

Linux Base - Capitolo n. 9

Edizioni ByteMan (07-11-2005)

revisione: 25/01/2008

Pacchetti (parte 1)

Il processo di installazione non si esaurisce con la memorizzazione del sistema operativo e del software di base sulle partizioni prescelte dell'hard disk, ma continua in seguito con l'aggiunta o con la sostituzione di software più moderno. Un tale lavoro di manutenzione, effettuato chiaramente con i diritti di amministratore, viene agevolato dalla presenza di opportuni strumenti che provvedono al controllo di tutta una serie di operazioni intermedie indispensabili affinché il nuovo software, pacchettizzato (file con suffisso **.deb**), arrivi correttamente a destinazione.

Per le distribuzioni Debian e derivate (Knoppix, Kanotix, Mepis, Xandros, Lindows/Linspire, Lycoris,...) sono disponibili, per chi inizia, almeno due possibilità: **dpkg** e **apt**. Sicuramente più popolare e più immediato nell'uso, **apt** (**a**dvanced **p**ackage **t**ool) sfrutta il sottostante livello costituito da **dpkg** (**d**ebian **p**ackage).

Occorre tenere presente che tutte le distribuzioni Debian si sviluppano seguendo un percorso a tre stadi: nascita (unstable), maturazione (testing), rilascio (stable).

■ **Unstable:**

Questo tipo di release contiene tutti i pacchetti nuovi che, non essendo stati ancora ben verificati sul campo, possono creare problemi di malfunzionamento. Chi preferisce questa distribuzione lo fa a proprio rischio, ma sperimenta da subito le ultime novità in sviluppo. L'installazione è prevista solo tramite rete (Internet).

■ **Testing:**

I pacchetti di questa release provengono dalla fase unstable ed hanno superato una serie di verifiche che ne garantiscono un buon grado di stabilità. Di queste versioni vengono rilasciate periodicamente delle immagini ISO su CD o DVD identificate da numeri progressivi e chiamate **snapshot** (fotografie).

■ **Stable:**

Costituisce l'ultimo stadio del ciclo di vita di una distribuzione. Proviene dall'ultima *snapshot* che non ha subito più aggiornamenti da un lungo periodo di tempo. Viene anche chiamata **frozen** (congelata) ed è utilizzata soprattutto sui server, dove la stabilità è la caratteristica maggiormente richiesta, mentre per gli utenti di un normale desktop potrebbe risultare obsoleta.

*Una curiosità delle distribuzioni Debian è data dai nomi ufficiali che vengono loro assegnati, sino ad ora sono stati scelti nomi appartenenti ai personaggi del film **Toy story**.*

*La unstable si chiamerà sempre **Sid**, ma avrà un nuovo nome quando diverrà testing, e con lo stesso passerà a stable.*

*I nomi delle ultime distribuzioni stabili di Debian sono: **Potato** (ver. 2.2), **Woody** (ver. 3.0) e **Sarge** (ver. 3.1). L'attuale testing si chiama **Etch**.*

Tutti i pacchetti Debian, classificati nelle tre classi di sviluppo, risiedono su dei server chiamati **repository** dai quali è possibile scaricarli ed installarli rispettandone le dipendenze.

Il file `/etc/apt/sources.list` contiene l'elenco di repository che viene usato dalla propria macchina.

APT

L'apt-system e' un insieme di script che si appoggia al comando dpkg e che si propone di facilitare la gestione di pacchetti e dipendenze nei sistemi Debian GNU/Linux. Notoriamente chi prende confidenza con Debian difficilmente sara' tentato di cambiare distribuzione, e questo anche grazie a questo splendido set di tools. Senza la pretesa della completezza questa brevissima panoramica ha

lo scopo di fornire un piccolo vademecum per mantenere aggiornato il nostro sistema operativo.

```
apt-setup
```

serve per aggiungere/aggiornare le fonti da cui scaricare i pacchetti

Il sistema di gestione dei pacchetti usa un suo database per tenere traccia di quali pacchetti sono installati, quali non lo sono e quali sono disponibili per l'installazione. Il programma `apt-get` usa questo database per capire come installare i pacchetti richiesti dall'utente e per scoprire quali ulteriori pacchetti sono necessari per farli funzionare correttamente.

```
apt-get update
```

aggiorna le informazioni disponibili sui pacchetti, con riferimento ai repository citati in `/etc/apt/sources.list`. Da utilizzare, con una certa regolarità, prima di procedere ad aggiornamenti con `apt-get install` oppure `apt-get upgrade`, questo comando mantiene il sistema informato su possibili aggiornamenti dei pacchetti, in particolare, sugli aggiornamenti per la sicurezza.

```
apt-get install [pkg_1, pkg_2,... pkg_n]
```

e' il comando che ci permette di installare/aggiornare uno o piu' pacchetti precompilati, tentando (e riuscendoci!) di risolvere automaticamente tutte le dipendenze richieste.

```
apt-get install [pkg_name] -s
```

