Petit guide du logiciel Jigdo de Débian

Adaptation française du *Debian Jigdo mini-HOWTO*

Peter Jay Salzman `<p@dirac.org>`

Adaptation française: Philippe Lhardy

`<philippelhardy CHEZ chez POINT com>`

Relecture de la version française: Claude Thomassin

`<cthomassin CHEZ waika9 POINT com>`

Préparation de la publication de la v.f.: Jean-Philippe Guérard

`<jean TIRET philippe POINT guerard CHEZ laposte POINT net>`

v1.1.fr.1.0

Le document source servant à créer les différents formats de ce guide pratique est disponible sur le site du projet traduc.org [http://ftp.traduc.org/doc-vf/HOWTO/telechargement/sgml/].

2004-02-28

Historique des versions

Version 1.1.fr.1.0	2004-02-28	PL, CT, JPG
	Première adaptation française	
Version 1.1	2004-01-08	PS
Version 1.0	2003-10-31	PS

Récupérer les images Iso des CD-ROM de la distribution Débian a toujours été un processus pénible, lent et suprêmement inefficace. Jigdo est un nouvel outil permettant de récupérer les images Iso Débian d'une façon simple, rapide et très efficace. Ce guide pratique vous expliquera pourquoi vous devez utiliser Jigdo, et présentera également son fonctionnement et son utilisation pour récupérer et mettre à jour des images Iso Débian.

Jigdo est un outil très général qui n'est pas réservé aux images Iso Débian. Les outils Jigdo peuvent être utilisés pour rendre n'importe quelle image Iso disponible au téléchargement d'une manière tout aussi simple, rapide et efficace. Ce guide pratique se concentrera principalement sur l'utilisation de Jigdo pour télécharger des images Iso Débian, mais abordera également son usage plus général.

Table des matières

1. Paperasse

1.1. Auteur et droits d'utilisation

Si vous désirez créer un travail dérivant de ce guide pratique ou bien publier ce guide pratique à des fins commerciales, j'apprécierais que vous me contactiez d'abord. Cela me donnera une chance de vous en fournir la version la plus récente. J'apprécierais également soit une copie de ce que vous faites, soit une pizza épinard, ail, champignon, féta et cœur d'artichaut.

If you want to create a derivative work or publish this HOWTO for commercial purposes, I would appreciate it if you contact me first. This will give me a chance to give you the most recent version. I'd also appreciate either a copy of whatever it is you're doing or a spinach, garlic, mushroom, feta cheese and artichoke heart pizza.

Le document source servant à créer les différents formats de ce guide pratique est disponible sur le site du projet traduc.org [http://ftp.traduc.org/doc-vf/HOWTO/telechargement/sgml/].

1.2. Remerciements

Initialement, je voulais simplement remercier l'auteur de Jigdo, Richard Atterer `<atterer CHEZ debian POINT org>`, d'avoir écrit Jigdo. Toute personne ayant obtenu des images Iso Débian par une autre méthode comprendra pourquoi. Cependant, mes remerciements se doivent d'aller plus loin. Ce guide pratique a démarré à partir de quelques pages web que j'avais rédigés sur mon expérience de Jigdo. Richard a pris le temps de m'envoyer par courrier électronique des corrections, des clarifications et des réponses très complètes à mes questions sur Jigdo. Depuis lors, il a relu mon travail plusieurs fois. Richard est un développeur qui ne se contente pas de s'occuper de son travail, mais qui s'intéresse aussi aux gens qui l'utilisent. C'est triste, mais c'est devenu de moins en moins courant dans le monde industrieux où nous vivons. Merci Richard !

Je désirerais remercier également Conrad Wood `<cnw CHEZ conradwood POINT net>`, Elcio Mello `<mello CHEZ ajato POINT com POINT br>`, Marcelo Ramos `<mramos CHEZ montevideo POINT com POINT uy>`, Yufeng Wang `<ipzh CHEZ 163 POINT net>` et Tsukasa Yamaguchi `<tyamagch CHEZ bd POINT mbn POINT or POINT jp>` pour avoir traduit ce petit guide. Je me sens vraiment honoré qu'ils aient trouvé que mes mots méritaient leur temps et leurs efforts. Merci, les gars !

Pour finir je voudrais remercier Mark van Lent `<mark CHEZ panic POINT et POINT tudelft POINT nl>` pour ses paroles pleines de gentillesse et ses corrections si nécessaire.

1.3. Commentaires et corrections

Je porte beaucoup d'attention aux personnes qui utilisent ce document. L'écriture, même de petits guides, demande beaucoup de temps, et je ne voudrais pas avoir investi autant d'efforts dans quelque chose que personne ne comprenne. Si vous avez des commentaires, des corrections ou bien des suggestions, même relatives à mon style dans la version originale, n'hésitez pas à m'envoyer un courrier électronique. Aussi longtemps que je ne serai pas totalement englué dans ma dissertation de doctorat et dans le livre que j'écris sur le débogage de code avec GDB et DDD pour No Starch Press, je ferai de mon mieux pour répondre à chacun des courriers que je recevrai [en anglais] à propos de ce petit guide.

N'hésitez pas à faire parvenir tout commentaire relatif à la version française de ce document à `<commentaires CHEZ traduc POINT org>` en précisant le titre et la version de ce document.

1.4. Dernières versions et traductions

Allemand	Conrad Wood <cnw CHEZ conradwood POINT net>.
Portugais	Elcio Mello <mello CHEZ ajato POINT com POINT br>.
Espagnol	Marcelo Ramos <mramos CHEZ montevideo POINT com POINT uy>.
Chinois	Yufeng Wang <ipzh CHEZ 163 POINT net>.
Japonais	Tsukasa Yamaguchi <tyamagch CHEZ bd POINT mbn POINT or POINT jp>. Disponible sur http://www.linux.or.jp/JF/JFdocs/Debian-Jigdo.

La version anglaise et toutes ses traductions sont disponibles sur mon site web : http://www.dirac.org/linux/debian/jigdo. Si vous désirez traduire ce petit guide dans une autre langue, je vous remercie de me contacter [en anglais] à l'adresse <p CHEZ dirac POINT org>.

La version anglaise stable de ce document est disponible sur le site du Projet de documentation Linux (LDP) http://tldp.org/docs.html dans la section « *howto* ». Si vous désirez voir le travail en cours, vous trouverez la toute dernière version de ce document sur http://www.dirac.org/linux/debian/jigdo.

La dernière version française [http://www.traduc.org/docs/howto/lecture/Debian-Jigdo.html] de ce document est disponible sur le site du projet traduc.org [http://www.traduc.org].

2. Pourquoi Jigdo?

2.1. Comment obtenir un jeu d'images Iso Débian ?

Si vous désirez votre propre jeu de CD Débian, il y a de multiples façons de procéder. Vous pouvez les acheter depuis la liste des sociétés [http://www.debian.org/CD/vendors/] qui distribuent des CD Débian. Cela vaut vraiment la peine car ces sociétés reversent des subsides au projet Débian. Vos dons contribuent donc à pérenniser ce projet.

