

# Debian Jigdo mini-HOWTO

Peter Jay Salzman

p@dirac.org

Copyright © 2001 Peter Jay Salzman  
2005-12-05 ver 1.8

## Estratto

Ottenere ISO di Debian (immagini ISO della distribuzione Debian) è stato sempre un processo doloroso, lento ed estremamente inefficiente. Jigdo è uno strumento per ottenere ISO Debian in un modo facile, veloce e molto efficiente. Questo HOWTO descrive perché bisognerebbe usare il jigdo, una piccola guida per capire come funziona e come usarlo per ottenere ISO Debian.

Jigdo è uno strumento molto generale e non è legato specificamente a ISO Debian. Gli strumenti di jigdo possono essere utilizzati per rendere ogni immagine ISO disponibile per essere scaricata in modo facile, veloce ed efficiente come avviene per le ISO Debian. Questo HOWTO comprenderà pure questo, ma metteremo a fuoco soprattutto lo scaricamento delle ISO Debian ISO.

## 1. Administrata

### 1.1. Condizioni autore e Copyright

Questo documento è un copyright (c) 2001 Peter Jay Salzman, <p@dirac.org (mailto:p@dirac.org)>. è consentita la copia, distribuzione e/o modifica di questo documento sotto i termini della Licenza Open Software (OSL), versione 1.1. Odio gli HOWTO che includono la licenza; è un assassinio di alberi. Si può leggere l'OSL al sito <http://opensource.org/licenses/osl-1.1.txt>.

Se si desidera creare un lavoro derivato o pubblicare questo HOWTO per scopi commerciali, gradirei essere contattato prima. Ciò mi darà un'opportunità di dare la versione più recente. Inoltre gradirei una copia di qualsiasi cosa si stia facendo facendo oppure spinaci, aglio, funghi, formaggio feta e una pizza al cuore di carciofo.

## 1.2. Ringraziamenti

Vorrei ringraziare l'autore di jigdo, Richard Atterer (mailto:atterer@debian.org), semplicemente per aver scritto jigdo. Chiunque abbia ottenuto ISO Debian attraverso altri mezzi saprà perché. Questo HOWTO è cominciato dalle pagine web che ho scritto sulla mia esperienza con jigdo. Richard ha impegnato tempo per spedirmi via e-mail varie correzioni, chiarificazioni e risposte alle domande che avevo su jigdo. Da allora, ha letto il mio lavoro molte volte. Richard è uno sviluppatore che si preoccupa non solo del suo lavoro, ma anche della gente che lo usa. Sfortunatamente, questo sta diventando meno comune in questo mondo indaffarato in cui viviamo. Grazie, Richard!

Inoltre vorrei ringraziare Conrad Wood (mailto:cnw@conradwood.net), Elcio Mello, Marcelo Ramos (mailto:mramos@montevideo.com.uy), Yufeng Wang, Tsukasa Yamaguchi, Yuri Kozlov (mailto:kozlov.y@gmail.com), ed Oguz Yarimtepe (mailto:oguz@comu.edu.tr) per aver tradotto questo mini-HOWTO in lingue diverse dall'inglese. Mi ritengo completamente onorato che abbiano trovato le mie parole degne del loro tempo e sforzo. Grazie Ragazzi!

In fine, vorrei ringraziare Mark van Lent (mailto:mark@panic.et.tudelft.nl), Gordon Huff, David Anselmi, Thierry Cabuzel (mailto:thierry.cabuzel@skynet.be), Russell L. Harris (mailto:rlharris@hal-pc.org), e Jens Seidel (mailto:tux-master@web.de) per le gentili parole e le correzioni.

## 1.3. Commenti e Correzioni

Io mi preoccupo molto delle persone che si servono di questo documento. Anche i mini-HOWTO occupano molto tempo per scriverli, e io non avrei investito così tanto lavoro in qualcosa che la gente non capisce. Se si hanno commenti, correzioni o suggerimenti, anche in materia di stili di scrittura, non esitare a mandarmi una email. Finché non sono completamente sommerso dalla mia dissertazione di PhD e dal libro che sto scrivendo sul codice di debugging con GDB/DDD per No Starch Press, farò del mio meglio per rispondere ad ogni email che ricevo riguardo questo mini-HOWTO. Notizie flash: ho completato il mio Ph.D.; ora sono inondato dalla ricerca di un lavoro. Qualcuno ha bisogno di noleggiare un fisico teoretico?

## 1.4. Ultima versione e Traduzioni

Italiano:

Classe 5B (a.s. 2005/06) ITIS "E.Medi" - Galatone(LE)

Mercuri Fabio <fabio.mercuri@istruzione.it>. Disponibile a <http://www.salug.it/~cicoriella/>.

Zuccalà Carla

Tedesco:

Conrad Wood <cnw@conradwood.net>.

Portoghese

Elcio Mello.

Spagnolo

Marcelo Ramos <mramos@montevideo.com.uy>.

Cinese

Yufeng Wang

Giapponese

Tsukasa Yamaguchi. Disponibile a <http://www.linux.or.jp/JF/JFdocs/Debian-Jigdo>.

Russo

Yuri Kozlov <kozlov.y@gmail.com>. Disponibile a  
[http://alioth.debian.org/project/showfiles.php?group\\_id=30279](http://alioth.debian.org/project/showfiles.php?group_id=30279).

Turco

Oguz Yarimtepe <oguzy@comu.edu.tr>. Disponibile a  
<http://docs.comu.edu.tr/howto/debian-jigdo.html> (<http://docs.comu.edu.tr/howto/debian-jidgo.html>).

In aggiunta agli URL dati sopra, tutte le traduzioni (così come la versione inglese) sono disponibili al mio sito web: <http://www.dirac.org/linux/debian/jigdo>. Se si volesse tradurre questo mini-HOWTO in un'altra lingua, per favore contattarmi su <p@dirac.org (mailto:p@dirac.org)>.

La versione inglese di questo HOWTO può anche essere trovata al Progetto Della Documentazione Di Linux: <http://tldp.org/docs.html>.