E' anche possibile simulare (con output su video) l'installazione di un pacchetto per controllare se ciò che stiamo installando/aggiornando modifichera' altri pacchetti già installati.

```
apt-get remove [pkg_1, pkg_2,... pkg_n]
```

e' l'esatto opposto del comando `apt-get install`.

```
apt-cache search [stringa]
```

usando questo comando possiamo cercare pacchetti in base al nome, alla descrizione, ecc ...

```
apt-cache show [pkg_name]
```

questo comando visualizza una breve descrizione del pacchetto da noi richiesto, fornendo versione, uso, dipendenze e altre utili informazioni.

```
apt-get source [pkg_name]
apt-get source [pkg_name] --compile
```

il primo comando da utilizzare nel caso volessimo installare i sorgenti di un pacchetto.
il secondo comando e' il passo finale per la compilazione di un pacchetto **.deb**, da lanciare dalla stessa directory in cui si sono scaricati i sorgenti da compilare.
NOTA BENE: i sorgenti così scaricati, una volta compilati, daranno origine ad un pacchetto **.deb** che andrà a sua volta installato!

```
apt-get build-dep [pkg_name]
```

questo e' un pacchetto fondamentale, specie se si lavora molto con i sorgenti. Si occupa di soddisfare le dipendenze del pacchetto che andremo a compilare dopo aver eseguito `apt-get source`.

```
apt-cdrom
```

aggiunge alle fonti per i pacchetti il contenuto di un CD debian.

```
apt-get upgrade
```

aggiorna i pacchetti installati sul nostro sistema, se sono disponibili versione piu' recenti. Da utilizzare con molta cautela perchè può coinvolgere parecchio software e provocare un certo grado di interattività con l'utente (potrebbe essere necessario rispondere a delle domande poste prima di effettuare delle scelte).

```
apt-get dist-upgrade
```

e' simile al precedente, ma aggiorna anche scripts di configurazione ed altro, valgono le stesse raccomandazioni.

```
apt-get clean
```

quando i pacchetti vengono scaricati ne viene memorizzata una copia in **/var/cache/apt/archive/** e quindi col passare del tempo questa directory tende a ingrandirsi notevolmente. Usare questo comando per cancellare il contenuto della directory ed evitare una cancellazione manuale che altrimenti disorienterebbe il sistema apt che continuerebbe a credere di avere una copia dei pacchetti in locale.

```
apt-get autoclean
```

da non confondere con il precedente, procede ad eliminare da **/var/cache/apt/archive/** solo le versioni obsolete dei vari pacchetti presenti.

DPKG

Costituisce il fondamento del sistema di gestione dei pacchetti Debian e permette la loro manipolazione diretta. Se sul disco fisso c'è un pacchetto di nome **myprog-1.2.-1.deb** e si desidera installarlo basta dare il comando:

```
dpkg -i myprog-1.2.-1.deb
```

Anche un pacchetto **.deb** scaricato da internet può essere installato direttamente in questo modo.

Per rimuovere lo stesso programma è sufficiente indicarne il nome con il comando:

```
dpkg -r myprog
```

che rimuove unicamente i binari (il programma stesso) ed i relativi script di avvio, ma non cancella nessuno degli eventuali file di configurazione. Reinstallando il programma in un secondo momento, verranno quindi usati i file di configurazione esistenti, evitando di doverli ricreare da zero. La rimozione (disinstallazione) di un pacchetto richiede che nessun altro pacchetto dipenda da quello che vogliamo rimuovere, nel qual caso dovremo prima rimuovere quelli da esso dipendenti.

Per ottenere l'eradicazione totale del programma (**P**urging): binari, script di avvio, file di configurazione, tutto, è necessario digitare:

```
dpkg -P myprog
```

Questo comando si occupa di rimuovere qualsiasi cosa legata al programma, ma nessun altro pacchetto deve dipendere da ciò che vogliamo rimuovere, altrimenti occorrerà rimuovere prima i pacchetti dipendenti. Ovviamente in caso di reinstallazione si dovrà procedere ad una nuova configurazione.

```
dpkg -l myprog
```

Mostra informazioni su versione e stato di un pacchetto installato. Ad esempio, nel caso di *nasm* potrebbe essere visualizzato quanto segue:

```
Desiderato=sconosciUto/Installato/Rimosso/P:eliminato/H:bloccato  
| Stato=Non/Installato/file Config./U:spacchett./conf. Fallita/H:inst.parzial.
```

```
|/ Err?=(nessuno)/H:bloc./necess.Reinst./X=entrambi (Stato,Err: maiusc.=grave)
||/ Nome           Versione           Descrizione
+++-----
ii  nasm           0.98.38-1.2       General-purpose x86 assembler
```

Nella tabella i primi tre caratteri "ii " informano sullo Stato dell'installazione e su eventuali Errori occorsi durante la stessa. Segue il nome dell'applicativo, la versione ed una breve descrizione. Omettendo il nome del programma è possibile ottenere la lista completa di tutti i pacchetti installati.

```
dpkg -L myprog
```

Elenca tutti i files relativi al pacchetto installato. Se si desidera scoprire dove sia stato posizionato tutto il materiale relativo ad un determinato pacchetto questo è il comando per scoprirlo. Sempre nel caso di **nasm** si potrebbe vedere quanto segue (abbreviato per motivi di spazio):

```
.....
/usr/bin/nasm
/usr/bin/ndisasm
/usr/bin/rdfdump
/usr/bin/ldrdf
.....
/usr/share/doc/nasm
/usr/share/doc/nasm/AUTHORS
/usr/share/doc/nasm/README
/usr/share/doc/nasm/html/nasmdoc0.html
/usr/share/doc/nasm/html/nasmdoc1.html
/usr/share/doc/nasm/html/nasmdoc2.html
.....
/usr/share/doc/nasm/examples/a32offs.asm
/usr/share/doc/nasm/examples/absolute.asm
/usr/share/doc/nasm/examples/aoutso.asm
/usr/share/doc/nasm/examples/aouttest.asm
/usr/share/doc/nasm/examples/aouttest.c
.....
/usr/share/doc-base/nasm-manual
.....
/usr/bin/rdf2com
.....
```

```
dpkg -S fileXYZ
```