Un autre moyen d'obtenir un jeu de CD Débian est de les graver vous-même. Pour cela, il faut d'abord obtenir l'image Iso puis la graver sur un CD-ROM vierge. Avant Jigdo, il était possible de créer des CD Débian de deux façons différentes :

1. Télécharger des images Iso
2. Utiliser un kit pseudo-image (PIK)

Ce document présente un outil appelé Jigdo, qui est la nouvelle meilleure façon d'obtenir des images Iso Débian. En fait, le kit pseudo-image est aujourd'hui officiellement mort et les parties le concernant ont été supprimées de ce document. Jigdo est la méthode canonique pour obtenir une image Iso Débian.

2.2. Pourquoi ne pas télécharger l'image entière ?

Il existe des sites miroirs qui permettent de télécharger des images Iso Débian via les protocoles HTTP et FTP. Le problème est qu'il y a très peu de sites miroirs et que leur bande passante n'est pas suffisante pour tous ceux qui désirent télécharger des images Iso Débian. Par exemple, il a été rapporté que le site `fsn.hu` avait saturé la connexion de son fournisseur d'accès. Le débit sortant atteint quelques téraoctets par mois !

De plus, les distributions *testing* et instable de Débian sont fréquemment mises à jour. Vos images

Iso risquent d'être déjà obsolètes le jour même de leur téléchargement, à moins que vous ne trouviez quelque moyen tordu de les mettre à jour, par exemple en les montant sur un périphérique de bouclage (*loopback*) et en utilisant rsync (ce que faisait le kit pseudo image). Donc si vous désirez des images à jour, vous devrez télécharger un nouveau jeu de CD chaque jour. Clairement, ce n'est pas la bonne façon de procéder !

Même si vous désirez télécharger une image Iso de la distribution stable Débian, ces images seront tout de même mises à jour après quelques mois. Télécharger des images Iso vous donnera donc des images à jour pour seulement quelques mois. À chaque sortie d'une nouvelle révision de la distribution stable Débian, vous devrez recommencer à zéro le pénible processus de téléchargement de l'ensemble des images. Ça n'est ni une bonne utilisation de votre temps, ni des ressources du miroir.

2.3. Qu'est-ce que Jigdo ?

Jigdo (qui signifie « *Jigsaw Download* » c'est-à-dire « téléchargement pièce par pièce ») a été écrit par Richard Atterer `<atterer CHEZ debian POINT org>` et est diffusé sous la licence publique générale GNU (GPL). C'est un outil qui permet le téléchargement et la mise à jour efficace des images Iso. De n'importe quelle image. Jigdo n'est pas spécifique à Débian. Débian l'a cependant choisi comme méthode officielle de téléchargement pour ses images Iso.

Une erreur très répandue est de croire que Jigdo crée des images Iso : il ne le fait pas. Examinons la façon de procéder de Jigdo pour obtenir une image Iso. Appelons Alain la personne qui diffuse l'image Iso (peut-être est-il le responsable de la publication des nouvelles versions de la distribution Débian). Et Béatrice, la personne qui veut télécharger l'image Iso (c'est peut-être une utilisatrice de la distribution Débian).

1. Tout d'abord Alain crée une image Iso pouvant être gravée sur CD-ROM. Pour ceci, il peut utiliser un utilitaire tel que mkisofs ou debian-cd. Il crée aussi deux fichiers associés à sa nouvelle image Iso : un fichier `.jigdo` et un fichier `.template`. Il ne lui reste plus qu'a diffuser (en téléchargement) ces deux fichiers, pour tous ceux qui désirent obtenir son image Iso.
2. La seconde étape est réalisée par Béatrice. Elle consiste à télécharger les fichiers `.jigdo` et `.template`. Puis a utiliser le programme jigdo-lite pour télécharger l'image Iso d'Alain à partir de ces deux fichiers…

L'outil Jigdo contient deux utilitaires : jigdo-file et jigdo-lite. jigdo-file sera utilisé par Alain pour créer les fichiers `.template` et `.jigdo` à partir de son image Iso. Jigdo-lite sera utilisé par Béatrice pour télécharger l'image en utilisant les fichiers `.jigdo` et `.template`. Si la seule chose que vous désiriez est de télécharger des images Iso Débian, vous n'aurez besoin que de jigdo-lite. Vous pouvez même oublier que jigdo-file existe `:-)`

Jigdo permet de s'affranchir des problèmes des autres méthodes de téléchargement d'images Iso Débian :

- Utiliser Jigdo est beaucoup plus rapide que de télécharger l'image Iso en entier ;
- Contrairement au téléchargement d'une image Iso entière, Jigdo peut utiliser un CD périmé (ou son image Iso montée en boucle) et télécharger *uniquement* les fichiers qui ont changés depuis la date de création du CD (de l'image Iso) afin de créer une image Iso à jour. Tout comme cvs lorsqu'il met à jour du code source ;
- jigdo-lite utilise wget qui, par défaut, utilise le protocole HTTP pour transférer les fichiers. Contrairement à rsync, HTTP n'est jamais bloqué par les pare-feu (sauf ceux derrière lesquels vous ne devriez pas envisager d'utiliser Jigdo).
- Jigdo est très respectueux de la bande passante des serveurs diffusant des images Débian. Les miroirs Débian peuvent accueillir un plus grand nombre d'internautes s'ils utilisent Jigdo plutôt que d'autres méthodes de téléchargement…

Jigdo est sans l'ombre d'un doute la meilleure façon d'obtenir des images Iso Débian.

3. Comment Jigdo fonctionne (optionnel)

Il n'est pas nécessaire de lire cette cette section pour utiliser Jigdo, mais cela peut vous aider à en démystifier le fonctionnement. Si vous n'êtes pas intéressé par les détails, allez simplement à la Comment installer Jigdo.

3.1. Préparation de l'image Iso pour le téléchargement

Une image CD est un système de fichiers appelé iso9660. Pour cette discussion, nous pouvons sans risque parler d'une image CD comme étant un gros fichier (de près de 650 Mo) appelé « image Iso » qui contient des fichiers à des positions différentes. Par exemple, si un CD contient un fichier de 567 octets nommé `LISEZ.MOI`, l'image Iso peut inclure le contenu de ce fichier entre les positions (*offset*) 20480000 et 20480566. Vous pouvez vous représenter une image CD comme ceci :

```
              ------------------------------------------------
Image Iso :   |xx| fichier-0 |x| fichier-1 |x| fichier-2 |xx|
              ------------------------------------------------
```

Les zones avec des « x » de l'image contiennent des choses telles que le contenu des répertoires, du remplissage par des zéro, le nom du disque, le bloc d'amorçage, et cætera.

jigdo-file utilise deux choses en entrée : l'image complète du CD (donc l'image Iso doit avoir été faite) et un ensemble de fichiers qui peuvent se trouver ou non dans l'image. Voici une représentation de ce qui est fourni en entrée à jigdo-file :

```
Image  ------------------------------------------------
 Iso : |xx| fichier-0 |x| fichier-1 |x| fichier-2 |xx|
       ------------------------------------------------
Fichiers  ------------- -------------                ------------- 
 libres : | fichier-0 | | fichier-1 |                | fichier-3 |
          ------------- -------------                ------------- 
```

Comme par magie, jigo-file retrouvera quels fichiers libres sont contenus dans l'image Iso et leur position (*offset*) à l'intérieur du fichier Iso. Il générera deux fichiers : un fichier « `.template` » et un fichier « `.jigdo` ».

