 (rev 06)

```

Une fois identifié le nom du pilote (emu10k1 dans notre exemple), éditez `/etc/make.conf` et ajoutez-y la variable `ALSA_CARDS`.

```

(Pour une seule carte de son :)
ALSA_CARDS="emu10k1"
(Pour spécifier plus d'une carte, séparez les noms avec des espaces :)
ALSA_CARDS="emu10k1 via82xx"
ALSA_CARDS="intel8x0"

```

```
# emerge alsa-driver
```

```
# emerge alsa-utils
```

```
# alsacnf
```

```
# rc-update add alsasound boot
```

```
* alsasound added to runlevel boot
```

```
* rc-update complete.
```

(Remplacez `<nom_d'utilisateur>` par le nom approprié.)

```
# gpasswd -a <nom_d'utilisateur> audio
```

```
Adding user <nom_d'utilisateur> to group audio
```

```
(Si ALSA a été compilé en modules :)
# /etc/init.d/alsasound start
* Loading ALSA modules ...
* Loading: snd-card-0 ... [ ok ]
* Loading: snd-pcm-oss ... [ ok ]
* Loading: snd-seq ... [ ok ]
* Loading: snd-emul0k1-synth ... [ ok ]
* Loading: snd-seq-midi ... [ ok ]
* Restoring Mixer Levels ... [ ok ]
(Si ALSA a été compilé dans le noyau :)
# /etc/init.d/alsasound start
* Loading ALSA modules ...
* Restoring Mixer Levels ... [ ok ]
```

(Ouvre le programme en mode console. Seuls les paramètres requis sont affichés.)

```
# alsamixer
```

```
# cat /dev/urandom > /dev/dsp
```

(Installer l'application que vous désirez :)

```
# emerge madplay mpg123 xmms
(Pour jouer des fichiers .ogg :)
# emerge vorbis-tools
```

```
# madplay -v /mnt/shyam/Music/Paul\ Oakenfold\ -\ Dread\ Rock.mp3
MPEG Audio Decoder 0.15.2 (beta) - Copyright (C) 2000-2004 Robert Leslie et al.
Title: Dread Rock
Artist: Paul Oakenfold
Album: Matrix Reloaded
Year: 2003
Genre: Soundtrack
Soundtrack
00:04:19 Layer III, 160 kbps, 44100 Hz, joint stereo (MS), no CRC
# ogg123 Paul\ Oakenfold\ -\ Dread\ Rock.ogg
Audio Device: Advanced Linux Sound Architecture (ALSA) output
Playing: Paul Oakenfold - Dread Rock.ogg
Ogg Vorbis stream: 2 channel, 44100 Hz
Genre: Soundtrack
Transcoded: mp3;160
Title: Dread Rock
Artist: Paul Oakenfold
Date: 2003
Album: Matrix Reloaded
Time: 00:11.31 [04:28.75] of 04:40.06 (200.6 kbps) Output Buffer 96.9%u
```

```
USE="-qt -kde gtk gnome hal howl"
```

Une fois cela fait, installez GNOME avec la commande **emerge** et les paquets **gnome** et **xscreensaver** :

```
# emerge gnome xscreensaver
```

Maintenant, mettez à jour vos variables d'environnement :

```
# env-update && source /etc/profile
```

```
# /etc/init.d/famd start
# rc-update add famd default
# /etc/init.d/hald start
# rc-update add hald default
# /etc/init.d/mDNSResponder start
# rc-update add mDNSResponder default
```

```
# rc-update add xdm default
```

Ensuite, éditez le fichier `/etc/rc.conf` et définissez la variable `DISPLAYMANAGER`.

```
DISPLAYMANAGER="gdm"
```

```
# emerge ntp-client
```

Ensuite vous configurez le fichier `/etc/ntp.conf` en lui spécifiant quel serveur de temps vous voulez connecter (j'ai choisi le serveur `chronos.cru.fr`).

