

Guida Per Utonti

Table of Contents

<u>Usare fetchmail.....</u>	1
<u>Introduzione.....</u>	1
<u>Configurazione di fetchmail.....</u>	1
<u>Altre opzioni di fetchmail.....</u>	3
<u>Installazione di Debian GNU/Linux 2.2 (Potato).....</u>	5
<u>Configurazione hardware utilizzata.....</u>	5
<u>Linee guida per l'installazione.....</u>	6
<u>Selezione del linguaggio per la tastiera.....</u>	6
<u>Preparazione delle partizioni per Linux.....</u>	6
<u>Indicare dove montare le partizioni.....</u>	6
<u>I moduli da far caricare al kernel di default.....</u>	7
<u>Il REBOOT.....</u>	7
<u>I servizi.....</u>	7
<u>La lista dei pacchetti.....</u>	7
<u>Semplice od avanzata ?.....</u>	7
<u>Personalizzare l'installazione.....</u>	7
<u>Le mie esperienze.....</u>	8
<u>Ringraziamenti.....</u>	9
<u>Installazione e configurazione del kernel 2.4.x.....</u>	10
<u>Tutto il procedimento in... 26 passi.....</u>	10
<u>Utilizzo dei winmodem Lucent.....</u>	16
<u>Procurarsi il necessario.....</u>	16
<u>Arrivano i problemi.....</u>	16
<u>Come risolvere i problemi.....</u>	17
<u>Nota per il kernel 2.4.....</u>	18
<u>Riferimenti.....</u>	18
<u>Utilizzo di CVS (concurrent versioning system).....</u>	19
<u>Introduzione.....</u>	19
<u>Utilizzo.....</u>	20
<u>Comandi utili.....</u>	21
<u>Breve manuale sull'uso del tool make.....</u>	23
<u>Introduzione: cosa e' make.....</u>	23
<u>Utilizzo.....</u>	24
<u>La sicurezza in MySql.....</u>	27
<u>Utenti e permessi.....</u>	27
<u>Aggiungere utenti e permessi.....</u>	28
<u>Modificare le password.....</u>	28
<u>Un piccolo approfondimento.....</u>	29
<u>Configurare il ppp per connettersi al proprio ISP.....</u>	30
<u>Il kernel e il supporto di rete.....</u>	30
<u>Informazioni sull'ISP.....</u>	31

Table of Contents

I file di connessione del ppp.....	32
Riferimenti.....	34
Come installare ReiserFS.....	35
Installazione.....	35
Note per l'uso con debian potato.....	36
Note per l'installazione del kenrnel 2.4.x.....	36
Introduzione ai comandi fondamentali di SQL.....	39
Scopo di questo documento.....	39
Convenzioni di questo documento.....	39
Righe e colonne.....	39
Quale comando per una determinata operazione ?.....	40
SELECT.....	40
INSERT.....	41
UPDATE.....	42
DELETE.....	43
Appendice A. Gli autori di GPU.....	44
Appendice C. GNU Free Documentation License (traduzione italiana).....	45
Appendice B. GNU Free Documentation License.....	52

Usare fetchmail

A cura di:

Cosimo Vagarini

<cosimo@firenze.linux.it>

Fetchmail e' un'utility per "scaricare la posta" da server di posta remoti e renderla disponibile al sistema locale di distribuzione della posta (MDA –Mail Delivery Agent–, di solito sendmail, postfix, exim, qmail ecc.)

Introduzione

Fetchmail e' in grado di collegarsi a server di posta che usano i protocolli: POP2, POP3, APOP, RPOP, KPOP, IMAP2bis, IMAP4, IMAP4–K4, IMAP–GSS, ESMTP e ETRN Una volta che fetchmail e' riuscito a ricevere un messaggio dal server di posta lo smista localmente collegandosi al server SMTP locale.

Nel caso che il MDA accetti il messaggio (disco pieno, problemi di permission, ecc.) oppure nessun MDA e' in ascolto sulla porta 25, il messaggio non risultera' consegnato e potra' essere recuperato la sessione successiva nel caso il problema sia stato, nel frattempo, risolto. Questo ci garantisce dal perdere messaggi in caso di problemi sul nostro filesystem. Nessun messaggio e' da considerarsi consegnato finche non lo e' realmente.