3.2. Le fichier `.template`

Partant d'une image Iso et d'un ensemble de fichiers (contenus ou non dans l'image Iso), jigdo-file générera un fichier `.template` pour cette image Iso. Voici à quoi ressemble ce fichier `.template` :

```
             ------------------------------------------------
.template :  |xx|  md5-0    |x|  md5-1    |x|ccccccccccc|xx|
             ------------------------------------------------
```

jigdo-file a déterminé que les fichiers `fichier-0` et `fichier-1` étaient contenus dans l'image Iso. Il a supprimé le contenu de ces fichiers et l'a remplacé par la somme de vérification md5 de chaque fichier (`md5-0` et `md5-1`).

Les données représentées par des « `x` » (contenu des répertoires, remplissage par zéro, et cætera) contenues dans l'image Iso sont compressées et écrites dans le fichier `.template`. Pour terminer, tous les fichiers contenus dans l'image Iso mais non fournis parmi les fichiers libres (tel que `fichier-2`) sont également compressés et écrits dans le fichier `.template`. Ceci est représenté par les « `c` » dans la représentation du fichier `.template`.

Les fichiers libres qui étaient fournis à jigdo-file et qui n'ont pas été trouvés dans l'image Iso (tel `fichier-3`) sont ignorés.

3.3. Le fichier `.jigdo`

À partir d'une image Iso et d'un ensemble de fichiers qui peuvent se trouver ou non dans l'image Iso, jigdo-file génère un fichier `.jigdo` correspondant à cette image Iso. Les fichiers `.jigdo` Débian sont compressés avec **gzip**, vous devrez donc utiliser **zcat** ou **zless** pour visualiser leur contenu. Voici à quoi ressemble ce fichier une fois décompressé avec gunzip :

```
md5-0=http://un_miroir.org/fichier-0
md5-1=http://un_miroir.org/fichier-1
md5-2=http://un_miroir.org/fichier-2
```

Le fichier `.jigdo` fournit tout simplement une correspondance entre la somme de contrôle md5 du fichier contenu dans l'image Iso et l'URL où télécharger ce fichier. Il y a d'autres choses à l'intérieur du fichier `.jigdo` si vous y regardez de plus près, vous verrez que le fichier `.jigdo` a le même format qu'un fichier « .ini ». Cela doit être compréhensible en soi, mais si vous désirez plus de détails, reportez-vous à la documentation de jigdo.

Le format montré ci-dessus n'est pas ce que vous verriez effectivement dans un fichier `.jigdo` normal, mais cela devrait fortement y ressembler. Si vous jetez un coup d'œil à la section [Servers] en bas du fichier `.jigdo`, vous verrez précisément ce qui diffère entre ce que j'ai montré ici et un vrai fichier `.jigdo`.

3.4. Télécharger l'image

Après avoir utilisé jigdo-file pour générer les fichiers `.jigdo` et `.template` à partir d'une image Iso, n'importe qui peut utiliser jigdo-lite pour télécharger cette image. jigdo-lite utilise wget pour télécharger tous les fichiers inclus dans l'image Iso, les rassemble et crée à la volée une copie de l'image Iso originale.

4. Télécharger votre première image Iso (en 5 étapes faciles)

Nous considérons ici que vous partez de zéro et que vous ne disposez d'aucune image Iso Débian. Une fois gravé votre jeu d'images Iso, vous pourrez ultérieurement utiliser jigdo-lite pour les mettre à jour. Nous couvrirons la mise à jour de vos images Iso dans la section suivante.

4.1. Installer Jigdo

Premièrement installez le paquet *jigdo-file* :

```
# apt-get install jigdo-file
```

Jigdo est en plein développement. Les corrections et les améliorations sont constantes. En conséquence, si vous utilisez les distributions Débian stable ou *testing*, récupérez *jigdo-file* depuis la distribution instable http://packages.debian.org/unstable/utils/jigdo-file.html. Au 28 février 2003 c'était la version 0.6.9. C'est cette version qui est utilisée pour les exemples de ce guide pratique.

Note aux utilisateurs de la distribution Woody (stable) : la version de jigdo-lite (révision 1) qui est incluse dans la distribution Woody n'est pas capable de télécharger les distributions Sarge (*testing*) et Sid (instable). Voir la Section 6.11, « [10 février 2003] : J'ai des difficultés à télécharger la distribution Sarge ou Sid avec Jigdo ». Une version corrigée de jigdo 0.6.5 a été soumise, afin de devenir une future version 0.6.5 révision 2.

4.2. Télécharger les fichiers `.template` et `.jigdo`

Pour chacune des images Iso que vous désirez récupérer, vous aurez besoins des deux fichiers `.jigdo` et `.template` de cette image. Les noms des deux fichiers suivent la même convention d'écriture :

```
distro-arch-n.jigdo
distro-arch-n.template
```

où *`distro`* est le nom de la distribution (par exemple « `sarge` »), *`arch`* est l'architecture de votre système (par exemple « `i386` ») et *`n`* est le numéro du disque (comme par exemple « `1` »).

Par exemple, sarge a 8 images, vous devrez donc télécharger 8 fichiers `.jigdo` et 8 fichiers `.template`. Ils peuvent être téléchargés depuis http://www.debian.org/CD/jigdo-cd/. Les premiers fichiers `.jigdo` et `.template` file sont nommés `sarge-i386-1.jigdo` et `sarge-i386-1.template` respectivement.

4.3. Lancer jigdo-lite

Lancez jigdo-lite et indiquez-lui le fichier `.jigdo` de l'image que vous désirez télécharger. Prenons par exemple la distribution Woody :

```
lucifer$ ls
sarge-i386-1.jigdo  sarge-i386-1.template
lucifer$ jigdo-lite sarge-i386-1.jigdo

Jigsaw Download "lite"
Copyright 2001-2003 by Richard Atterer <jigdo@atterer.net>
Getting mirror information from /etc/apt/sources.list

-----------------------------------------------------------------
Images offered by `sarge-i386-1.jigdo':
1: 'Debian GNU/Linux testing "Sarge" - Official Snapshot i386 Binary-1 CD' (sarge-

Further information about `sarge-i386-1.iso':
Generated on Fri, 7 Feb 2003 20:31:28 -0700

-----------------------------------------------------------------
If you already have a previous version of the CD you are
downloading, jigdo can re-use files on the old CD that are also
present in the new image, and you do not need to download them
again. Mount the old CD ROM and enter the path it is mounted under
(e.g. `/mnt/cdrom').
Alternatively, just press enter if you want to start downloading
the remaining files.
Files to scan:
```

Si vous suspendiez jigdo-lite avec **Ctrl-z** (ne le faites pas, je vais vous dire ce que vous verriez) et regardiez le résultat d'un commande **ls**, vous trouveriez dans le répertoire un nouveau fichier nommé `sarge-i386-1.jigdo.unpacked`. En effet, les fichiers `.jigdo` sont compressés avec gzip. Ce fichier est simplement la version décompressée du fichier `.jigdo`.