Trova il pacchetto che contiene il file *fileXYZ*. Molto utile se si desidera conoscere a quale pacchetto è legato un determinato file.

```
dpkg -s myprog
```

Visualizza un quadro riassuntivo del pacchetto installato. Per esempio nel caso di **nasm** si avrà:

```
Package: nasm
Status: install ok installed
Priority: optional
Section: devel
Installed-Size: 2776
Maintainer: Christian Kesselheim
Architecture: i386
Version: 0.98.38-1.2
Depends: libc6 (>= 2.3.2.ds1-21)
Description: General-purpose x86 assembler
 Netwide Assembler.  NASM will currently output flat-form binary files,
 a.out, COFF and ELF Unix object files, and Microsoft 16-bit DOS and
 Win32 object files.
.
```

```
Also included is NDISASM, a prototype x86 binary-file disassembler
which uses the same instruction table as NASM.
```

```
.
NASM is released under the GNU Lesser General Public License (LGPL).
```

Alien

Il formato utilizzato da Debian non è, purtroppo, l'unico disponibile per archiviare pacchetti per Linux. Uno dei più noti, e storicamente uno dei primi, è il formato rpm, utilizzato nativamente da molte distribuzioni (RedHat, Mandrake/Mandriva, Suse, ...). Questo formato dispone dei **suoi** strumenti di installazione, il più noto dei quali si chiama proprio **rpm** (**red-hat package manager**). Quando si utilizza una distribuzione Linux, è quantomeno fastidioso e non molto raccomandato dover mescolare applicazioni installate a partire da archivi in formato diverso da quello che si usa normalmente. Ciò proprio perché non è più possibile tenere traccia, in un modo univoco, della posizione dei file appartenenti a ogni pacchetto (senza contare le altre conseguenze).

Fortunatamente vengono in aiuto i programmi di conversione che permettono di trasformare un archivio da un formato a un altro, anche se non sempre funzionano perfettamente. A questi si affiancano poi degli applicativi che permettono di ispezionare il contenuto di file impacchettati in vari formati, e di estrarne quello che si desidera.

Alien è un programma che consente di convertire un pacchetto archiviato in un altro formato di archiviazione. Precisamente, è in grado di generare archivi in formato Debian, Red Hat, Stampede e Slackware, a partire da questi formati e anche da un semplice archivio tar+gzip. Non è in grado di gestire i pacchetti sorgenti. L'eseguibile che compie tutto il lavoro è **alien**, e la sintassi per il suo utilizzo è mostrata nello schema seguente:

```
alien --to-deb [<opzioni>] <file-da-convertire>...
alien --to-rpm [<opzioni>] <file-da-convertire>...
alien --to-tgz [<opzioni>] <file-da-convertire>...
alien --to-slp [<opzioni>] <file-da-convertire>...
```

Alien ha la necessità di conoscere soltanto in quale formato finale occorre produrre la conversione. Il tipo di archivio sorgente viene individuato automaticamente, probabilmente in base all'estensione usata nel nome del file. Se con le opzioni non si specifica in quale formato convertire, si ottiene un archivio Debian, per esempio:

```
alien file-da-convertire.rpm
```

genera automaticamente il corrispondente pacchetto **deb**. Ovviamente è possibile anche la conversione in senso opposto.

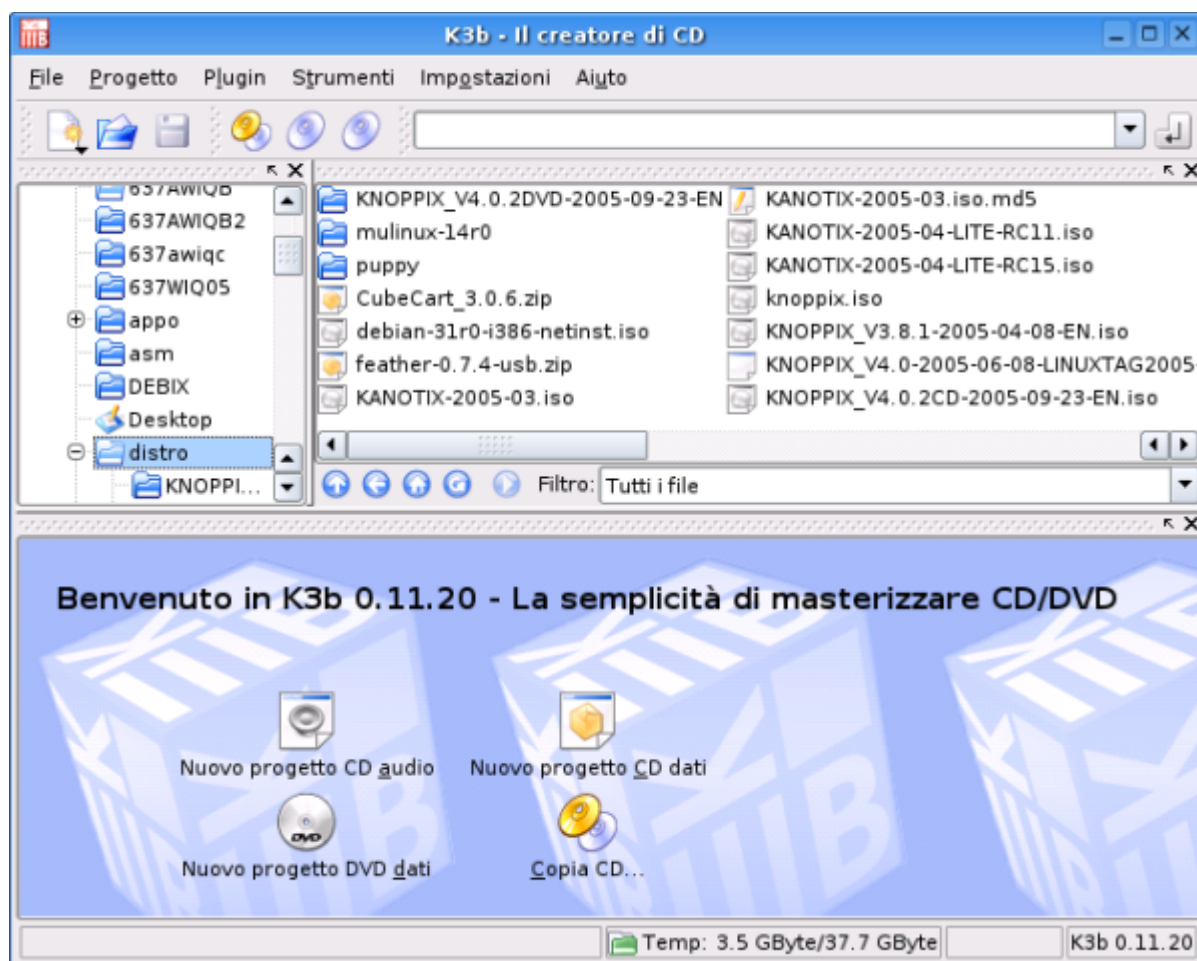
Una volta ottenuto il pacchetto **deb** basta usare il comando **dpkg -i**, già visto in precedenza, per procedere alla effettiva installazione dello stesso.