Fetchmail puo' ricevere parametri dalla riga di comando o utilizzare un file di configurazione (~/.fetchmailrc, se non diversamente specificato sulla riga di comando (opzione –f filename)). Nel caso che vengano passati parametri dalla riga di comando e il file ~/.fetchmailrc sia esistente, i parametri della riga di comando hanno la precedenza sul file di configurazione.

Configurazione di fetchmail

Vediamo come configurare fetchmail in modo che legga le nostre caselle di posta e ci faccia trovare nella nostra mailbox (/var/spool/mail/nomeutente) i messaggi ricevuti.

Bisogna innanzitutto creare il file .fetchmailrc nella nostra directory /home. All'interno del file inseriremo i comandi da dare a fetchmail. Ogni riga che inizia con il seno '#' viene considerata una riga di commento e, quindi non interpretata.

Supponiamo di avere un utente con logon–id rossim che si deve collegare al server di posta POP3 mail.pippo.it con utente mario e con password cielo.

```
/home/rossim/.fetchmailrc:
#
# Scarica la posta di mario@pippo.it e la passa all'utente rossim
#
poll mail.pippo.it proto POP3 user mario with pass cielo is rossim here
#
# Fine di .fetchmailrc
```

#

La sintassi e' piuttosto chiara, comunque, analizziamola:

poll mail.pippo.it

indichiamo il server di posta da contattare

proto POP3

indichiamo il protocollo da utilizzare

user mario

indichiamo a fetchmail di collegarsi al server di posta utilizzando l'account mario

with pass cielo

indichiamo la password da utilizzare per scaricare la posta

is rossim here

indichiamo che la posta di mario@pippo.it deve essere consegnata all'utente rossim sulla nostra macchina

La password all'interno di .fetchmailrc deve essere in chiaro. Si potrebbe pensare ad un problema di sicurezza ma si deve considerare che il file di configurazione deve avere permesso di lettura e scrittura per il proprietario e nessun permesso ne' per il gruppo ne' per gli altri utenti altrimenti fetchmail interrompe la sua esecuzione. In pratica:

```
$ ls -la .fetchmailrc
-rw----- 1 rossim    rossim      379 Mar 20 16:57 .fetchmailrc
```

In questo caso *SOLO* l'utente rossim (oltre a root) potra' vedere il contenuto del file di configurazione e quindi i problemi di sicurezza sono solo quelli legati al protocollo POP3.

Per far si che fetchmail scarichi la posta secondo quanto gli abbiamo indicato nel file di configurazione basta invocare fetchmail.

```
$ fetchmail
fetchmail: 3 messages for mario at mail.pippo.it (12418 octets)
reading message 1 of 3 (3456 octets) ... flushed
reading message 2 of 3 (1342 octets) . flushed
....
e cosi via.
```

Fatto questo troveremo i nostri tre messaggi in /var/spool/mail/rossim pronti per essere letti con un client di posta o per essere trattati in altro modo. Quindi ogni volta che vogliamo scaricare la posta elettronica dobbiamo invocare fetchmail. Nel caso che volessimo che fetchmail controllasse se ci sono nuovi messaggi ogni 10 minuti dovremo invocare fetchmail come demone.

```
$ fetchmail -d 600
```

fetchmail rimane quindi sempre in esecuzione e ogni 600 secondi (10 minuti) "entra in azione", rimettendosi in background una volta terminato, fino a che non saranno di nuovo trascorsi 10 minuti. Nel caso che il nostro Rossi Mario abbia piu' caselle di posta, basta configurare il nostro .fetchmailrc affinche' possa contattare tutti i server di posta relativi e prendere la posta degli accounts indicati. Esempio di .fetchmailrc per un un utente che abbia due caselle di posta presso pippo.it e una presso posta.com.

```
/home/rossim/.fetchmailrc
# Scarica la posta di mario@pippo.it e la passa all'utente rossim
poll mail.pippo.it proto POP3 user mario with pass cielo is rossim here
# Scarica la posta di m.rossi@pippo.it per rossim
poll mail.pippo.it proto POP3 user m.rossi with pass mare is rossim here
# Scarica la posta di mario@posta.com per rossim con protocollo IMAP4
poll mail.posta.com proto IMAP user mario with pass sole is rossim here
#
# Fine di .fetchmailrc
#
```