Là où nous en sommes arrivé, jigdo-lite nous explique que, si nous disposons d'une version antérieure du premier CD de la distribution Sarge, nous devrions lui indiquer le chemin vers ce CD. C'est de cette façon qu'il est possible de mettre à jour une image Iso (ou de terminer un téléchargement incomplet). Comme nous supposons que vous partez de zéro et que vous ne disposez d'aucune image Iso Débian, vous n'avez rien à indiquer ici. Nous verrons cela dans la Section 5, « Mettre à jour votre image », donc contentez-vous d'appuyer sur **ENTRÉE**.

4.4. Indiquer un site miroir

Vous allez voir :

```
-----------------------------------------------------------------
The jigdo file refers to files stored on Debian mirrors. Please
choose a Debian mirror as follows: Either enter a complete URL
pointing to a mirror (in the form
`ftp://ftp.debian.org/debian/'), or enter any regular expression
for searching through the list of mirrors: Try a two-letter
country code such as `de', or a country name like `United
States', or a server name like `sunsite'.
Debian mirror [http://linux.csua.berkeley.edu/debian/]:
```

jigdo-lite est assez malin pour utiliser le même site miroir que celui que vous avez défini pour les mises à jour Débian en le récupérant dans le fichier `/etc/apt/sources.list`1. Si vous désirez utiliser un autre site miroir, vous pouvez le faire ici en suivant ces instructions. Si le miroir indiqué est celui que vous désirez, appuyez sur **ENTRÉE**. Jigdo-lite écrira alors un fichier `~/.jigdo-lite` dans votre répertoire personnel.

Ensuite, si le fichier `.jigdo` que vous utilisez mentionne un paquet qui doit être téléchargé depuis un serveur Non-US2, jigdo-lite va vous demander d'indiquer un miroir Débian Non-US. Le message affiché (et votre réponse) seront très similaires au dialogue qui a été décrit au paragraphe précédant (mais cette fois pour un miroir Non-US `(-:`).

```
-----------------------------------------------------------------
The jigdo file also refers to the Non-US section of the Debian
archive. Please repeat the mirror selection for Non-US. Do not
simply copy the URL you entered above; this does not work because
the path on the servers differs!
Debian non-US mirror [http://linux.csua.berkeley.edu/debian-non-US//]:
```

Jigdo-lite va enregistrer votre choix dans le fichier `~/.jigdo-lite`. Cependant, si l'image que vous êtes sur le point de télécharger ne contient pas de logiciels non-US, cette question ne vous sera pas posée.

Si, ultérieurement, vous désirez changer les miroirs par défaut utilisés par Jigdo, il suffira de modifier les deux lignes suivantes :

```
debianMirror='http://un-miroir-a-utiliser/debian/'
nonusMirror='http://un-autre-miroir/debian-non-US/'
```

dans le fichier `~/.jigdo-lite`.

4.5. Téléchargement de l'image Iso

Lorsque vous aurez indiqué à jigdo-lite les sites miroirs, jigdo-lite va commencer à télécharger les fichiers pour reconstruire l'image Iso :

```
Not downloading .template file - `sarge-i386-1.template' already present

-----------------------------------------------------------------
Merging parts from `file:' URIs, if any...
Found 0 of the 826 files required by the template
Will not create image or temporary file - try again with different input files
--09:35:12--  http://linux.csua.berkeley.edu/debian/pool/main/p/pack/pack_3.10-1_i
    => `sarge-i386-1.iso.tmpdir/linux.csua.berkeley.edu/debian/pool/main/p/pack/pa
Resolving linux.csua.berkeley.edu... done.
Connecting to linux.csua.berkeley.edu[128.32.112.231]:80... connected.
```

1vous avez sans doute remarqué que jigdo-lite utilisait le fichier `/etc/apt/sources.list` dans la capture d'écran précédente.
2Les serveurs non-US sont les serveurs Débian situés hors des États-Unis d'Amérique afin de pouvoir distribuer des outils dont l'exportation depuis les États-Unis est illégale, notamment les outils de chiffrement.

```
HTTP request sent, awaiting response... 200 OK
Length: 1,911,624 [application/x-debian-package]

19% [======>                              ] 378,304      149.87K/s    ETA 00:09
```

De nombreux messages vont défiler sur votre écran ; si c'est très perturbant pour vous, reportez-vous à la Section 6.12, « Jigdo-lite est trop bavard. Comment puis-je supprimer tous ces messages ? ». Pendant que jigdo-lite télécharge les paquets, basculez vers une autre console (ou bien ouvrez un autre **xterm**) et lancez un **ls** dans le répertoire d'où vous exécutez jigdo-lite. Il devrait maintenant y avoir 6 fichiers dans ce répertoire :

- `sarge-i386-1.iso.list`
- `sarge-i386-1.iso.tmp`
- `jigdo-file-cache.db`
- `sarge-i386-1.iso.tmpdir/`
- `sarge-i386-1.jigdo`
- `sarge-i386-1.jigdo.unpacked`
- `sarge-i386-1.template`

Le répertoire `sarge-i386-1.iso.tmpdir/` contient tous les paquets Débian téléchargés par Jigdo. Assez régulièrement, ce répertoire est vidé et les fichiers qu'il contient sont écrits dans `sarge-i386-1.iso.tmp`. Ce fichier est une version temporaire incomplète de l'image Iso que vous êtes en train de récupérer. Notez que `sarge-i386-1.iso.tmp` n'apparaîtra qu'à partir de la première fois où `sarge-i386-1.iso.tmpdir/` aura été vidé.

`jigdo-file-cache.db` est un fichier de base de données Berkeley (format *Berkeley DB*) contenant une somme de contrôle md5 pour chacun des fichiers lus dans les répertoires que vous avez indiqués à la question `Files to scan:`. Il est décrit dans la Section 7.3, « jigdo-file-cache.db ».