## 2. Perché jigdo?

### 2.1. Come si ottiene un insieme di immagini ISO Debian?

Se si vuole un insieme di CD Debian ci sono molti modi per ottenerli. Un modo è comprarli da venditori (<http://www.debian.org/CD/vendors/>) che forniscono CD Debian. Ciò in definitiva è la cosa più meritevole, poiché alcuni fornitori donano i soldi di nuovo al progetto Debian. Le donazioni contribuiscono ad assicurare la sopravvivenza di Debian per lungo tempo.

Un altro modo per ottenere un set di CD Debian è quello di farsene una propria copia. Ciò richiede di ottenere un'immagine ISO e in seguito di scrivere quell'immagine ISO su un CD vuoto. Prima di jigdo, c'erano due modi di creare CD Debian:

1. Scaricarsi l'intera ISO
2. Usare il kit di pseudo-immagine (PIK)

Questo documento riguarda il modo più nuovo e migliore per ottenere ISO Debian, per mezzo di uno strumento chiamato jigdo. Infatti, il PIK ora è ufficialmente morto e tutti gli ulteriori riferimenti ad esso sono stati rimossi da questo documento. Il metodo canonico per ottenere ISO Debian è quello di usare jigdo.

## 2.2. Perché non scaricare l'intera immagine ISO?

Ci sono dei server mirror che offrono di scaricare ISO Debian via http e ftp. Il problema è che ci sono pochi siti mirror, e la loro portata non può soddisfare la richiesta di coloro che desiderano ISO Debian. Per esempio, fsn.hu, secondo come riferito, ha saturato la connessione del relativo fornitore. Il traffico in uscita arriva a pochi terabytes al mese!

In aggiunta, "Debian testing" è instabile e si aggiorna spesso. Le proprie ISO possono non essere le più aggiornate lo stesso giorno che sono state scaricate a meno che non si trovino alcuni metodi particolari per aggiornarle, come installare le ISO su un dispositivo loopback e usare rsync (quello che ha fatto il PIK). Così se si desiderano immagini ISO aggiornate, si deve scaricare un nuovo insieme di immagini ISO ogni giorno. Chiaramente, questo non è il modo con il quale si desidera ottenere ISO Debian!

Anche se si desidera scaricare le immagini "Debian stable", esse vengono ancora aggiornate ogni pochi mesi. Lo scaricamento di immagini ISO darà le immagini aggiornate per alcuni mesi, ma ogni volta che una nuova revisione stabile di Debian viene rilasciata bisognerà rifare da zero quel processo antipatico di scaricamento dell'intero set di ISO. Ciò non è un buon uso del proprio tempo e delle risorse del server mirror.

## 2.3. Cos'è Jigdo?

Jigdo (che sta per "Jigsaw Download") è stato scritto da Richard Atterer (mailto:atterer@debian.org), ed è rilasciato con licenza GNU GPL. È uno strumento che permette scaricamento e aggiornamento efficienti di un'immagine ISO. Qualsiasi immagine ISO. Jigdo non è specificamente di Debian, comunque Debian lo ha scelto come metodo ufficiale di scaricamento delle immagini ISO.

Un'idea sbagliata comune è che il jigdo crei le immagini ISO; non è così. Discutiamo il processo generale di come jigdo permette di ottenere un'immagine ISO. Supponiamo che Adam (è il manager del rilascio di Debian) sia la persona che offre l'immagine ISO. Supponiamo che Betty (un utente Debian) sia la persona che desidera scaricare l'immagine ISO.

1. Adam per prima cosa crea un'immagine ISO adattata per scriverla su un CD. Potrebbe usare un programma di utilità come mkisofs o debian-cd per creare l'immagine ISO. Inoltre lui crea due piccoli file associati con una sua immagine ISO recentemente creata: un file `.jigdo` e un file `.template`. Rende questi due file scaricabili da chiunque desideri ottenere la sua immagine ISO.
2. Betty poi scarica i file `.jigdo` e `.template`. Lei usa il jigdo-lite con questi due file per scaricare l'immagine ISO di Adam.
3. Quando Debian si aggiorna, Adam crea una nuova versione delle ISO e genera i nuovi file `.jigdo` e `.template`.
4. Quando Betty vuole aggiornare i suoi CD, scarica i nuovi file `.jigdo` e `.template` e li usa con jigdo-lite per aggiornare la sua copia delle immagini ISO. La cosa importante qui è che scarica solo le differenze tra la sua vecchia ISO e la nuova ISO di Adam. Non deve di nuovo scaricare le parti che sono immutate.

Jigdo ha due programmi di utilità: `jigdo-file` (usato da Adam) che crea i file `.jigdo` e `.template` file, e `jigdo-lite` (usato da Betty) che usa questi due file per scaricare o aggiornare le ISO. Se tutto ciò che si desidera fare è ottenere Debian ISOs, usare solo `jigdo-lite`. Ci si può persino dimenticare che `jigdo-file` esista :-)

Jigdo va bene per tutti i problemi degli altri metodi per ottenere le ISO Debian:

- È molto più veloce del trasferimento dell'intera immagine ISO;
- Invece di scaricare l'intera immagine ISO, può usare un vecchio CD (o una vecchia immagine ISO montata con il parametro "loop"), scarica *soltanto* i file che sono cambiati da quando il CD (immagine ISO) è stato creato e genera una nuova ISO aggiornata. È molto simile ai cvs per aggiornare il codice sorgente.
- Il jigdo-lite usa il wget che, come predefinito, usa l'HTTP per trasferire i file. Diversamente da rsync l'HTTP non è mai stato bloccato dai firewall (tranne quelli dietro i quali non bisognerebbe usare jigdo per iniziare).
- Jigdo è molto "gentile" con la larghezza di banda dei server che offrono le immagini di Debian. I mirror di Debian possono gestire un carico di gente che usa jigdo, per scaricare le immagini Debian, più grande di quello che utilizzano altri metodi.