Una volta eseguito fetchmail (esattamente come nell'esempio precedente) nella mailbox di rossim troveremo tutti i messaggi delle sue tre caselle postali. Possiamo anche, ovviamente, aver bisogno di scaricare la posta per diversi utenti contemporaneamente in modo che poi, i singoli utenti, attingano direttamente alle loro mailboxes presenti sulla nostra macchina.

```
.fetchmailrc
# Scarica la posta di mario@pippo.it e la passa all'utente rossim
poll mail.pippo.it proto POP3 user mario with pass cielo is rossim here
# Scarica la posta di gianni@pippo.it per bianchig
poll mail.pippo.it proto POP3 user gianni with pass ferie is bianchig here
# Scarica la posta di anna@pippo.it per verdia
poll mail.pippo.it proto POP3 user anna with pass vita is verdia here
#
# Fine di .fetchmailrc
#
```

E' evidente che l'utente che esegue fetchmail anche per altri utenti deve avere i diritti di scrittura nella directory /var/spool/mail.

Altre opzioni di fetchmail

fetchmail prevede la possibilita' di ricevere come parametri (oppure come comandi all'interno di .fetchmailrc) numerose opzioni. Non staro' qui ad elencarle tutte (verificare le man page di fetchmail(1)) ma solo a descrivere quello di uso piu' comune.

-f filename

con questa opzione si indica di utilizzare un file di configurazione alternativo a .fetchmailrc. E' ovvio che questa opzione e' valida solo da riga di comando.

-S host

con questa opzione si indica a fetchmail il nome dell'host a cui consegnare la posta (potremmo per esempio avere piu' di un server sulla nostra rete, ma un solo processo che recupera tutti i messaggi di posta per gli utenti di tutta le rete).

-e [N]

questa opzione e' valida solo se la posta risiede su un server IMAP. Ed indica se cancellare immediatamente la posta man mano che viene ricevuta. E' utile per evitare, nel caso di caduta della linea durante la ricezione della posta, il recupero degli stessi messaggi piu' volte. D'altro canto, nel caso di caselle di posta molto grosse, potrebbe portare a dei rallentamenti eccessivi. Se indicato il parametro N, l'expunge sara' eseguito ogni N messaggi.

-m mda

serve ad indicare l'MDA al quale fetchmail passera' la posta.

[Indietro](#)

Introduzione ai comandi
fondamentali di SQL

[Partenza](#)

[Risali](#)

[Avanti](#)

La sicurezza in MySQL

Guida Per Utonti

[Indietro](#)

Capitolo 2. Documenti non ancora classificati

[Avanti](#)

Installazione di Debian GNU/Linux 2.2 (Potato)

A cura di:

Umberto Genovese

<umbegen@tin.it>

Vorrei mettere a disposizione di tutti le mie esperienze (oramai diverse) sulla installazione della Debian Potato.

Configurazione hardware utilizzata

La Debian Potato (d'ora in poi semplicemente DbP) la uso su due macchine diverse come potenza di calcolo e architettura:

Personal Computer "1":

Pentium 133 Mhz
RAM 48 M
Scheda Video PCI Cirrus Logic 5422 1M ram
Pioneer CD-ROM ATAPI Model DR-A04S 0105
HD Fujitsu MPC3032AT IDE :

Filesystem	Size	Name
/dev/hda3	46M	/
/dev/hda2	11M	/boot
/dev/hda5	84M	/var
/dev/hda6	1.1G	/usr
/dev/hda7	697M	/home
/dev/hda1	1.0G	/Windows 95
swap	48M	

Personal Computer "":

PentiumII 266 Mhz
RAM 64 M
Scheda Video AGP Matrox G400 32M ram
HD QUANTUM FIREBALL SE4.3A IDE:

Filesystem	Size
/dev/hda1	2.5G /fat32 (os: Windows 98)
/dev/hda2	1.5G /fat32 (pronta per BeOS)