Lorsque vous en serez arrivés là, allez faire une partie de Quake III, car cela va prendre un certain temps. Et vous feriez mieux de jouer sur une autre machine car Jigdo utilise le disque de manière intensive lorsqu'il vide les fichiers dans le fichier `.iso.tmp`. Au bout d'un moment, le téléchargement se terminera et vous verrez ceci :

```
FINISHED --13:32:58--
Downloaded: 7,469,872 bytes in 9 files
Found 9 of the 9 files required by the template
Successfully created ‘sarge-i386-3.raw'

-----------------------------------------------------------------
Finished!
The fact that you got this far is a strong indication that ‘sarge-i386-3.raw'
was generated correctly. I will perform an additional, final check,
which you can interrupt safely with Ctrl-C if you do not want to wait.

OK: Checksums match, image is good!
```

Ce qui signifie que jigdo-lite a fini de créer l'image. Après vous avoir proposé d'arrêter là (avec **Ctrl-c**), il a réalisé une vérification supplémentaire pour s'assurer que l'image était correcte. Pour cela, il a calculé la somme de contrôle de l'image et vérifié qu'elle correspondait bien à la valeur attendue. L'image téléchargée est donc valide.

5. Mettre à jour votre image

Je suppose que vous avez lu la section précédente, suivi ses instructions, gravé vos fichiers Iso nouvellement créés sur des CD et que vous vous sentez maintenant calme et détendu. Cependant, tôt ou tard, certains paquets seront mis à jour. Vous déciderez alors peut-être de donner vos CD à un débutant lors d'un atelier d'installation[3] de votre GUL[4] local et de vous graver un nouveau jeu de CD. Puisque vous êtes en bonne voie pour devenir un gourou de Jigdo, nous n'entrerons pas autant dans les détails que précédemment.

La première chose à faire est de télécharger à nouveau les fichiers `.jigdo` et `.template` pour les images que vous désirez mettre à jour. Vous vous demandez sans doute pourquoi vous devez les télécharger à nouveau. La raison en est que les images que vous désirez charger ont changé. Des fichiers ont pu être ajoutés ou supprimés et, même si cela n'est pas le cas, les sommes de contrôles contenues dans les fichiers `.jigdo` et `.template` auront changées.

Arrivé là, vous aurez en main un CD Débian périmé ou son image Iso sur votre disque. Continuons à suivre les étapes de récupération d'une image Iso à jour. Si vous avez un CD, mettez-le dans votre lecteur de CD et montez-le :

```
$ mount /cdrom
```

Si au lieu d'un CD vous avez un fichier Iso que vous désirez mettre à jour, montez-le en boucle (`loop`). Vous devez être sous le compte `root` pour ce faire.

À titre d'exemple, nous allons réaliser la mise à jour de mon image de la distribution Woody. En effet, j'ai remarqué que des mises à jour de sécurité avaient été publiées pour cette distribution :

```
# mount -o loop woody-i386-1.iso /mnt
```

Maintenant lancez jigdo-lite avec en argument le nom du fichier `.jigdo` :

```
$ jigdo-lite woody-i386-1.jigdo

-----------------------------------------------------------------
Jigsaw Download "lite"
Copyright 2001-2002 by Richard Atterer <jigdo@atterer.net>
Loading settings from `/home/p/.jigdo-lite'

-----------------------------------------------------------------
Images offered by `woody-i386-1.jigdo':
  1: Debian GNU/Linux 3.0 r0 Woody - Official i386 Binary-1 CD (debian-30r0-i386-b

Further information about `debian-30r0-i386-binary-1.iso':
Generated on Thu, 18 Jul 2002 14:34:12 +0100

-----------------------------------------------------------------
If you already have a previous version of the CD you are
downloading, jigdo can re-use files on the old CD that are also
present on the new image, and you do not need to download them
again.  You found the secret message; you're a very careful
reader.  Mount the old CD ROM and enter the path it is mounted
under (e.g. `/mnt/cdrom'). Alternatively, just press enter if you
want to start the download of any remaining files.

You can also enter a single digit from the list below to
select the respective entry for scanning:
  1: /mnt
Files to scan:
```

3 *Install party*.
4 Groupe d'utilisateurs de Linux.

Jigdo-lite vous demande de lui indiquer le point de montage de votre CD (si vous mettez à jour un CD) ou de votre image Iso montée en boucle si vous utilisez un fichier Iso). J'utilise un fichier Iso monté en boucle sur `/mnt`, donc je vais saisir `/mnt`. Si vous mettez à jour un CD, indiquez le point de montage de votre CD, qui sera probablement `/cdrom`. Dans tous les cas, jigdo-lite va parcourir le répertoire de votre périphérique monté, déterminer quels sont les fichiers qui nécessitent une mise à jour et réutiliser les fichiers qui n'en ont pas besoin. Vous verrez certainement un écran du genre :

```
Files to scan: /mnt/other

Not downloading .template file - `woody-i386-1.template' already present
jigdo-file: Output file `debian-30r0-i386-binary-1.iso' already exists - delete
it or use --force
jigdo-file failed with code 3 - aborting.
```

Que s'est-il passé ? En fait, je voulais vous le montrer car vous y serez confronté tôt ou tard. Je met à jour un fichier Iso, mais le fichier périmé utilisé est dans le répertoire où je travaille. Jigdo-lite veut générer un fichier nommé `woody-i386-1.iso` mais il y a déjà un fichier de ce nom dans le répertoire courant (l'image périmée). Jigdo-lite ne peut pas détruire ce fichier, donc il s'arrête et prévient que je peux soit détruire ce fichier ou utiliser l'option `--force` pour l'écraser. Vous pouvez aussi déplacer ou renommer le fichier, mais je pense que jigdo-lite considère que vous le savez déjà. :-)

Ne soyez pas intimidé par l'idée de déplacer ou de renommer le fichier image du simple fait qu'il soit monté en boucle. Le système de fichiers utilise des inœuds (*inodes*). Même si vous déplacez ou renommez le fichier, l'inœud restera le même et le système de fichiers sous `/mnt` ne sera pas endommagé. De même, supprimer le fichier Iso n'abîmera pas le système de fichiers qui a été monté. L'inœud du fichier ne sera désallouée que lorsque son compteur de références arrivera à zéro. Monter le fichier Iso incrémente son compteur de références de un, donc le fichier ne sera réellement détruit que lorsque vous aurez exécuté la commande **rm** sur le fichier *et* démonté le périphérique de bouclage. Tous ceux qui mettent à jour des CD n'ont pas à s'inquiéter de cela. :-)

Je vais donc renommer le fichier Iso en `woody-i386-1.iso.old` et relancer jigdo-lite. C'est reparti :

```
$ jigdo-lite woody-i386-1.jigdo

-----------------------------------------------------------------
Jigsaw Download "lite"
Copyright 2001-2002 by Richard Atterer <jigdo@atterer.net>
Loading settings from `/home/p/.jigdo-lite'

-----------------------------------------------------------------
Images offered by `woody-i386-1.jigdo':
  1: Debian GNU/Linux 3.0 r0 Woody - Official i386 Binary-1 CD
       (debian-30r0-i386-binary-1.iso)

Further information about `debian-30r0-i386-binary-1.iso':
Generated on Thu, 18 Jul 2002 14:34:12 +0100

-----------------------------------------------------------------
If you already have a previous version of the image you are
downloading, jigdo can re-use files on the old image that are also
present on the new image, and you do not need to download them
again. Mount the old CD ROM and enter the path it is mounted under
(e.g. `/mnt/cdrom'). Alternatively, just press enter if you want
to start the download of any remaining files.
You can also enter a single digit from the list below to
select the respective entry for scanning:
  1: /mnt
Files to scan: /mnt
Not downloading .template file - `woody-i386-1.template' already present
…
Found 1200 of the 1224 files required by the template
```

```
...
```

jigdo-lite se rappelle que je voulais parcourir le répertoire `/mnt` et m'indique que je peux soit entrer un `1` pour parcourir ce répertoire, soit entrer le nom d'un nouveau répertoire à parcourir. Comme je suis retors, j'indique une nouvelle fois le nom du répertoire.