Chiaramente, jigdo è il metodo migliore per ottenere ISO Debian.

### 3. Come funziona jigdo (facoltativo)

Non c'è bisogno di conoscere questo materiale per scaricare ISO Debian. Ma ci può aiutare a scoprire come jigdo funziona. Se non si è interessati nei dettagli andare semplicemente avanti alla la Sezione 4.

### 3.1. Prepare l'ISO per il trasferimento

Un'immagine CD è un file system denominato iso9660, ma per questa discussione possiamo parlare in modo sicuro di un'immagine cd come un grande file denominato "immagine ISO" (circa 650MB) che contiene i file in varie posizioni. Per esempio, se un cd contiene un file di 567 byte chiamato README, l'immagine ISO potrebbe avere i contenuti del file README fra le posizioni 20480000 e 20480567. Si può visualizzare un'immagine CD come:

```
-----
ISO Image:  |xxxx| file-0 |xx| file-1 |xxx| file-2 |x| file-3 |xxxx|
-----
```

Le aree di "x" dell'immagine contengono cose come le informazioni di una directory, il riempimento zero, il nome del disco, il settore di boot, ecc.

jigdo-file prende due cose come input: l'immagine CD completa (è necessario che l'ISO sia già fatta) e un insieme di file che possono o non possono essere nell'immagine. Ecco una visualizzazione dell'input del jigdo-file:

```
-----
ISO Image:  |xxxx| file-0 |xx| file-1 |xxx| file-2 |x| file-3 |xxxx|
-----

Loose Files:  -----
              | file-0 | | file-1 | | file-3 | | file-4 |
              -----
```

Con magia, jigdo-file scopre quali dei file sciolti sono contenuti nell'immagine ISO e le loro posizioni all'interno del file ISO. Dà in output due file: un file .template e un file .jigdo

### 3.2. Il file .template

Dati in input un'immagine ISO e una serie di file che possono o non possono essere nell'immagine ISO, il jigdo-file dà in output un file .template per quell'immagine ISO. Ecco a cosa assomiglia il file .template

```
-----
.template:  |xxxx| md5-0 |xx| md5-1 |xxx|cccccccc|x| md5-3 |xxxx|
-----
```

Il jigdo-file scoprirà che i file-0, file-1 e file-3 erano contenuti nell'immagine ISO. Rimossi i contenuti di questi file li sostituisce con il checksum md5 del relativo file (md5-0, md5-1, ecc).

I dati "x" (informazioni directory, riempimento zero, etc.) all'interno dell'immagine ISO sono compressi e riscritti nel file `.template`. Infine, tutti i file all'interno dell'immagine ISO che non sono stati forniti come file sciolti (tipo `file-2`) sono compressi e scritti sul file `.template`. Ciò è indicato come dati "c" nella visualizzazione del file `.template`.

I file sciolti forniti a `jigdo-file` che non sono stati trovati nell'immagine ISO (come `file-4`) sono ignorati.

### 3.3. Il file `.jigdo`

Dando in input un'immagine ISO e una serie di file sciolti i quali possono o no essere nella immagine ISO, `jigdo-file` genera il file `.jigdo` relativo. I file `.jigdo` di Debian sono compressi con `gzip`, quindi c'è bisogno di usare `zcat` o `zless` per vederlo. Ecco come appare un file `.jigdo` quando viene decompresso:

```
md5-0=http://somemirror.org/file-0
md5-1=http://somemirror.org/file-1
md5-2=http://somemirror.org/file-2
md5-3=http://somemirror.org/file-3
```

Il file `.jigdo` fornisce semplicemente una corrispondenza tra `md5sum` di un file all'interno dell'immagine ISO e la URL di download di questo file. Ci sono altre cose all'interno del file `.jigdo`, e se si guarda all'interno si vede che il `jigdo-file` ha lo stesso formato di un file `.ini`. Dovrebbe essere autoesplicativo, ma se si vogliono maggiori dettagli è necessario consultare la documentazione di `jigdo`.

Il formato sopra descritto non è uguale a quello che si potrebbe vedere in un tipico file `.jigdo`, ma è molto simile. Osservando la sezione [Servers] in fondo al file `.jigdo`, si vedrà esattamente la differenza fra ciò che si è scritto sopra e un reale file `.jigdo`.

### 3.4. Download dell'immagine

Una volta usato `jigdo-file` per generare il file `.jigdo` e `.template` per un'immagine ISO, ognuno può usare `jigdo-lite` per scaricare quell'immagine. `jigdo-lite` scarica tutti i files dell'ISO di Debian usando `wget`, li assembla e crea una copia al volo dell'immagine ISO.

## 4. Scaricare la propria Prima Immagine (In 5 semplici passi)

Supponiamo di partire da zero e non di non avere già disponibile alcuna ISO Debian. Una volta masterizzato il proprio set di immagini ISO è possibile usare `jigdo-file` per aggiornarlo. L'aggiornamento

delle immagini ISO sarà trattato nella prossima sezione.

## 4.1. Installare Jigdo

Prima si installa il pacchetto `jigdo-file`:

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

Jigdo è in pieno sviluppo. Correzioni e miglioramenti sono costanti, se si sta usando la versione stabile o di prova, si può scaricare la versione instabile di `jigdo-file` all'indirizzo <http://packages.debian.org/unstable/utils/jigdo-file.html>. Al 28 Novembre 2005 essa è alla versione 0.7.2-2.

## 4.2. Download dei file `.template` e `.jigdo`

Per ogni immagine ISO che si vuol scaricare, c'è bisogno dei relativi file `.template` e `.jigdo`. Il nome di entrambi i file è dato con la stessa convenzione:

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

dove `distro` è il nome della distribuzione (come "sarge"), `arch` è l'architettura (come "i386") e `n` è il numero del disco (come "1").