Controller ADAPTEC 2940 UltraWide SCSI
PIONEER Model: DVD-ROM DVD-303
Type: CD-ROM ANSI SCSI revision: 02
QUANTUM Model: FIREBALL SE6.4S Rev: PJ09
Type: Direct-Access ANSI SCSI revision: 02

Filesystem	Size	Name
/dev/sda2	1.4G	/
/dev/sda1	7.6M	/boot
/dev/sda5	463M	/var
/dev/sda6	1.8G	/usr
/dev/sda7	2.2G	/home

swap 64M

Linee guida per l'installazione

La Debian è nota per non essere molto user-friendly, specialmente per l'installazione: ho sentito di leggendo metropolitane che parlano di giorni di lavoro (forse lo erano per le prime versioni, questo non lo so), ma chi ha installato solo RedHat e/o Mandrake può avere difficoltà all'inizio. L'installazione di DbP può essere fatta da CD, da floppy e da DOS (anche da Windows, ma si deve settare l'esecuzione della finestra DOS in modalità esclusiva) Se si ha una macchina che prevede il boot da CD non ci sono problemi: basta avviare la macchina col primo CD e l'installazione parte da sé. Altrimenti si può fare il dischetto di boot con l'utility 'rawrite' che chiede il percorso dell'immagine da copiare (CDROM:/install/boot.img) e il drive del floppy (gen. 'a:'). Se non si vuole o non si può fare il floppino si può sempre entrare nell'unità CD da MS-DOS (ex. 'd:') nella directory install (cd install) e lanciare 'boot.bat' (una nota per chi è alle prime armi: per usare il sistema MS-DOS ci sono due modi: all'avvio di Windows premere il tasto F8 per scegliere la modalità MS-DOS (non la modalità provvisoria: non si potrebbe usare il CDROM) o chiudere Windows e scegliere 'riavvia in modalità MS-DOS'). Qualsiasi metodo sia scelto, a questo punto l'installazione è iniziata: Linux viene caricato interamente in RAM per cui non ci sono problemi se qualcosa in questa fase va storto.

Questi sono i passaggi richiesti durante l'installazione semigrafica, ma efficiente:

Selezione del linguaggio per la tastiera

Per l'Italia deve essere selezionata la "qwerty-it"

Preparazione delle partizioni per Linux

in questa fase è bene essere sicuri: da qui dipende il sistema che si va a installare; tralascio la fase di preparazione dello spazio su disco (ognuno può usare 'PartitionMagic' che non è free o 'fips' che è un utility per MS-DOS ed è free) per raccomandare che le partizioni (si può fare anche con una sola partizione per il S.O. più una di swap, ma non è conveniente secondo me) non siano sottodimensionate: Debian è permalosa e non continua se non ha spazio sufficiente. Una nota sulla partizione di swap: io uso una swap pari alle dimensioni della RAM disponibile sul PC, più grande serve solo se si ha poca RAM a disposizione, altrimenti è uno spreco dello spazio sul disco. Comunque ho riassunto le dimensioni delle partizioni delle mie macchine da cui si può prendere spunto. I dischi in generale non accettano più di 4 partizioni primarie per cui si farà ricorso alle partizioni logiche per un numero superiore di esse, l'importante è ricordarsi lo scopo per cui sono state create. L'utility usata per le partizioni durante l'installazione è 'cfdisk', che è molto più potente dell'analogo per DOS 'fdisk' e ha un ottimo help per i comandi per i vari passaggi. Dimensionate le partizioni, bisogna assegnare loro il filesystem da costruire e indicarlo con il comando 't': 83 è l'ID per le partizioni del S.O. Linux (ext2) e 82 è l'ID per la partizione separata di swap (linux swap). Salvare tutto su disco con 'w' e si può continuare.

Indicare dove montare le partizioni

Adesso si devono indicare le partizioni create all'installazione affinché vengano montate in modo coerente al loro scopo:

NOME	SIGNIFICATO
/	principale (root) da cui partono tutti le partizioni montate e le directory di sistema
/boot	le immagini del kernel da caricare all'avvio
/var	i servizi e la cache

```
/usr      i programmi
/home     gli utenti, i loro dati ecc.
/swap     generalmente invisibile, ma c'è
```

A questo punto l'installazione copia il kernel in /boot e crea un sistema minimo partendo dall'immagine del CD, ma è solo il minimo indispensabile!