Les points de suspension représentent du texte qui bouge rapidement. Le premier « … » est la liste dynamique des fichiers que jigdo-lite est en train de parcourir. Le second indique la progression de l'écriture de `woody-i386-1.iso.tmp`. Une fois que jigdo-lite a terminé le parcours des fichiers et l'écriture dans l'image Iso temporaire il affiche :

```
Copied input files to temporary file `woody-i386-1.iso.tmp'
  - repeat command and supply more files to continue

-----------------------------------------------------------------
If you already have a previous version of the image you are
downloading, jigdo can re-use files on the old image that are also
present on the new image, and you do not need to download them
again. Mount the old CD ROM and enter the path it is mounted under
(e.g. `/mnt/cdrom'). Alternatively, just press enter if you want
to start the download of any remaining files.
You can also enter a single digit from the list below to
select the respective entry for scanning:
  1: /mnt
Files to scan:
```

Puisque, en théorie, vous n'avez pas d'autre source de fichiers à parcourir, hormis l'image Iso que vous avez montée (ou votre CD), appuyez sur **ENTRÉE**. Jigdo-lite va ensuite vous demander quels miroirs vous voulez utiliser, tout comme il l'a fait la première fois que vous avez téléchargé votre fichier Iso. Vous avez déjà répondu auparavant à ces questions. Si vous ne vous en souvenez vraiment plus, vous devriez certainement relire la Section 4.4, « Indiquer un site miroir ».

À partir de là, vous verrez jigdo-lite accomplir sa magie. Maintenant, est-ce que ce n'était pas simple ?

6. Foire aux questions (FAQ)

Les questions suivies d'une date sont des questions relatives à une situation temporaire. Si vous apprenez que la situation temporaire relatée dans l'une de ces questions a changé, je vous remercie de me contacter (en anglais) à l'adresse <`p CHEZ dirac POINT org`> et de me le faire savoir, afin que je puisse la retirer de ce petit guide.

6.1. Comment faire pour que Jigdo utilise mon mandataire (*proxy*) ?

Éditez le fichier `~/.jigdo-lite` (ou `jigdo-lite-settings.txt` pour la version Windows) dans un éditeur de texte et trouvez la ligne commençant par « `wgetOpts` ». Vous pouvez y ajouter les options suivantes :

```
-e ftp_proxy=http://mandataire-local:port/
-e http_proxy=http://mandataire-local:port/
--proxy-user=identifiant
--proxy-passwd=mot_de_passe
```

Substituez-y bien évidemment les valeurs correctes pour votre serveur mandataire. Les deux dernières options ne sont nécessaires que si votre mandataire utilise l'authentification par mots de passes. Les options doivent être ajoutées à la fin de la ligne wgetOpts et avant le caractère « `'` » final. Toutes les options doivent se trouver sur une seule ligne.

Sous Linux, une autre façon de procéder est de définir les variables d'environnement `ftp_proxy` et `http_proxy`, par exemple en les ajoutant dans le fichier `/etc/environment` ou dans le fichier `~/.bashrc`.

6.2. Jigdo-lite échoue avec une erreur — Ai-je téléchargé tous ces mégaoctets en vain ?

Évidemment, *Cela ne devrait pas se produire*™, mais pour des raison diverses et variées vous pourriez vous retrouver dans un état où un gros fichier « `.iso.tmp` » a déjà été généré et où jigdo-lite a apparemment des problèmes et vous répète continuellement de recommencer le téléchargement. Dans un tel cas, vous pouvez tentez plusieurs choses :

- Recommencer simplement le téléchargement en appuyant sur **Entrée**. Il est possible que certains fichiers n'aient pu être téléchargés à cause de problèmes réseaux temporaires ou de dépassement de délais — une autre tentative de récupération des fichiers manquants va être réalisée.
- Essayez d'utiliser un miroir différent. Certains miroirs Débian sont légèrement désynchronisés — un autre miroir contiendra peut-être les fichiers qui ont été supprimés de celui que vous avez indiqué ou bien il aura déjà des fichiers qui n'ont pas encore été répliqués sur votre miroir.
- Récupérez les parties manquantes de l'image en utilisant rsync [http://rsync.samba.org]. Tout d'abord, vous devrez trouver une URL rsync correcte pour l'image que vous téléchargez : choisissez un serveur qui offre un accès rsync aux images Iso de la distribution stable [http://www.debian.org/CD/mirroring/rsync-mirrors] ou *testing* [http://www.debian.org/CD/http-ftp/#testing], puis déterminez le nom de fichier et le chemin correct. Vous pourrez lire le contenu des répertoires en utilisant la commande :

  ```
  rsync rsync://cdimage.debian.org/debian-cd/
  ```

  Ensuite, renommez le fichier temporaire de jigdo-lite afin de supprimer son extension « `.tmp` ». Indiquez à **rsync** à la fois l'URL distante et le nom du fichier local :

  ```
  rsync rsync://serveur.org/chemin/binary-i386-1.iso binary-i386-1.iso
  ```

  Il pourrait être intéressant d'utiliser les arguments optionnels `--verbose` et `--progress` pour voir être informé de la progression et `--block-size=8192` pour améliorer la vitesse.
- Si tout le reste échoue, vos données téléchargées ne seront toujours pas perdues. Sous Linux vous pouvez monter en boucle (`loop`) le fichier `.tmp` pour accéder aux paquets qui ont déjà été téléchargés. Il est donc possible de les réutiliser pour générer une image à partir d'un nouveau fichier jigdo (tel que le dernier instantané5 journalier de la distribution *testing* si votre téléchargement raté était aussi un instantané *testing*). Pour faire cela, lancez les commandes suivantes sous le compte `root` dans le répertoire du téléchargement ayant échoué :

  ```
  mkdir mnt
  mount -t iso9660 -o loop *.tmp mnt
  ```

  Ensuite, dans un répertoire différent, démarrez un nouveau téléchargement. À l'invite « `Files to scan` », indiquez le chemin du répertoire `mnt`.

6.3. [11 août 2002] : Pourquoi les traductions de ce guide pratique ne sont-elles diffusées par le projet de

5 *Snapshot*.

documentation Linux (LDP) ?