Per esempio, `sarge` ha otto immagini, quindi è necessario scaricare 8 file `.jigdo` e 8 file `.template`. Questi possono essere scaricati da <http://www.debian.org/CD/jigdo-cd/>. Il primo file `.jigdo` e `.template` hanno nome rispettivamente `sarge-i386-1.jigdo` e `sarge-i386-1.template`

## 4.3. Avviare `jigdo-lite`

Lanciare `jigdo-lite` e fornire il file `.jigdo` dell'immagine che si vuol scaricare. Usando `Sarge` come nell'esempio:

```
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-i386-1.iso)
```

```
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:
```

Se si interrompe jigdo-lite con **control+z** (non si deve fare; è per mostrare cosa succede) e si osserva l'output del comando **ls**, si trova un nuovo file nella directory chiamato `sarge-i386-1.jigdo.unpacked`. Ciò svela che il file `.jigdo` è compresso con `gzip`. Questo file è semplicemente una versione decompressa del file `.jigdo`.

Quando jigdo-lite chiede se si ha una versione precedente del primo CD di sarge, si dovrà dare il percorso al CD. Questo è il modo per aggiornare le proprie immagini ISO (o completare i download incompleti). Poiché stiamo supponendo di partire da zero e di non avere ancora ISO Debian ISO, non abbiamo nulla da indagare. Questo è trattato nella la Sezione 5, quindi basta premere **ENTER**.

Si veda anche la Sezione 7.2.

## 4.4. Specificare un server mirror

Si vedrà:

```
-----
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/]:
```

Normalmente, `jigdo-lite` estrae il server dal proprio `/etc/apt/sources.list`. Volendo usare un server diverso, si deve specificare qui. Se questo è il server che si vuole usare, premere **ENTER**. `Jigdo-lite` scriverà quindi un file `.jigdo-lite` nella cartella home dell'utente.

Dopo, se il file `.jigdo` che si sta usando si riferisce a un pacchetto che bisogna scaricare da un server mirror Non-US, `jigdo-lite` vi chiederà un server mirror Debian Non-US. Il messaggio che appare (e la propria risposta) sarà molto simile alla finestra di dialogo del server mirror del paragrafo precedente.

```
-----
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` scriverà la vostra scelta in `~/.jigdo-lite`. Comunque, se l'immagine che si sta per scaricare non contiene software Non-US non si vedrà questo messaggio.

Se in futuro e in qualunque momento si vuole cambiare il server predefinito usato con `jigdo`, si dovranno modificare queste due righe in `~/.jigdo-lite`:

```
debianMirror='http://some-mirror-to-use/debian/'
nonusMirror='http://some-other-mirror/debian-non-US/'
```

## 4.5. Download delle ISO

Dopo aver specificato i server mirror, `jigdo-lite` inizierà il download dei file per assemblare l'immagine 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://mirror/debian/pool/main/p/pack/pack_3.10-1_i386.deb
=> 'sarge-i386-1.iso.tmpdir/mirror/debian/pool/main/p/pack/pack_3.10-1_i386.deb
Resolving linux.csua.berkeley.edu... done.
Connecting to linux.csua.berkeley.edu[128.32.112.231]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1,911,624 [application/x-debian-package]

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

Ci saranno molti messaggi che scorrono sullo schermo; se confondono, consultare la la Sezione 6.13. Mentre jigdo-lite sta scaricando i pacchetti, aprire un'altra console (oppure aprite un altro xterm) ed eseguire **ls** nella cartella in cui è eseguito jigdo-lite. Ora ci dovrebbero essere sei file in questa cartella:

- 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

La cartella `sarge-i386-1.iso.tmpdir/` contiene tutti i pacchetti Debian che jigdo-lite scarica. Spesso la cartella si svuota e i file vengono scritti in `sarge-i386-1.iso.tmp`, che è una versione temporaneamente incompleta dell'immagine ISO che si vuole ottenere. Notare che `sarge-i386-1.iso.tmp` non apparirà fino a quando `sarge-i386-1.iso.tmpdir/` non si svuota la prima volta.

`jigdo-file-cache.db` è un file Berkeley DB che contiene gli md5sum di ogni file letto in ingresso quando si specifica una cartella alla richiesta `Files to scan:`. Ciò è descritto in la Sezione 7.3.

A questo punto, si può giocare a Quake III perché questo porterà via un po' di tempo (si potrebbe giocare su un'altra macchina perché jigdo consuma molte risorse del disco quando riversa i file nel file `.iso.tmp`). A un certo punto il download terminerà e si vedrà:

```
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!
```

## 5. Aggiornamento dell'immagine

Presumibilmente, si è letta l'ultima sezione, seguito le istruzioni, masterizzata la nuova ISO su CD ci si sente confusi. Presto o tardi, alcuni pacchetti saranno aggiornati ed ora si vorrà donare il vecchio CD ad un qualunque "novellino" alla festa dell'installazione del proprio LUG e masterizzare un insieme di CD aggiornati. Siccome si sta per diventare un jigdo-guru, non entreremo in particolari penosi come nell'ultima sezione

Il primo passo è scaricare di nuovo i file .jigdo e .template, per le immagini che si desidera aggiornare. Ci si potrebbe chiedere perché è necessario scaricarli nuovamente. Il motivo è che l'immagine aggiornata da scaricare è cambiata. Dei file possono essere stati aggiunti o rimossi, ma, anche se non lo fossero, tutti i pacchetti o file aggiornati avranno un checksum diverso dal checksum elencato nei file .jigdo e .template usati quando si è scaricata la prima volta l'immagine

A questo punto, o si è in possesso di un vecchio CD o di una vecchia ISO sull'hard disk. Andiamo per gradi su come ottenere un file ISO aggiornato. Se è un CD, inserirlo nel lettore cd e montarlo:

```
$ mount /cdrom
```

In altro modo, avendo un file ISO da aggiornare, montarlo in un loop device (da utente root per poterlo fare). Usando Woody, per esempio:

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

Ora avviamo jigdo-lite con il file .jigdo per argomento.