I moduli da far caricare al kernel di default

non si può saltare questa fase, perché necessaria al proseguimento, ma si può non selezionare nulla tranquillamente, perché i moduli necessari al corretto funzionamento il kernel se li prende da sé secondo le esigenze della macchina e poi basta lanciare 'modconf' come root se ce ne fosse bisogno in seguito.

II REBOOT

viene chiesto se installare il LILO (LIinux LOader), in quale disco e in quale partizione e se si vuol fare il floppy per il boot dall'immagine del kernel che è sul disco rigido. Poi viene chiesto il riavvio della macchina per il proseguimento.

I servizi

Dopo il reboot vengono chiesti i servizi per le password (shadow password si/no), l'immissione di password di root e l'attivazione degli utenti (personalmente gli utenti li configuro sempre dopo l'installazione). Ora viene la parte più manuale e secondo me più interessante: la creazione del database delle applicazioni da installare.

La lista dei pacchetti

Viene richiesto di inserire i CD dell'installazione, ma si possono aggiungere anche altri CD che contengono dei file DEB, perché se ne vuole l'installazione all'inizio o comunque se ne prevede il loro utilizzo in futuro (così ho fatto per Helix Gnome). Quando si ritiene che il database è completo ne verrà mostrato il contenuto in formato testo con l'editor 'ae', basterà fare ^Q (control-Q) per uscirne e continuare con l'installazione.

Semplice od avanzata ?

La "semplice" prevede i servizi base e la scelta di interi gruppi di servizi (gnome, x11, linguaggi vari ecc.), la "complessa" di scegliere i pacchetti uno per uno (bisogna sapere quali e a cosa servono) Praticamente è tutto finito: adesso verranno spaccettati i pacchetti e configurati per l'uso, in qualche caso verranno poste delle domande su cosa si voglia e come configurare vari pacchetti (sendmail, gmc, ecc.) per cui è bene conoscere un attimino come è strutturata la macchina che si vuol configurare (porte seriali, il nome dell'host, dove è attaccato il mouse ecc.).

Personalizzare l'installazione

Quando anche questo passaggio è finito, la Debian Potato è pronta: ma per un perfetto lavoro rimangono solo alcune cose che andranno fatte a mano: ma niente di cui aver paura! La prima cosa che appare è la richiesta di login: è necessario entrare come root per creare gli utenti e i loro profili con 'adduser' e finire con i settaggi:

Il server X: io uso per la sua corretta configurazione l'utility 'XF86Setup' (si scrive così!) per settare mouse, tastiera, scheda video, monitor e risoluzione dello schermo (ma quando farà anche il caffè?), la quale per funzionare ha bisogno del Xserver per VGA, per cui se non parte probabilmente sarà necessario installarlo con: 'apt-get install xserver-vga16'. C'è poi da creare il file '.xsession' nelle directory principali degli utenti che specifica le caratteristiche del window-manager da caricare all'avvio di X (ex. 'gnome-session' per Gnome) con qualsiasi editor di testo (joe, emacs e vi sono i più noti) e renderlo eseguibile con 'chmod +x .xsession'.

L'ambiente di sistema: anche se si è scelto di avere la tastiera italiana, il S.O. non può sapere a priori che l'utente è italiano: per cui le lettere accentate in console, alcuni messaggi di sistema o funzioni come 'man', i formati valuta-data-ora non saranno visualizzate correttamente per cui col solito editor di testo sarà necessario editare il file '/etc/environment' e specificare la lingua (LANG=it_IT) perché tutto sia a posto.

Altri servizi come la ricezione-spedizione della posta o la preparazione della connessione a Internet non rientrano negli scopi di questo documento, ma con la Debian funzionante non è difficile metterli a posto: la man-pages di Linux sono abbastanza esaustive e rendono il compito alla portata di tutti!