J'ai eu beaucoup de difficultés à faire publier les traductions de ce guide pratique par certains éditeurs des projets de documentation Linux (LDP) non-anglophones.

Marco Budde `<Budde CHEZ tu TIRET harburg POINT de>`, l'éditeur du projet de documentation Linux allemand, a refusé d'accepter la traduction allemande de ce document, car elle était écrite au format DocBook et non au format LinuxDoc. Ceci en dépit du fait que le format DocBook soit le format SGML préféré du Projet de documentation Linux anglophone. C'est une honte qu'il y ait au sein de la communauté des logiciels ouverts (*open source*) des gens qui sabotent notre communauté de l'intérieur.

L'éditeur LDP portugais, Alfredo Carvalho `<ajpc CHEZ poli POINT org>` a complètement ignoré la traduction en portugais que je lui ai fait parvenir.

Si vous tenez à avoir des documents du Projet de documentation Linux dans ces langues, je vous invite à écrire à ces éditeurs et à leur demander de bien vouloir agir d'une manière plus responsable lorsqu'ils reçoivent des documents traduits. En attendant, vous pouvez obtenir ces traductions sur mon site personnel, http://www.dirac.org/linux/debian/jigdo.

6.4. jigdo met beaucoup de temps à télécharger les fichiers car wget se déconnecte et se reconnecte au serveur ftp entre chaque fichier. Existe-t-il un moyen de rendre cela plus rapide ?

Le téléchargement peut être accéléré en utilisant un serveur HTTP au lieu d'un serveur FTP — FTP n'est pas très efficace pour le téléchargement d'un grand nombre de petits fichiers. Je vous recommande également de mettre à jour la version de wget que vous utilisez, car les dernières versions permettent les connexions HTTP persistantes, ce qui permet d'obtenir un petite accélération supplémentaire.

Malheureusement, même avec des connexions HTTP persistantes, la vitesse de téléchargement ne sera jamais aussi bonne qu'avec le téléchargement d'un unique fichier Iso. De telles vitesses ne peuvent être atteinte que par des transferts HTTP en série (*pipelining*) — La version en mode graphique de Jigdo permettra les transferts en série.

6.5. Que se passe-t-il si mon téléchargement Jigdo s'interrompt ?

Si le téléchargement est interrompu, il vous suffit de relancer jigdo-lite et d'appuyer sur **ENTRÉE** à chaque question posée. Jigdo-lite reprendra là où il en était.

6.6. Il n'est pas possible de finir mon téléchargement Jigdo, car le fichier `.jigdo` est défectueux. Après avoir téléchargé un nouveau fichier `.jigdo` valide, devrais-je télécharger à nouveau toutes les données ?

Il est possible que le fichier `.jigdo` que vous avez téléchargé soit défectueux. C'est très rare, mais cela arrive de temps en temps avec des cibles mouvantes telles que les distributions Débian *testing* et instable.

Si vous vous apercevez que le fichier `.jigdo` est défectueux, vous devrez télécharger un nouveau fichier `.jigdo` (lorsqu'un fichier corrigé sera disponible), mais vous n'aurez *jamais* besoin de télécharger à nouveau toutes les données de l'image Iso.

Vous pouvez utiliser la même astuce de montage en boucle que nous avons utilisé pour mettre à jour notre image Iso. La différence est que vous ne disposez pas d'une image `.iso` complète pour démarrer. Cependant, le fichier `.iso.tmp` est également une image Iso et peut être utilisé pour recommencer le téléchargement sans avoir à re-télécharger toutes les données précédemment téléchargées.

Montez simplement en boucle (`loop`) le fichier `.iso.tmp` sur `/mnt`. Lorsque vous relancerez jigdo-lite avec le fichier `.jigdo` corrigé, dites simplement à Jigdo de parcourir le répertoire `/mnt`. N'oubliez pas de renommer ou de déplacer le fichier `.iso.tmp` afin qu'il n'interfère pas avec la création d'un nouveau `.iso.tmp` par jigdo-lite.

6.7. Puis-je utiliser Jigdo pour télécharger des images de DVD ?

Absolument ! Le procédé est identique au téléchargement d'une image de CD. La seule différence est qu'il vous faudra télécharger les fichiers `.jigdo` et `.template` correspondant aux DVD Débian au lieu de ceux correspondant aux CD. Vous pourrez trouver les fichiers `.jigdo` et `.template` permettant de télécharger des images de DVD Débian sur http://www.debian.org/CD/jigdo-cd/.

Notez que vous devrez disposer d'un noyau Linux en version 2.4 ou supérieure pour créer des fichiers images de la taille d'un DVD. Sous Windows, il est impossible pour le moment de créer des images de la taille d'un DVD du fait de la non-compatibilité avec les fichiers de grande taille de la bibliothèque C++ utilisé la version mingw de gcc.

6.8. Puis-je graver le fichier `.iso.tmp` sur un CD ?

Grâce à Gordon Huff et David Anselmi, nous savons maintenant que la réponse est « oui vous pouvez ». Mais, plus important, Gordon nous a donné une bonne raison de vouloir le faire. Paraphrasons Gordon :

> Le Windows 98 de mon ami a une connexion *sympa* au câble. J'arrive le matin, je démarre Jigdo (plus d'un, en fait) et nous partons faire les boutiques, re-fixons le plant de kiwi, mettons en place le sapin de Noël et ses décorations, commandons et partageons pizza et mettons en route la télé satellite.
>
> Je quitte le domicile de mon ami avec plusieurs `.iso.tmp` gravés sur des CD-RW. Arrivé à la maison, j'utilise les images Iso non terminées pour mettre à jour ma configuration Jigdo, qui utilise un accès téléphonique.

6.9. Pourquoi est-ce que Jigdo ne fait rien ? Il télécharge des paquets et les détruits. Je sais qu'il ne les ajoute pas au fichier `iso.tmp` car sa taille ne change pas !

Jigdo marche très bien — Le fichier `.tmp.iso` a dès le début sa taille finale, mais est rempli de zéros. Plus tard, des morceaux de cette image seront remplacées par les données téléchargées.

Vous pouvez vous rendre compte que Jigdo progresse en regardant les messages « `Found X of the Y files required by the template`[6] » qui sont affichés de temps à autre. La seconde valeur « `Y` » doit décroître. Quand elle atteindra zéro, le téléchargement sera terminé.

6.10. J'ai des difficultés à faire fonctionner jigdo-easy

Voyez la Section 7.1, « jigdo-easy ».

[6] J'ai trouvé X fichiers sur les Y fichiers nécessaires au patron.