```
$ 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 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:
```

jigdo-lite chiedendo di indicargli il percorso del proprio CD montato (se si sta aggiornando un CD) o il proprio file ISO montato in loop (se si sta usando un file ISO). Nell'esempio si sta usando un file ISO montato in `/mnt`, perciò si introdurrà `/mnt`. Se si sta aggiornando un CD inserire la directory dove è montato, molto probabilmente `/cdrom`. In entrambi i casi jigdo-lite scansionerà le directory del proprio dispositivo montato, determinerà quali file devono essere aggiornati e riuserà quei file che non hanno bisogno di essere aggiornati. Vedere anche la Sezione 7.2.

Si potrebbe vedere qualcosa come:

```
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.
```

Che cosa è successo? Si vuole mostrare questo perché è un problema che prima o poi si presenterà. Si sta aggiornando un file ISO, ma il mio vecchio file di immagine è nella stessa directory in cui si sta lavorando. Jigdo-lite vuole generare un file chiamato `woody-i386-1.iso` ma c'è già un file con quel nome nella directory corrente (vecchia immagine). Jigdo-lite non vuole distruggere quel file, mi fa sapere che posso cancellare quel file o usare `--force` per sovrascrivere il file. È possibile anche rinominarlo o spostarlo, ma credo che jigdo-lite presupponga che questo sia già noto. :-)

Non si abbia paura di spostare o rinominare il file di immagine solo perché è stato montato. Il filesystem usa gli inodes sotto il "cappuccio" dei nomi e, anche se si sposta o rinomina il file, l'inode rimane lo stesso. Non si danneggerà il filesystem montato in `/mnt`. Per quanto riguarda la cancellazione del file ISO, anche ciò non danneggerà il filesystem montato. L'inode del file sarà disallocato solo quando il conteggio di riferimento dell'inode sarà zero. Il montaggio dell'immagine ISO aumenta il conteggio di riferimento, in modo che il file viene realmente cancellato solo dopo aver usato il comando `rm` sul file e smontato il dispositivo. Se si sta aggiornando il CD, di tutto ciò, non ci si deve preoccupare. :-)

Quindi per risolvere il problema, per esempio, si rinomina il file ISO in `woody-i386-1.iso.old` e si eseguirà di nuovo il jigdo-lite. Provando ancora:

```
$ 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** ricorda che si voleva esaminare /mnt e informa che è possibile digitare 1 ed esaminare quella cartella o digitare di nuovo la cartella. Volendo essere ostinati, si digita di nuovo il nome della cartella.

L'ellipsis rappresenta una parte di testo che cambia rapidamente La prima ellipsis è un elenco dinamico dei file che jigdo-lite sta esaminando. La seconda ellipsis denota il progresso nello scrivere woody-i386-1.iso.tmp. Una volta che jigdo-lite finisce, di esaminare i file e di scrivere il file ISO provvisorio, stampa:

```
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:
```

Poichè normalmente non si ha un'altra fonte di file da esaminare diversa dal proprio file ISO montato in loop (o dal CD), premere **ENTER**. Jigdo-lite chiederà quale server si vuole usare proprio come quando si è scaricata l'immagine ISO per la prima volta. Si è già risposto prima a queste domande, ma se veramente non si ricorda, è possibile rileggere la Sezione 4.4.

A questo punto, si vedrà jigdo-lite lavorare magicamente. Non è semplice?

## 6. Domande frequenti

Le domande precedute da una data indicano che essa è temporanea (una domanda che si riferisce ad una situazione provvisoria). Vedendo una di queste domande e sapendo che la situazione provvisoria è cambiata, per favore contattarmi (mailto:p@dirac.org) e farmi sapere, in tal modo posso rimuovere la domanda dal mini-HOWTO.

### 6.1. Perché jigdo chiede *due volte* di esaminare i file esistenti? Non è abbastanza dire sì una volta?

Continua a chiedere questo finchè non si digita un percorso da analizzare. L'idea è che si può voler analizzare diversi vecchi CD, così è possibile inserirne uno dopo l'altro nel lettore e poi continuare a fornire il percorso "D:\\" (o altro). Vedere anche la Sezione 7.2.

### 6.2. Jigdo ha problemi a scaricare alcuni nomi di file.

Quando si scaricano le immagini di Debian sotto Windows, sembra che jigdo-lite abbia difficoltà a scaricare uno o più file seguenti:

```
libbusiness-onlinepayment-bankofamerica-perl_XXX_all.deb
libbusiness-onlinepayment-authorizenet-perl_XXX_all.deb
libbusiness-onlinepayment-payconnect-perl_XXX_all.deb
libmasonx-request-withapachesession-perl_XXX_all.deb
libtemplate-plugin-calendar-simple-perl_XXX_all.deb
```

Si spostino le cartelle jigdo scaricate al livello superiore possibile più vicino alla cartella radice.

L'NTFS filesystem ha un limite di 255 caratteri sul percorso di un file. Quando jigdo-lite scarica file da Internet fa la copia di una struttura di una cartella del server nelle sue cartelle di scaricamento. Con i loro lunghi nomi i pacchetti Debian in esempio possono eccedere la lunghezza di percorso permesso che

porta ad avere messaggi errati come "Cannot write to `[very long pathname]` (No such file or directory)".

Alcune persone ora si potrebbero chiedere: perché jigdo-lite usa "--force-directories" che crea queste gerarchie di cartelle problematiche?

Alcune versioni di jigdo-lite non lo usavano, ma alcune persone hanno richiesto l'utilizzo di "--continue" per evitare che i file semi-scaricati siano ignorati e cancellati quando si interrompe e si riavvia jigdo-lite.

Sfortunatamente, esso porta ad un problema: i server Debian contengono diversi file nominati allo stesso modo (es. "root.bin") in diverse cartelle, e se si interrompe jigdo-lite "al tempo giusto" durante lo scaricamento, c'è alta probabilità che la ripresa dello scaricamento concateni i dati al file semi-scaricato sbagliato, guastandolo e facendo fallire l'intero scaricamento con jigdo.