Le mie esperienze

La macchina (1) è veramente un accrocchio di tecnologia più o meno datata, addirittura i chip della RAM non sono nemmeno tutti uguali, ma funziona benissimo da oltre un anno circa 13 ore al giorno senza il minimo problema (ho cambiato diverse volte distribuzioni di Linux senza mai un problema e rischiare di perdere i miei dati in Windows). Dopo un paio di mesi dalla 1a installazione di DbP ho avuto il noioso problema di non avviare il server X: la /var (che era insieme alla partizione /) aveva riempito la partizione impedendomi di fare tutto: nemmeno man qualche cosa! A quel punto ho deciso di rifare tutto da capo e... 2a installazione: tempo fermo macchina una giornata (mentre si fa tutt'altro lavoro, installare Debian non è cosa veloce). Ho scelto così come da specchio precedente le mie partizioni: ho cercato di separare in partizioni distinte i servizi essenziali del sistema, dando molto spazio alla /usr per i programmi che via via installo e/o scarico dalla rete. Con la /var separata spero di avere molti problemi in meno in futuro, casomai si riempisse di nuovo (chissà perché lo fece? ;-)

La macchina (2) che ho a casa è obiettivamente tutt'altra cosa: l'aver due hard disk diversi, anche come tecnologia (IDE e SCSI) mi ha un attimino complicato le cose, complice il fatto di avere un lettore CD-ROM SCSI che non ne vuol sapere di fare il boot (anche la macchina (1) non lo fa, ma il bios è vecchiotto e non lo prevede), per cui ho usato il loadlin presente nella directory install del 1o cdrom, che contiene le immagini da caricare in ram per l'avvio della procedura dell'installazione. Il problema sorge in due momenti distinti e ho dovuto lavorare con 'ae' per sistemare le cose. Procediamo con ordine: l'operazione iniziale dell'installazione prevede di settare la tastiera (ovviamente qwerty-it per l'Italia) e il partizionamento dell'HD. Successivamente l'inst. va a cercare di accedere al CD-ROM in /dev/sr0 (perché è SCSI) ma non lo trova e non lo monta, quindi non si va avanti. Ho risolto l'inconveniente andando a vedere cosa c'è in /dev/ con la shell (la Debian ha pensato anche a questo: tra le opzioni permette di accedere al sistema in creazione con una shell) e il dispositivo sr0 non esiste: ho fatto un link simbolico a /dev/scd0 e tutto ha ripreso a funzionare (ln -s /dev/scd0 /dev/sr0). Il secondo problema si è presentato subito prima del reboot: il sistema avendo riconosciuto solo la parte SCSI della macchina e non avendo montato l'HD IDE, cerca di installare LILO sul master boot record dell'HD SCSI o sulla partizione di boot (per me /dev/sda1). Ancora una volta ho aperto la shell e sono andato direttamente sulla directory /etc del disco attraverso il mount point /target e ho modificato il lilo.conf nella sezione boot (boot=/dev/hda invece di boot=/dev/sda) e indicato anche quale S.O. deve essere di default ('man lilo' per i dettagli). Una volta che LILO è stato installato, non rimane che il reboot del sistema per finire ma... i pacchetti che dovranno essere selezionati non saranno trovati sul cdrom se prima non viene modificato un link simbolico in /dev (la directory che è scritta sull'HD), quindi sempre da shell, prima del reboot, mount point /target (che fa riferimento appunto all'HD), ho modificato il link simbolico /dev/cdrom

che puntava sempre a /dev/sr0 reindirizzandolo al dispositivo giusto /dev/scd0:

```
# rm /dev/cdrom
# ln -s /dev/scd0 /dev/cdrom
```

Reboot, selezione pacchetti e voilà: la Debian Potato è installata e funzionante!

Ringraziamenti

Al Mio Maestro Franco Vite (non è così utonto come dice) che mi ha indirizzato all'open source

Alla lista Tech del Firenze.Linux.it per i continui suggerimenti che lì si trovano

A tutte le altre comunità Linux per il continuo lavoro di ricerca e divulgazione

[Indietro](#)

Configurare il ppp per connettersi al proprio ISP

[Partenza](#)

[Risali](#)

[Avanti](#)

Come installare ReiserFS.