Applicativi (parte 3)

Il trattamento di dispositivi multimediali è una delle attività che viene svolta sempre più frequentemente in qualsiasi campo dell'informatica. Linux mette a disposizione i suoi tools, con interfaccia grafica o a linea di comando, ma sempre open source. In questa sezione ci occuperemo di presentare alcuni di questi strumenti.

K3B

E' una delle tante interfacce grafiche per masterizzare sotto linux, si appoggia infatti ad apposito software presente nel sistema operativo. Il suo uso è abbastanza semplice ed intuitivo e si diventa immediatamente operativi. Ad una prima occhiata può sembrare molto scarno, però fa solo quello che deve fare e lo fa bene: masterizza CD e DVD senza troppi fronzoli e funzioni ridondanti.



Si tenga presente che prima di cominciare ad utilizzare un qualsiasi applicativo per la masterizzazione è importante configurare il proprio sistema affinché riconosca correttamente la periferica hardware disponibile. Al giorno d'oggi quasi tutte le distribuzioni Linux riescono a riconoscere perfettamente ogni periferica ma è buona norma, nel caso in cui qualcosa non dovesse andare a buon fine, conoscere il modo per configurare correttamente la propria Linux box col supporto per la masterizzazione.

Xine

Si tratta di un player video dalle qualità invidiabili anche dagli analoghi prodotti dei sistemi operativi proprietari. Con Xine sarà possibile trasformare la propria GNU/Linux box in una vera e propria sala cinematografica per guardare un film. Sviluppato per sistemi Unix, distribuito con licenza GPL, supporta praticamente tutti i tipi di codifica (mpeg 1 e 2, mp3, avi, Ogg Vorbis ecc). Oltre a permettere la visione di file nei formati precedentemente elencati consente anche la visione diretta di VideoCD, SVCD e DVD. A causa delle

questioni legali legate alle problematiche di copyright e di blocco per regione dei DVD sarà necessario utilizzare xine in abbinamento a programmi esterni o plugin come xine_dvd_plugin che consentono la riproduzione di tutti i tipi di DVD. Alla base di Xine ci sono le librerie xine-lib e il suo engine, usato da molti altri player video a scopi di playback e video processing.