### 6.3. Come far usare a jigdo il mio proxy server?

Editare `~/.jigdo-lite` (o `jigdo-lite-settings.txt` per la versione Microsoft Windows) in un editor di testi e trovare la linea che inizia per "wgetOpts". Le seguenti opzioni possono essere aggiunte a quella linea:

```
-e ftp_proxy=http://LOCAL-PROXY:PORT/
-e http_proxy=http://LOCAL-PROXY:PORT/
--proxy-user=USER
--proxy-passwd=PASSWORD
```

Naturalmente si sostituiranno i valori corretti per il proprio proxy server. Le ultime due opzioni sono necessarie soltanto se il proprio proxy usa l'autenticazione tramite password. Le opzioni devono essere aggiunte all'estremità delle linee dei wgetOpts prima del carattere finale '. Tutte le opzioni devono stare su una linea.

Alternativamente, sotto linux è possibile anche installare le variabili d'ambiente `ftp_proxy` e `http_proxy`, per esempio nel file `/etc/environment` o `~/.bashrc`.

### 6.4. Jigdo-lite si interrompe con un errore - ho trasferito tutti quei MB dal sistema inutilmente?

Se jigdo-file si interrompe dopo il trasferimento di un pezzo considerevole di contenuti della ISO, si avrà un grande file ".iso.tmp". Ci sono parecchie cose da provare per salvare il proprio trasferimento:



- Ricominciare il trasferimento premendo **RETURN**. Forse alcuni file non si sono potuti trasferire a causa del timeout o di altri errori transitori. Provare a trasferire ancora i file mancanti.
- Provare un server differente. Alcuni mirror di Debian sono leggermente fuori sincronizzazione, forse un mirror differente tiene ancora i file che erano stati cancellati da quello che si era specificato, oppure è stato aggiornato con dei file che non sono ancora presenti sul proprio mirror. Questo mi è accaduto più volte.
- Richiamare le parti mancanti dell'immagine usando rsync (<http://rsync.samba.org>). In primo luogo, trovare l'URL corretto del rsync dell'immagine che si sta trasferendo: scegliere un server che offre l'accesso rsync a immaginistable (<http://www.debian.org/CD/mirroring/rsync-mirrors>) o testing (<http://www.debian.org/CD/http-ftp/#testing>) determinare il percorso ed il nome dei file corretti. Gli elenchi delle cartelle possono essere ottenuti con comandi come **rsync**  
**rsync://cdimage.debian.org/debian-cd/.**

Poi rimuovere l'estensione ".tmp" dal file temporaneo di jigdo-lite rinominandolo e passando sia l'URL remoto che il nome del file locale a: rsync: **rsync rsync://server.org/path/binary-i386-1.iso binary-i386-1.iso** Si potrebbe aver bisogno di usare le opzioni di rsync `--verbose` e `--progress` per ottenere i messaggi di stato e `--block-size=8192` per aumentarne la velocità.

- Sotto linux, si può montare con il parametro `loop` il file `.tmp` per accedere ai pacchetti che già sono stati trasferiti e riutilizzarli per generare un'immagine da un file `.jigdo` più recente. Per fare questo prima digitare i seguenti comandi come utente root nella cartella con il trasferimento interrotto: **mkdir mnt; mount -t iso9660 -o loop \*.tmp mnt**. In seguito iniziare un nuovo trasferimento in una cartella differente e inserire il percorso della cartella "mnt" alla richiesta "File to scan".

Sotto Microsoft Windows si può fare la stessa montando l'immagine ISO temporanea usando software per gestire "drive virtuali". Daemon tools (<http://www.daemon-tools.cc>) e Nero Image Drive sono entrambi molto popolari. Vedere anche <http://tinyurl.com/c39zr> per ulteriori opzioni.

## 6.5. [11 Agosto 2002]: Perché non ci sono le traduzioni di questo HOWTO su LDP?

Sto avendo difficoltà ad ottenere le traduzioni di questo HOWTO presentate ai redattori non inglesi di LDP.

L'editore tedesco di LDP, Marco Buddè <[Budde@tu-harburg.de](mailto: Budde@tu-harburg.de)> rifiuta di accettare la traduzione tedesca perché è stata scritta in Docbook e non in Linuxdoc anche se Dockbook è il linguaggio SGML preferito per LDP. È una vergogna che ci sia della gente della comunità open source che vuole sabotare la nostra comunità dall'interno.

L'editore portoghese di LDP, Alfredo Carvalho <[ajpc@poli.org](mailto: ajpc@poli.org)>, ha completamente ignorato il mio invio della traduzione portoghese.

Se si tiene ad avere documenti di LDP in queste due lingue si è invitati a scrivere a questi editori e chiedere loro di essere più responsabili nell'accettare i documenti tradotti. Per il momento si possono scaricare queste traduzioni dal mio sito web personale <http://www.dirac.org/linux/debian/jigdo>.

vergognati, Marco Budde <Budde@tu-harburg.de>.

vergognati, Alfredo Carvalho <ajpc@poli.org>.

## 6.6. Che cosa faccio se il trasferimento con jigdo viene interrotto?

Se il proprio trasferimento viene interrotto, tutto ciò che si deve fare è riavviare il jigdo-lite e premere **ENTER** a tutte le domande di richiamo. Jigdo-lite riprenderà da dove ha terminato.

## 6.7. Il mio trasferimento con jigdo non si completerà perché il file .jigdo è corrotto. Quando scarico un nuovo file corretto .jigdo, devo trasferire di nuovo tutti i dati ?

È possibile riscontrare che il jigdo che si è trasferito è corrotto. È raro, ma accade di tanto in tanto con prodotti in evoluzione come Debian testing o unstable.

Se si riscontra che il .jigdo è corrotto si dovrà trasferire un nuovo file .jigdo (quando uno corretto è disponibile), ma non ci sarà bisogno di trasferire ancora tutti i dati dell'immagine ISO.