6.11. [10 février 2003] : J'ai des difficultés à télécharger la distribution Sarge ou Sid avec Jigdo

Si vous utilisez une distribution Débian version Potato ou Woody : suite à un changement dans Jigdo, les versions de jigdo-lite des distributions Potato et Woody (r0 et r1) ne peuvent pas télécharger les images des distributions Sid et Sarge. Les paquets *jigdo-file* des distributions Sarge (version 0.7.0-2) et Sid (version 0.7.0-2) peuvent télécharger des images Sarge et Sid. De même une version corrigée de *jigdo-file* (0.6.5) a été proposée afin d'être incluse dans la 2e révision de la distribution Woody. Elle sera, espérons-le, incluse dans la Woody r2 lorsque celle-ci sortira. Cependant, du fait des corrections et des améliorations qu'elles contiennent, je vous incite à utiliser le *jigdo-file* de Sarge ou de Sid.

Si vous utilisez Sarge ou Sid, alors vous risquez d'avoir besoin d'aide. Parcourez les archives de la liste de discussion debian-cd, et si cela ne résout pas votre problème envoyez-leur une demande d'aide (cf. la Section 7.4, « Ressources »).

6.12. Jigdo-lite est trop bavard. Comment puis-je supprimer tous ces messages ?

Jigdo-lite utilise wget, et les messages affichés par wget peut être très verbeux. Si cela vous dérange, vous pouvez rendre wget plus silencieux en ajoutant `--non-verbose` au commutateur `wgetOpts` dans votre fichier `~/.jigdo-lite`. Si vous voulez que wget n'affiche aucun message, utilisez `--quiet` dans le commutateur `wgetOpts`.

6.13. Puis-je utiliser Jigdo sur d'autres plates-formes que Linux ?

Certainement. Si vous êtes intéressés par la distribution Woody ou Potato et que vous êtes sous Windows, le vieux SunOS, HP-UX ou IRIX, vous pouvez utiliser jigdo-easy. Reportez-vous à la Section 7.1, « jigdo-easy » et à la Section 7.4, « Ressources ».

Si vous désirez télécharger la distribution Potato, Woody, Sarge ou Sid depuis Microsoft Windows, jigdo-lite a été porté sur cette plate-forme et peut être obtenu depuis le site principal de Jigdo (cf. la Section 7.4, « Ressources »).

6.14. Sous MS Windows, pourquoi ai-je souvent un message d'erreur « `No such file or directory` » (impossible de trouver le fichier ou le répertoire) ?

Sous Windows, il se peut que vous vous aperceviez que jigdo-lite télécharge quelques fichiers mais échoue lors de la lecture de leur contenu. Cela produira un message d'erreur « `No such file or directory` » (impossible de trouver le fichier ou le répertoire).

Il semble que cela se produise si les noms des fichiers lus par Jigdo dépassent une certaine taille. La solution est de déplacer ce qui a déjà été téléchargé dans un dossier plus proche de la racine du disque. Ceci afin de limiter la taille du nom complet du fichier.

6.15. Sous MS Windows, pourquoi mon image ne peut-elle dépasser 2 Go ?

Lorsque vous créez des images DVD sous Windows en utilisant jigdo-win, vous devez être conscient qu'il existe une limite de 2 Go pour la taille des fichiers. Ceci est causé par la bibliothèque C++ du compilateur mingw32 qui est utilisé pour compiler jigdo-win. Apparemment, cette bibliothèque n'est pas compatible avec les fichiers de grande taille du fait d'une omission de ses développeurs. Ce qui implique que cette limite de 2 Go sera présente même sous le système de fichier NTFS.

Il est probable que cette limitation sera corrigée dans une version à venir de mingw32.

7. Errata

7.1. jigdo-easy

Jigdo-easy, de Anne Bezemer, est un logiciel dérivé de jigdo-lite qui est portable sur un plus grand nombre de plates-formes, dont Windows, le vieux SunOS, HP-UX et IRIX. Jigdo-easy est aussi plus simple à utiliser que jigdo-lite. Cependant, du fait des changements récents apportés à Jigdo, Jigdo-easy ne marche que pour télécharger les distributions Potato et Woody. Jigdo-easy ne permet pas de télécharger une image Iso de Sarge ou de Sid. Voir la Section 7.4, « Ressources » et la Section 6.13, « Puis-je utiliser Jigdo sur d'autres plates-formes que Linux ? ».

7.2. Interface graphique

Une interface graphique GTK+ pour Jigdo est en cours de développement. Elle n'est pas encore totalement fonctionnelle, mais finira par être disponible. Il y aura une interface graphique pour Linux et pour Windows.

7.3. `jigdo-file-cache.db`

Ce fichier de base de données contient les sommes de contrôle md5 des fichiers lus lorsque vous indiquez un répertoire en réponse à la question `Files to scan:`. Si vous faites parcourir à jigdo-file une seconde fois le même répertoire, son parcours sera très rapide.

Ce fichier peut se révéler très utile dans le cas suivant : la révision 0 de votre distribution est mise à jour et devient une rév. 1. Dans les images CD de la rév. 1, des paquets peuvent avoir été déplacés du CD `n` au CD `n+1` ou vice-versa. Si vous avez une liaison particulièrement lente (un modem), vous souhaiterez sans doute éviter de télécharger à nouveau ces paquets. Pour cette raison, lorsque vous téléchargerez une nouvelle version du CD `n`, vous ferez parcourir à jigdo-lite les trois CD `n-1`, `n` et `n+1` (voire même les 8 CD si vous voulez être sûr à 100%).

Si jidgo-lite doit parcourir le même CD encore et encore alors qu'il met à jour les 8 images, cette base de données empêchera toutes les données présentes sur le CD d'être lues plusieurs fois.

Cette base de données est bien plus importante quand on *génère* des fichiers `.jigdo`, car vous ne voudrez pas relire l'intégralité de votre miroir Débian de 50 Go chaque fois que vous générez un nouveau fichier `.jigdo`.

7.4. Ressources

Ce guide pratique tire à sa fin, mais j'ai pensé que je ne vous laisserai partir qu'avec quelques liens et références pour en savoir plus sur les outils Jigdo et la façon dont ils fonctionnent.

http://atterer.net/jigdo (en anglais)	Voici le site de Jigdo. Vous devriez sans l'ombre d'un doute visiter ce site ; il contient beaucoup d'informations sur les versions de Jigdo pour les différentes plates-formes, les interfaces graphiques et d'une manière générale tout ce qui se rapporte à Jigdo.
http://cdimage.debian.org/~costar/jigdo (en anglais)	La Page Débian pour jigdo-easy (cf. la Section 7.1, « jigdo-easy »).
http://www.debian.org/CD/jigdo-cd	La page principale de Jigdo.

file.html	Si vous utilisez la distribution Potato ou Woody, pensez à mettre à jour *jigdo-file* vers la version `0.7.0-2` provenant de *testing* ou d'instable (cf. la Section 6.11, « [10 février 2003] : J'ai des difficultés à télécharger la distribution Sarge ou Sid avec Jigdo »).
http://lists.debian.org/search.html (en anglais)	Vous pouvez utiliser cette page pour faire des recherches dans les archives de la liste de discussion debian-cd.
http://www.debian.org/MailingLists/subscribe	La page permettant de s'abonner à la liste de discussion debian-cd.