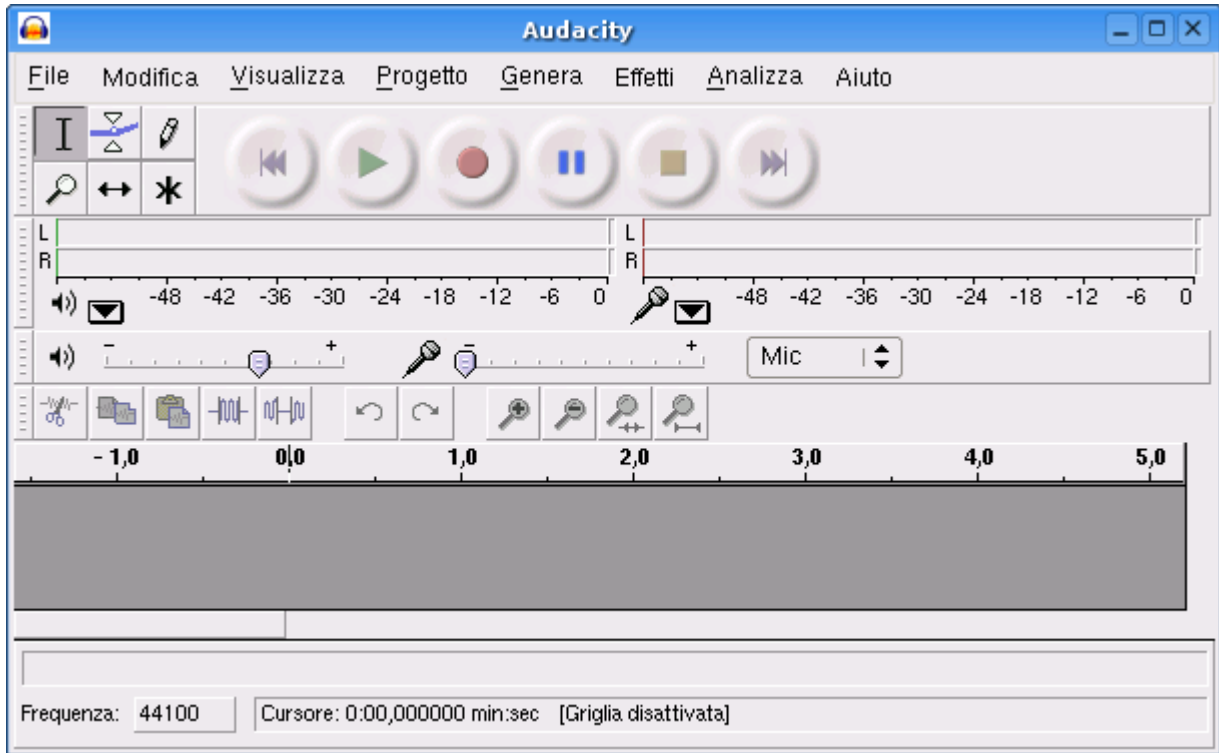
Per il funzionamento del software è necessario un sistema i386 compatibile di produzione recente (pentium o superiore) e un sistema operativo Linux o FreeBSD con installato X11. A seconda del tipo di video che si vorrà riprodurre potrebbe essere necessaria una maggiore potenza di calcolo per evitare di avere una visione a scatti. Considerando che con un Pentium II a 400 MHz è possibile vedere un DVD senza problemi di fluidità delle immagini, si può affermare che sui moderni personal computer equipaggiati con processori da qualche GHz non ci saranno problemi di riproduzione.

Audacity

E' un editor e registratore audio semplice e libero, disponibile per Windows, Mac OS X, GNU/Linux, e altri sistemi operativi. Si può usare Audacity, ad esempio, per:

- Registrare audio dal vivo
- Convertire dischi e musicassette in formato digitale oppure su CD
- Modificare file audio in formato Ogg Vorbis, MP3 e Wav
- Tagliare, copiare, dividere, e mixare file audio
- Cambiare la velocità o l'intonazione di una registrazione

Ma la lista completa delle sue funzionalità è ancora più estesa ed esula dallo scopo di questa semplice presentazione.



Il formato di registrazione è a 16 bit stereo con qualsiasi frequenza si desidera, vengono utilizzati gli ingressi della scheda audio. Importa i formati WAV, MP3, AIFF, AU, IRCAM, OGG. Esporta come WAV, OGG o MP3 se si è installato il LAME.

Le capacità di edit sono piuttosto avanzate e molto intuitive, sono disponibili: taglia, copia, incolla, mix tracce, analisi di spettro, amplificazione, effetti.

Importa anche file MIDI, limitandosi però a riprodurli senza dare la possibilità di editarli, ed è possibile aggiungere tutti i plug-ins VST.

Non si tratta, però, di un player e non funziona come applicazioni tipo Winamp. Ha lo svantaggio che non è possibile registrare più di due tracce separate, neppure disponendo di hardware dedicato.

 **Windows** Esiste anche nella versione per Windows e sostituisce, praticamente, Sound Forge del costo di oltre 250 Euro. La versione più recente è attualmente la 1.2.4b che si può scaricare dal sito di riferimento: <http://audacity.sourceforge.net>.

La procedura guidata di installazione (in inglese, anche se il programma è in italiano) propone valori adatti per la maggior parte degli utenti. Si possono impostare la qualità di compressione dei file .OGG prodotti in "File/Preferenze/Formati file". Un file Ogg Vorbis (.OGG) di qualità 3 è leggermente superiore a un file MP3 di qualità standard (128 kbit/s), e le dimensioni del file sono anche notevolmente più contenute. La qualità 5 produce file dello stesso livello di un CD audio.

Avendo installato le apposite librerie per l'esportazione MP3, si possono collegare con la funzione "Cerca Libreria" in "File/Preferenze/Formati file".

Audacity salva i file in formato **OGG Vorbis**; è un formato audio di **eccellente qualità** che sostituisce MP3 e non ha, al contrario di MP3, problemi di brevetti e **licenze d'uso**. Volendo salvare in MP3, occorre scaricare l'apposito modulo dalla stessa pagina da cui si scarica Audacity.

Il manuale di Audacity ed altra documentazione utile (in gran parte in inglese) possono essere scaricati da <http://audacity.sourceforge.net/help/documentation>.

Installazione (parte 4)

Un'altra prova sul campo

Andata a buon fine la prima installazione sul campo (Seminario 7), ecco presentarsi un altro collega con il suo bravo portatile da convertire a Linux. Si tratta questa volta di un non recentissimo Sharp su cui si dovrà reinstallare, ex novo, tutto. Sembra un lavoro più facile del precedente, ma ...