Guida Per Utonti

[Indietro](#)

Capitolo 2. Documenti non ancora classificati

[Avanti](#)

Installazione e configurazione del kernel 2.4.x

A cura di:

Sergio Ballestrero e theOld

<>

Queste istruzioni si riferiscono alla RedHat 7.0, ma, escluse alcune particolarità (riferimenti ai pacchetti RPM, uso dell'apposito compilatore `gcc`) sono applicabili a qualsiasi distribuzione basata su kernel 2.2.x

Installare un nuovo kernel non è un'operazione estremamente complessa, ma l'iter da seguire è piuttosto rigoroso e bisogna sempre assicurarsi che non si siano verificati errori fra uno step e l'altro.

Innanzitutto è bene tenere presente che, se esiste una versione precompilata del nuovo kernel in formato rpm, è consigliabile utilizzare questa, piuttosto che imbarcarsi nel procedimento più insidioso descritto di seguito: io non ho potuto utilizzare il file rpm, poichè dovevo sperimentare alcune patch, e quindi dovevo ricompilare il kernel dai sorgenti; inoltre queste particolari patch richiedevano una versione non disponibile come `kernel-source*.rpm`.

Tutto il procedimento in... 26 passi

Il procedimento elenca tutti gli step seguiti per aggiornare il kernel del mio Linux Red Hat 7.0 dalla versione 2.2.16-22 (quella installata dal cd) alla 2.4.2.

Passo 0 . Le modifiche al sistema sono consentite solo all'utente root, quindi è necessario effettuare il login come root; ed è preferibile poi attivare l'ambiente grafico, sia per la facilità d'uso del filemanager e delle sue funzionalità (copia e incolla, etc.), sia per poter utilizzare il comando `make xconfig` (descritto in seguito), che considero un po' più comodo rispetto al `make menuconfig`, eseguibile anche da console.

Passo 1 . Per prima cosa, bisogna procurarsi i sorgenti. Su Internet si può andare sul sito <ftp://ftp.kernel.org/pub/linux/kernel/v2.4> e prelevare il file "**linux-2.4.x.tar.bz2**", (dove la 'x' sta per la versione corrente o ultima del kernel. Al momento è la 2.4.3). È consigliabile il file di tipo .bz2 perchè ha una compressione maggiore e, quindi, si scarica più velocemente.

Nota: [assicuratevi di avere il programma `bzip2`]

Passo 2 . Spostate tale file nella directory `/usr/src`. Aprite la shell (ricordo che sono l'utente root) e posizionatevi nella stessa directory. In questa dir è presente una directory simbolica (un link simbolico ad una directory, per essere più precisi), chiamata "linux", che punta all'attuale versione del kernel, ovvero alla dir (nel mio caso) `/usr/src/linux-2.2.16`. Per evitare che la nuova versione che si sta per installare possa mescolarsi con l'attuale, è necessario rinominare tale dir in "linux.OLD" (è sempre meglio conservare con un altro nome, piuttosto che cancellare). Tale operazione è semplice e si compone di un solo comando: **`mv linux linux.OLD`**

Passo 3. Nella shell, eseguite il comando `tar Ixvf linux-2.4.2.tar.bz2` che provvede a scompattare il file generando la directory `/usr/src/linux`.

Passo 4. Rinominate tale directory in `/usr/src/linux-2.4.2` e poi generare un file di collegamento simbolico "linux" che punta alla predetta dir. E' agevole fare ciò con le funzionalità del filemanager. Volendo, comunque, utilizzare la console anzichè il filemanager si ottiene lo stesso risultato eseguendo i seguenti comandi:

```
mv linux linux-2.4.2    (per rinominare la dir)
ln -sf linux-2.4.2 linux    (per creare il collegamento simbolico)
```

Passo 5. Nella shell, posizionatevi nella dir `/usr/src/linux` (oppure `/usr/src/linux-2.4.2` che è la stessa cosa)

Passo 6. E' opportuno, anche se non indispensabile in questa fase, creare dei collegamenti simbolici nella directory `/usr/include`, per far sì che eventuali compilazioni di altri programmi o pacchetti puntino alle librerie del nuovo kernel. Poichè è buona norma non cancellare files o directory preesistenti, rinominate le dir "asm", "linux" e "scsi" presenti nella `/usr/include` apponendo a tutte e tre il suffisso ".OLD": Dopo si possono creare i seguenti collegamenti simbolici:

```
ln -sf /usr/src/linux/include/asm-i386 /usr/include/asm
ln -sf /usr/src/linux/include/linux /usr/include/linux
ln -sf /usr/src/linux/include/scsi /usr/include/scsi
```

Tenete presente, comunque, che futuri pacchetti da installare potrebbero richiedere le vecchie include. Qualora si verificano errori in fase di installazione, ripristinate le tre directory allo stato originario.