```
# NOTES:
# - you should only have to update the server line below
# - if you start getting lines like 'restrict' and 'fudge'
#   and you didnt add them, AND you run dhcpcd on your
#   network interfaces, be sure to add '-Y -N' to the
#   dhcpcd_ethX variables in /etc/conf.d/net

# Name of the servers ntpd should sync with
# Please respect the access policy as stated by the responsible person.
#server          ntp.example.tld          iburst

#server pool.ntp.org
server chronos.cru.fr
##
# A list of available servers can be found here:
# http://www.pool.ntp.org/
# http://www.pool.ntp.org/#use
# A good way to get servers for your machine is:
# netselect -s 3 pool.ntp.org
##

# you should not need to modify the following paths
driftfile        /var/lib/ntp/ntp.drift
#server ntplocal.example.com prefer
#server timeserver.example.org

# Warning: Using default NTP settings will leave your NTP
# server accessible to all hosts on the Internet.

# If you want to deny all machines (including your own)
# from accessing the NTP server, uncomment:
#restrict default ignore

# To deny other machines from changing the
# configuration but allow localhost:
restrict default nomodify nopeer
restrict 127.0.0.1

# To allow machines within your network to synchronize
# their clocks with your server, but ensure they are
# not allowed to configure the server or used as peers
# to synchronize against, uncomment this line.
#
#restrict 192.168.0.0 mask 255.255.255.0 nomodify nopeer notrap
```

Puis on va lancer tout cela automatiquement grâce à la commande **rc-update** :

```
# rc-update add ntp-client default
```

Si vous avez recompilé les sources de firefox, il vous sera impossible d'installer et d'utiliser Flash. En effet il n'existe pas de version 64 bits de Flash. Vous devrez donc installer les binaires 32

bits de Firefox et le paquet `netscape-flash` pour pouvoir bénéficier de toutes les applications et sites web en flash.

Afin de bénéficier de l'impression via Cups, j'ai installé les paquets suivants : `cups`, `foomatic-db`, `gimp-print`, `app-text/ghostscript-esp`, `hpijs`.

Si vous désirez que votre utilisateur lambda puisse accéder aux ressources de la machine (carte son, cdrom etc...), il est nécessaire de l'ajouter à différents groupes. La commande est la suivante :

```
# gpasswd -a nom_utilisateur cdrom
# gpasswd -a nom_utilisateur audio
```

Voici mon fichier `/etc/group`

```
/etc/group
root::0:root
bin::1:root,bin,daemon
daemon::2:root,bin,daemon
sys::3:root,bin,adm
adm::4:root,adm,daemon
tty::5:
disk::6:root,adm
lp:!:7:lp,yann
mem::8:
kmem::9:
wheel::10:root,yann
floppy::11:root
mail::12:mail
news::13:news
uucp::14:uucp
man::15:man
cron::16:cron
console::17:
audio:!:18:yann
cdrom:!:19:yann
dialout::20:root
ftp::21:
sshd::22:
at::25:at
```

```
tape::26:root
video:!:27:root,yann
squid::31:squid
gdm::32:gdm
xfs::33:xfs
games:!:35:yann
named::40:named
mysql:x:60:
postgres::70:
cdrw::80:
nut::84:
usb::85:
vpopmail:x:89:
users::100:games,yann
nofiles:x:200:
qmail:x:201:
postfix:x:207:
postdrop:x:208:
smmsp:x:209:smmsp
slocate::245:
portage::250:portage
utmp:x:406:
nogroup::65533:
nobody::65534:
messagebus:x:407:
haldaemon:x:408:
rpc:x:111:
ntp:x:123:
```

La commande `rc-update show` permet de voir tout ce qui est lancer au démarrage.

```

biboo etc # rc-update show
      alsasound | boot
      bootmisc | boot
      checkfs   | boot
      checkroot | boot
      clock     | boot
      coldplug  |      default
      consolefont | boot
      crypto-loop |
      cupsd     |      default
      dbus      |
      domainname | boot default
      esound    |
      famd      |      default
      gpm       |      default
      hald      |      default
      hdparm    |
      hostname  | boot
      hotplug   |      default
      keymaps   | boot
      local     |      default nonetwork
      localmount | boot
      mDNSResponder |      default
      modules   | boot
      mysql     |
      net.eth0  |      default
      net.lo    | boot
      netmount  |      default
      nifd      |
      nscd      |
      ntp-client |      default
      ntpd      |
      numlock   |      default
      portmap   |
      rmnologin | boot
      rsyncd    |
      spamd     |
      sshd      |      default
      syslog-ng |      default
      urandom   | boot
      vixie-cron |      default
      xdm       |      default
biboo etc #

```

17 Conclusion

C'en est fini de cet article, toutes remarques et corrections sont les bienvenues à l'adresse morere@univ-metz.fr