Caratteristiche del portatile

Modello Sharp
CPU Pentium III a 500 MHZ
Memoria RAM 128 MB
Lettore CD-ROM esterno collegato via USB (non bootabile)
Lettore floppy esterno collegato via USB (bootabile)
Hard disk da 12 GB
Windows XP già installato sull'intero disco con FAT32

La situazione che occorre affrontare prevede, quindi, un boot da floppy ed un partizionamento dell'hard disk senza il vincolo di preservare la preesistente installazione di Windows. Disponendo della connessione permanente ad Internet del laboratorio, si decide di utilizzarla per installare Debian. Si scaricherà prima solo una piccola parte di Debian, richiesta per iniziare il processo di installazione, e poi si installerà il resto mediante il programma di installazione. Questa modalità di installazione via rete richiede o una connessione dial-up PPP analogica ad un Internet provider o un accesso Internet via Ethernet (per i laptop usando possibilmente una scheda PCMCIA). Noi abbiamo utilizzato la seconda possibilità. Abbiamo quindi scaricato tre immagini di floppy disk (file con la dimensione di un floppy disk) dal sito:

http://ftp.ie.debian.org/debian-cd/3.1_r0a/i386/other/floppy/

ed, in particolare, le immagini:

1. **boot.img** come floppy di avvio
2. **root.img** contenente il file system (richiesto dopo il floppy di avvio)
3. **net-drivers.img** con i drivers per le schede di rete più note (se necessario)

Per realizzare i tre floppy a partire dai file immagine abbiamo utilizzato il comando **dd** che legge un flusso di dati da una sorgente e lo scrive in uscita con le modifiche eventualmente richieste. Il comando:

```
dd if=boot.img of=/dev/fd0 bs=1440k
```

ha creato il primo dei tre floppy, e in modo analogo si è proceduto per gli altri due floppy.

ATTENZIONE a ciò che si scrive: il comando **dd** non ammette ripensamenti e un comando sbagliato può distruggere definitivamente dati importanti.

Si noti che il parametro **if** consente di specificare il file/dispositivo di input, mentre il parametro **of** consente di specificare il file/dispositivo di output. E' infatti possibile l'operazione inversa, partendo da un floppy si può ricavare il relativo file immagine:

```
dd if=/dev/fd0 of=disk.img
```

Distribuzione utilizzata

A differenza dell'esperienza del seminario 7, abbiamo voluto utilizzare direttamente la distribuzione **Debian** capostipite delle varie **Knoppix**, **Kanotix**, etc. A causa della necessità di avviare da floppy, la quantità di software iniziale sarà limitatissima e quindi gli strumenti per la preparazione dell'installazione saranno esclusivamente testuali, essendo impossibile l'uso di strumenti ad interfaccia grafica.

Si comprende, in questo caso, quanto sia **indispensabile saper operare nelle due modalità: a linea di comando e con interfaccia grafica.**

Schema dell'installazione

1. Avvio con boot da floppy
2. Ripartizionamento dell'hard disk
3. Configurazione Installazione Base
4. Avvio Installazione Base
5. Riavvio computer e completamento installazione da rete
6. Riavvio computer e test finale