È possibile usare lo stesso trucco di montare l'immagine ISO con il parametro loop usato per l'aggiornamento dell'immagine. La differenza è che non c'è nessuno file .iso completo con cui ricominciare, ma anche il file iso.tmp è un'immagine ISO e può essere usata per finire il trasferimento senza dover ritrasferire tutti i dati che sono stati trasferiti prima che il file jigdo corrotto causasse l'arresto del jigdo-lite. Semplicemente si monta con loop il file iso.tmp sulla cartella /mnt e quando si riavvia jigdo-lite, con il corretto file .jigdo, si specifica /mnt /mnt come directory da scandire. Non dimenticarsi di rinominare o spostare il file iso.tmp in modo che esso non interferisca con jigdo-lite il quale vorrà creare un nuovo file iso.tmp.

## 6.8. Posso usare il jigdo per trasferire le immagini per DVD?

Assolutamente; il processo è identico al trasferimento delle immagini CD. L'unica cosa differente da fare sarà di trasferire i file .jigdo e .template relativi ai DVD invece che ai CD. È possibile trovare i file .jigdo e .template relativi ai DVD all'indirizzo <http://www.debian.org/CD/jigdo-cd/>.

Su linux, c'è bisogno del kernel 2.4 o successivo per creare file di dimensione come quella dei DVD.

Su MS Windows, è necessario usare `jigdo-win-0.7.1a` (rilasciato il 21 luglio 2004) o successivo per creare immagini di dimensione come quella dei DVD. Ciò succede a causa di un bug nel supporto ai file di grandi dimensioni del Mingw32, il compilatore usato per creare gli eseguibili di MS Windows. Il bug è stato riparato in questa data e la versione `jigdo-win-0.7.1a` è stata rilasciata.

## 6.9. Posso masterizzare il file `.iso.tmp` in un CD?

Grazie a Gordon Huff e David Anselmi, ora sappiamo che la risposta è "si si può". Ma, molto importante, Gordon ha dato una buona ragione per non volerlo fare. Parafrasando Gordon:

Il Win98 di un mio amico ha una \*buona\* connessione cablata. Io arrivo la mattina, faccio partire jigdo (più di uno, attualmante) e dopo andiamo in negozio, allacciamo la parte posteriore della pianta del kiwi, mettiamo sopra le luci e l'albero di Natale, tagliamo l'albero, ordiniamo e dividiamo la pizza e accendiamo l'antenna parabolica.

Poi lascio i miei amici con diverse `iso.tmp` su CD-RW. Quando torno a casa, uso le ISO che non sono finite per aggiornare la mia installazione jigdo con una connessione dial-up.

## 6.10. Jigdo-lite è corrotto! Trasferisce i pacchetti e li cancella. Io mi accorgo che non li scrive nel file `.iso.tmp` perché la dimensione del file non è cambiata!

Jigdo lavora molto bene -- il file `.iso.tmp` è creato all'inizio con la relativa dimensione finale, ma riempito con zero byte. Dopo, parti di esso sono sovrascritte con i dati scaricati.

Si può dire che jigdo procede guardando i messaggi "Found X of the Y files required by the template" che sono stampati di tanto in tanto. Il primo valore "X" dovrebbe aumentare. Quando X è uguale a Y, il download è finito.

## 6.11. Sto avendo difficoltà per far lavorare jigdo-easy.

Guardare la Sezione 7.1.

## 6.12. Per aggiornare le immagini voglio che jigdo-lite faccia la scansione di 14 immagini montate con loop, in una volta.

## Come posso fare?

Quando si aggiornano le immagini dei CD è noioso montare con loop e smontare le immagini. Tuttavia, per default il kernel Linux supporta soltanto otto periferiche montate con loop, e il menu di jigdo-lite dei percorsi precedentemente inseriti ha soltanto cinque voci.

Per fare lo scan di molte immagini montate con loop si deve in primo luogo dire al kernel Linux di supportare più delle otto periferiche di default. Questo viene fatto con il parametro "max\_loop" quando si carica il modulo, con "modprobe loop max\_loop=16" come comando di linea o aggiungendo la linea "options loop max\_loop=16" al file /etc/modules.conf. In Debian, si deve mettere questa linea in un file chiamato /etc/modutils/local-loop e dopo eseguire **update-modules** perché cambiamenti diretti a /etc/modules.conf saranno sovrascritti.

Avendo montato individualmente le immagini, è possibile passare a jigdo-lite la cartella padre dei punti di montaggio per effettuare la scansione. Per esempio, se le immagini sono montate sotto /mnt/myloopmounts/imagel/ etc., passare "/mnt/myloopmounts" come percorso da scansionare. Se passare la cartella padre è sconveniente è possibile anche creare una directory e riempirla con collegamenti ai punti di montaggio.

### 6.13. Jigdo-lite è troppo prolisso. Come posso eliminare qualcuno o tutti i suoi messaggi?

Jigdo-lite usa wget, e l'output di wget può essere ridotto. Se questo è sconvolgente, è possibile fare in modo che wget sia zittito aggiungendo --non-verbose all'opzione wgetOpts del proprio file ~/.jigdo-lite. Se si vuole che wget non visualizzi alcun messaggio si dovrà usare --quiet all'opzione wgetOpts.

### 6.14. Posso usare jigdo su altre piattaforme diverse da Linux?

Certamente. Se si è interessati a Potato o Woody, sotto Microsoft Windows, vecchio SunOS, HP-UX e IRIX è possibile usare jigdo-easy. Vedere la Sezione 7.1 e la Sezione 7.4.

Se si vuole scaricare Potato, Woody, Sarge o Sid sotto Microsoft Windows, jigdo-lite è stato tradotto a questa piattaforma e può essere scaricato dal sito principale di jigdo (la Sezione 7.4).

### 6.15. Su MS Windows, perché ottengo il messaggio d'errore "No such file or directory"?

Si può verificare che sotto MS Windows, jigdo-lite scaricherà qualche file, ma poi fallisce la lettura del relativo contenuto e si produrrà un messaggio d'errore "No such file or directory".