1 - Avvio con boot da floppy

A computer spento, inserire il floppy n.1 (boot), poi premere il pulsante/interruttore di avviamento e verificare che il boot stia avvenendo da floppy. Se così non fosse, riavviare, entrare in setup, preselezionare il boot da floppy, salvare e proseguire. Alla schermata di prompt è sufficiente premere invio ed attendere.

```
Welcome to Debian GNU/Linux 3.0!

This is the Debian Rescue disk. Keep it once you have installed your system,
as you can boot from it to repair the system on your hard disk if that ever
becomes necessary (press <F3> for details).

On most systems, you can go ahead and press <ENTER> to begin installation.
You will probably want to try doing that before you try anything else. If
you run into trouble or if you already have questions, press <F1> for
quick installation help.

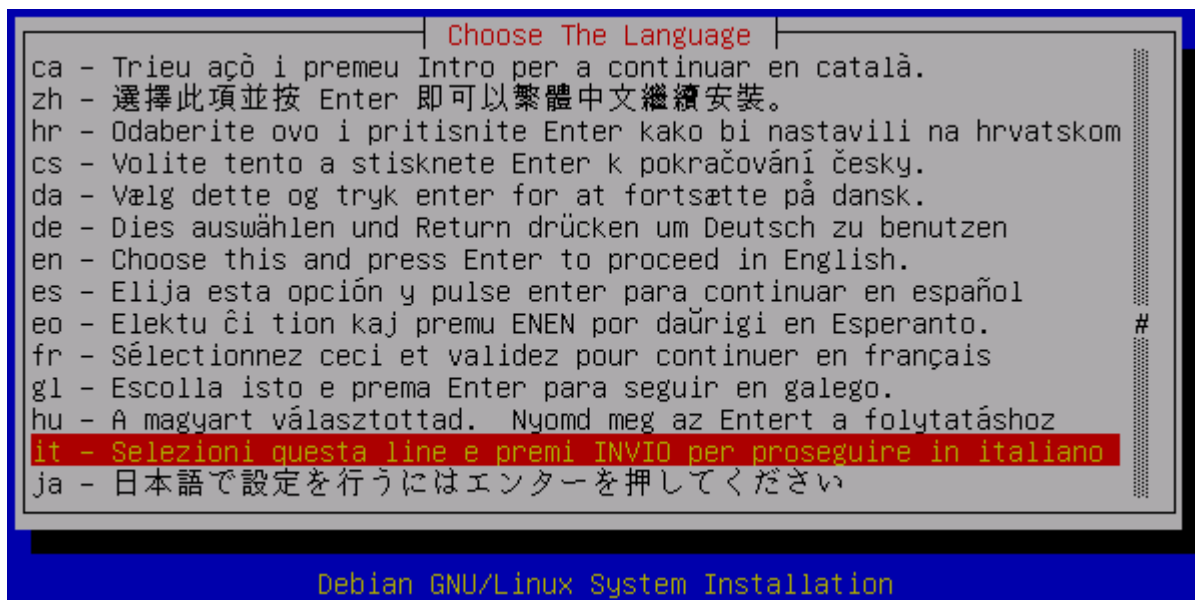
WARNING: You should completely back up all of your hard disks before
proceeding. The installation procedure can completely and irreversibly
erase them! If you haven't made backups yet, remove the rescue disk
from the drive and press <RESET> or <Control-Alt-Del> to get back to
your old system.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law. For copyright information, press <F10>.

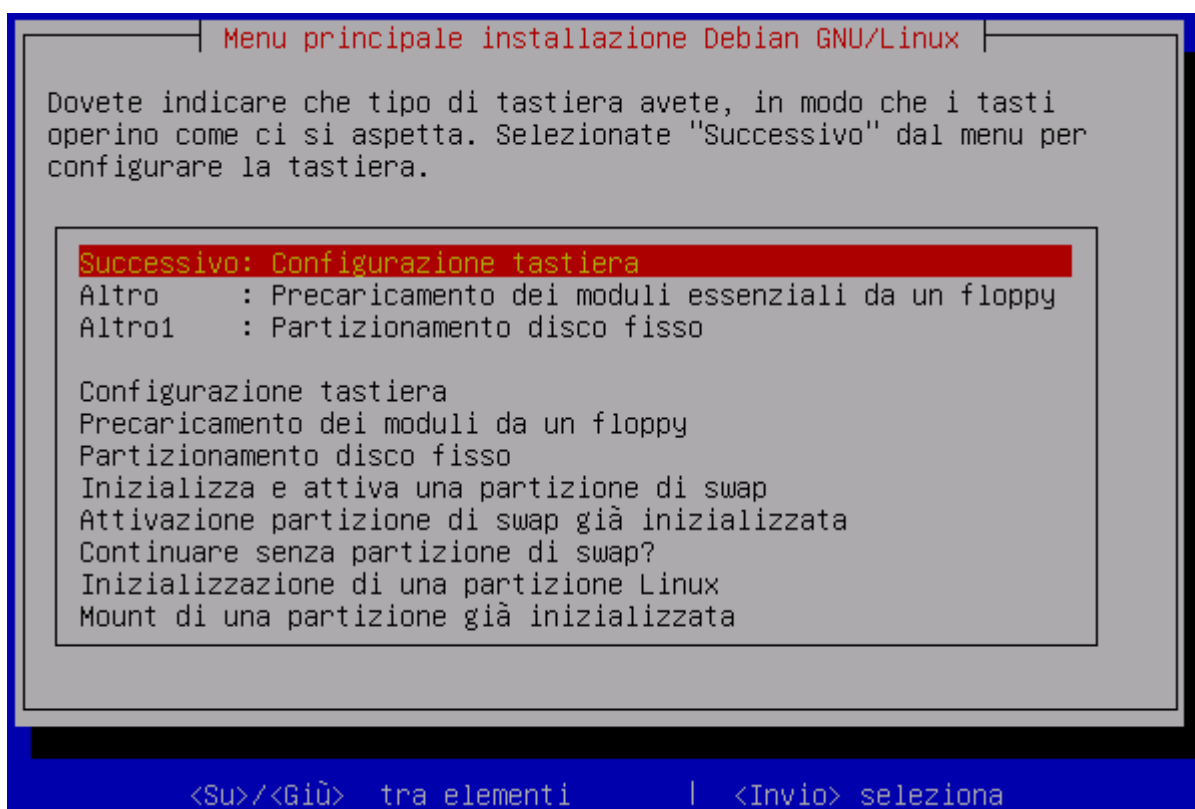
This disk uses the Linux kernel 2.4.18-bf2.4

Press <F1> for help, or <ENTER> to boot.
boot: _
```

Quando verrà richiesto il secondo floppy (root) inserirlo, premere ancora invio ed attendere istruzioni. Proseguendo e confermando si arriverà alla scelta della lingua con una schermata come la seguente:



Selezioneremo la lingua italiana come interfaccia con l'installatore, e proseguiremo quindi fino a quando ci verrà proposto il menù generale, dal quale per prima cosa selezioneremo la tastiera italiana:



2 - Ripartizionamento dell'hard disk

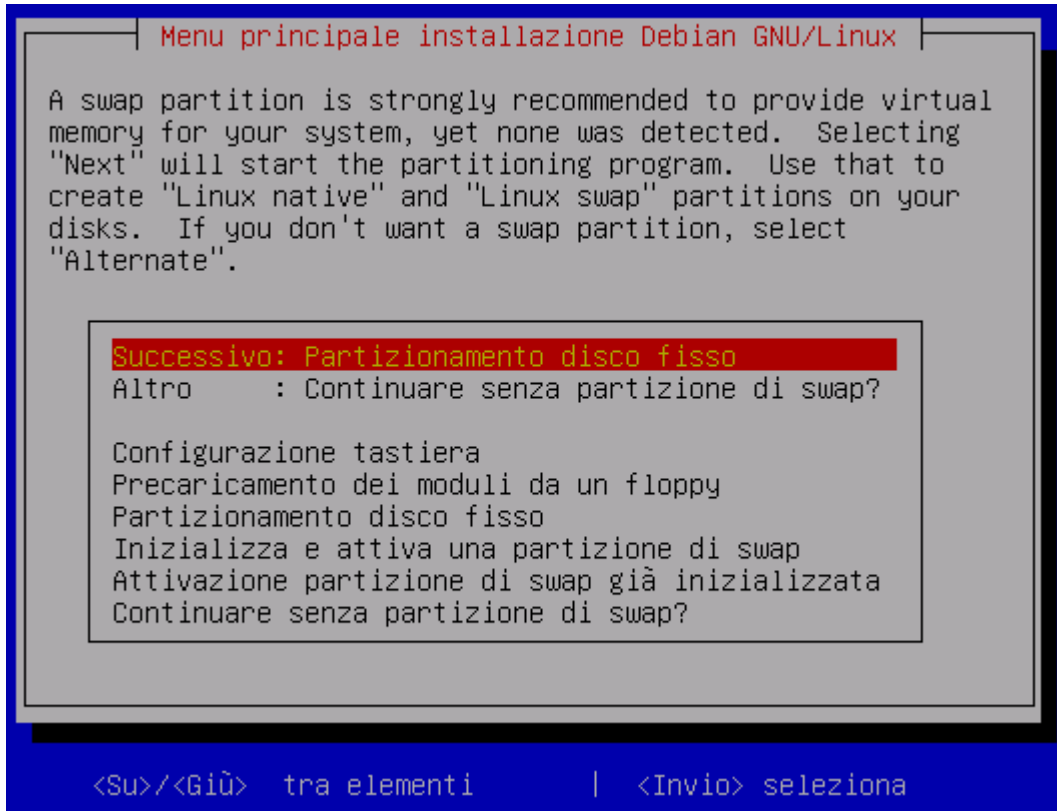
Abbiamo preselezionato la voce **Partizionamento del disco fisso** dal menù generale, quindi scelto il dispositivo, l'unico presentato, **/dev/hda**.

Abbiamo constatato la presenza di 2 partizioni preesistenti:

- **hda1** di circa 300 MB apparentemente non utilizzata
- **hda2** pari a 11,6 GB di tipo FAT32 contenente Windows XP

Si è deciso, momentaneamente, di non toccare la **hda1** e di ripartizionare invece la **hda2** per le necessità di Linux. Abbiamo realizzato quindi una partizione Reiserfs per Linux pari a circa 11,2 GB ed una partizione di swap pari a circa 400 MB.

Ecco una delle schermate presentatesi durante le fasi di partizionamento:



Alla richiesta di conferma la situazione venutasi a creare era la seguente:

- **hda1** di circa 300 MB momentaneamente non utilizzata
- **hda2** pari a 11,2 GB di tipo ReiserFS per Linux
- **hda3** di circa 400 MB di tipo swap per Linux

3 - Configurazione Installazione Base

Come al solito, prima di avviare l'installazione, verrà posta una serie di domande per configurare i principali parametri di utenza:

- Nome completo: **Caio Tizio** (Non ha nessuna importanza!)
- Nome Utente: **ctizio** (E' fondamentale: servirà per il Login.)
- Password Utente: (Sceglierne una appropriata.)
- Password Utente: (Ridigitarla per sicurezza.)
- Password Root: (Sceglierne una appropriata.)
- Password Root: (Ridigitarla per sicurezza.)
- Nome della macchina: **deb01** (E' il nome con cui verrà individuato il computer in rete.)
- Dove installare grub: **MBR** (Abbiamo deciso che *grub* fungerà da boot loader del sistema.)

A questo punto partirà la fase successiva di installazione vera e propria.

4 - Avvio Installazione Base

Questa procedura è tutta automatica e ci potremo concedere un quarto d'ora di pausa. Il tempo dell'installazione dipenderà anche dalla velocità del computer.

Quando la procedura sarà giunta al termine apparirà una schermata di congratulazioni per l'avvenuta installazione di Debian ad indicare che tutto si è concluso regolarmente.

5 - Riavvio computer e completamento installazione da rete

A questo punto non resta che spegnere il computer, estrarre il floppy dal lettore, e riavviare il computer. Al boot dovrebbe apparire presto una schermata contenente un menù simile a questo:

```
GNU GRUB  version 0.95  (638K lower / 186304K upper memory)

Debian, kernel 2.4.8.1-3-386
Debian, kernel 2.4.8.1-3-386 (recovery mode)

Use the ↑ and ↓ keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

Con l'uso dei tasti freccia (su e giù) è possibile selezionare il sistema operativo di avvio, in questo caso risulta preselezionato **Debian**. Si noti, nella figura, che per Debian-Linux esiste anche una seconda possibilità di selezione, questo accade perchè durante l'installazione viene prevista una eventuale situazione di emergenza e viene predisposta una copia ridotta del sistema operativo per simili evenienze.

Riavviata la macchina, verificato che tutto funziona regolarmente, occorrerà configurare la connessione alla rete impostando i necessari parametri.

```
ifconfig eth0 192.168.12.nn netmask 255.255.255.0 broadcast 192.168.12.255
route add default gw 192.168.12.3
echo "nameserver 151.99.125.2" >> /etc/resolv.conf
```

Da questo punto in poi si è potuta utilizzare la rete (non piu' i dischetti) per installare il sistema di base ed il resto dei programmi.

In base al tipo di connessione di cui si dispone, trascorrerà quindi un buon intervallo di tempo per completare l'installazione. Nel nostro caso l'attesa è stata di circa 45 minuti.

6 - Riavvio computer e test finale

A questo punto abbiamo veramente finito, il computer è stato spento e riavviato. Ed è iniziata la messa a punto e la personalizzazione: verifica dell'ambiente grafico, risoluzione ottimale del video, aggiunta di qualche altro pacchetto, e così via.

Informazioni complete e dettagliate sull'installazione di Debian sono contenute nel manuale di installazione scaricabile da <http://www.debian.org/releases/stable/i386/>