Sembra che questo succeda se la lunghezza dei nomi dei file che jigdo processa supera un certo limite. La soluzione è di spostare quanto scaricato su una cartella superiore, più vicino alla directory principale del drive.

## 6.16. Su MS Windows, perché la mia immagine non cresce oltre i 2 GB?

Si sta usando una vecchia versione di jigdo. Per piacere aggiornarlo al `jigdo-win-0.7.1a` o successivo. Vedere la Sezione 6.8.

## 6.17. Su MS Windows, `jigdo-lite.bat` fallisce con il messaggio d'errore "sh" non trovato.

Questo significa che il `PATH` nel file `.bat` è errato. Per alcune ragioni, questo è il caso in cui si avvia jigdo su una condivisione di rete Windows usando come percorso `\\UnServer\File\jigdo`. Soluzione: usare "Connetti Unità di Rete" (nel "menu strumenti di esplora risorse") per assegnare una lettera al drive come "z:", dopo doppio clic sul file `.bat` da `z:\jigdo`. In alternativa un'altra soluzione è spostare tutto il contenuto della sottocartella `jigdo-bin` dove risiede il file `.bat`.

## 6.18. Posso eseguire istanze multiple di jigdo-lite per scaricare immagini in parallelo?

Assolutamente, comunque per evitare contrasti con i nomi dei file, si dovrebbe avviare ogni istanza di jigdo-lite in una propria cartella separata. Si possono avviare quante istanze si vuole, andare a letto, e quando ci si sveglia tutte le immagini ISO aspetteranno sul proprio hard disk. Considerare che jigdo-lite consuma intensivamente banda e CPU, quindi non si vorrà usare il proprio computer con istanze multiple eseguite in tandem.

## 6.19. C'è un interfaccia GUI disponibile?

Si sta lavorando un interfaccia GTK+ per jigdo. Entrambi i client GUI di Linux e Microsoft Windows sono progettati. Sfortunatamente è stato fatto l'80% in oltre un anno e mezzo, quindi non sperare molto per un prossimo rilascio.

## 7. Errata

### 7.1. jigdo-easy

Jigdo-easy, di Anne Bezemmer, è una diramazione di jigdo-lite che è compatibile con una vasta gamma di sistemi, incluso Microsoft Windows, vecchio SunOs, HP-UX e IRIX. È anche più facile da usare rispetto a jigdo-lite, ma a causa dei cambiamenti fatti a Jigdo funzionerà solo con Potato e Woody. Jigdo-easy non sarà capace di scaricare Sarge e Sid. Vedere la Sezione 7.4 ed la Sezione 6.14.

### 7.2. Altro riguardo le fonti di scansione

Per ora si sa che, quando jigdo-lite chiede i file per la scansione, è possibile utilizzare 3 fonti:

- Una copia montata di un CD o DVD scaduto che si vuole aggiornare.
- Una copia montata con loop di un'immagine ISO scaduta presente sull'hard disk.
- Una copia montata con loop del file provvisorio `.iso.tmp`, quando una esecuzione precedente di jigdo-lite si è interrotta.

Come mette in evidenza Jens Sedel c'è un'altra fonte piuttosto efficiente che si potrebbe usare per una fonte di scannerizzazione: la propria cache di apt. Apt usa la directory `/var/cache/apt/archives` come cache. Ci saranno molti pacchetti di Debian in questa cartella, ed essi possono essere usati come sorgente di scansione per jigdo-lite! Quando verrà chiesta una cartella da scandire è possibile usare anche questa cartella.

Se si sta editando manualmente il file `~/jigdo-lite` fare attenzione che le cartelle per la scansione multipla siano separate da spazio, ad esempio:

```
scanMenu=' /var/cache/apt/archives/ /cdrom/'
```

### 7.3. jigdo-file-cache.db

La cache contiene le md5sum dei file letti quando si fornisce una cartella al prompt `Files to scan:`. Se jigdo-lite scandisce una cartella una seconda volta la scansione sarà molto veloce.

Ciò sarebbe utile nei casi seguenti: rev0 si aggiorna al rev1. Con le immagini CD di rev1 alcuni pacchetti possono essere spostati da  $CD_n$  a  $CD_{n+1}$ , o viceversa. Se si ha un collegamento particolarmente lento (es: Modem), si cercherà di evitare di scaricare di nuovo questi pacchetti. Per questa ragione quando si

scaricano nuove versioni di CD  $n$ , si dovrebbe far scannerizzare al jigdo-lite i 3 CD  $n-1$ ,  $n$  e  $n+1$  (oppure tutti e 8 i CD per essere sicuri al 100%).

Se jigdo-lite ha scandito gli stessi CD ripetutamente, durante l'aggiornamento delle 8 immagini CD, la cache eviterà che tutti i dati dei CD vengano letti più volte.

La cache è molto più importante quando *generano* i file jigdo perché non si vuole che jigdo-file legga nell'intero mirror Debian da 50GB per ogni file jigdo generato.

## 7.4. Risorse

Questo HOW TO si sta per concludere, ma penso che dovrò lasciarvi con un po' di link di riferimento per imparare di più riguardo agli strumenti jigdo e come essi lavorano.

<http://atterer.net/jigdo>

Questo è il sito principale di Jigdo. Si dovrebbe visitare questo sito, un po' di informazioni riguardo le porte, client GUI e altro riguardante jigdo.

<http://cdimage.debian.org/~costar/jigdo>

La pagina di Debian per jigdo-easy (la Sezione 7.1).

<http://www.debian.org/CD/jigdo-cd>

La pagina principale di Debian per jigdo.

<http://packages.debian.org/testing/utils/jigdo-file.html>

La pagina web ufficiale per i pacchetti di Debian jigdo-file.

<http://lists.debian.org/search.html>

Si può usare questa pagina per cercare gli archivi della mailing list di debian-cd.

<http://www.debian.org/MailingLists/subscribe>

La pagina di registrazione della mailing list di debian-cd.

<https://lists.berlios.de/mailman/listinfo/jigdo-user>

La pagina ufficiale di registrazione alla mailing list di Jigdo.