Passo 7. Nella Red Hat 7.0 il compilatore gcc è una versione 'sviluppo', non adatta per compilare il kernel. Al suo posto si deve usare il `kgcc`. Per fare in modo, quindi, che il sistema utilizzi il `kgcc`, occorre modificare il file "Makefile" presente nella directory corrente, ovvero `/usr/src/linux`. Attivate un programma di edit per modificare il Makefile. Più o meno al rigo 29 è presente la seguente l'istruzione:

```
CC = $(CROSS_COMPILE) gcc
```

sostituite `gcc` con `kgcc` e salvate il file.

Passo 8. Se fosse necessario installare una patch, bisogna farlo adesso.

Nota: [se e' la prima volta che si compila un kernel, e la patch non e' indispensabile all'avvio del sistema, puo' essere consigliabile rimandare l'applicazione delle patch a dopo aver compilato e testato il kernel non patchato]

Scaricata la patch da Internet (per esempio "nuova-patch.gz") in una directory apposita (ad es. `/usr/src/download`), la si può applicare con il seguente comando:

```
patch -p1 < /usr/src/download/nuova-patch.gz
```

E' opportuno che la patch da installare sia relativa alla esatta versione del kernel: è infatti sconsigliabile applicare una patch creata per un kernel diverso da quello che si sta installando – si rischiano conflitti nell'applicazione della patch, e anche se questa andasse a buon fine, il kernel risultante potrebbe non essere completamente affidabile.

Nota: [Se si devono applicare piu' patch, e' bene cercare informazioni sulla compatibilita' di queste patch (ed eventualmente l'ordine corretto in cui applicarle), e prestare molta attenzione che non si verifichino errori nell'esecuzione del comando patch.]

Passo 9 . A questo punto può iniziare la fase di configurazione e compilazione del kernel. Se si desidera partire da una situazione "pulita", senza dipendenze che potrebbero essere errate, si può eseguire il comando

```
make mrproper
```

Attenzione, però, che questo comando cancella i files generati in un precedente tentativo di configurazione, compreso il file .config che contiene le scelte operate nel passo successivo. Eseguirlo, comunque, non è dannoso. Si può evitarlo, per non perdere tempo, mentre si fanno delle prove e poi, determinata la configurazione ideale, ripartire da questo passo, eseguire questo comando (dopo aver salvato il .config per poi ripristinarlo) e poi proseguire nella configurazione.

Nota: [e' consigliabile eseguire make mrproper se il kernel e' gia stato compilato e si e' installata una nuova patch]

Nota: [sembra essere indispensabile eseguire make mrproper se il kernel e' gia stato compilato e si vuole modificare "*Set version information on all module symbols*" o "*Symmetric multi-processing support*"]

Passo 10 . Eseguite il comando

```
make xconfig
```

Questo comando serve per impostare il .config di cui si è già parlato. Dopo qualche messaggio nella shell, si apre una finestra con una miriade di opzioni. Ad alcune è facile rispondere (se non ho dischi SCSI posso sicuramente disabilitare la relativa opzione) per altre, invece, la scelta fra attivazione, attivazione con modulo o disattivazione non è affatto intuitiva.

Inoltre, le scelte proposte di default non sempre si rivelano le più felici [specie per un utilizzo come workstation], e gli help non sono esaustivi [e anche la Documentation e' spesso eccessivamente tecnica; solitamente una ricerca in rete, quando se ne ha il tempo, fornisce documentazione piu' leggibile]. E' bene, quindi, avere un esempio di una precedente configurazione e cercare di impostare le stesse opzioni, salvo le modifiche che si ritengono opportune.

Nella Red Hat 7.0 il .config relativo alla configurazione base è stato copiato nel file "/usr/src/linux-2.2.16/configs/kernel-2.2.16-i686.config". Lo si può aprire con un editor o, meglio ancora, avviare una seconda shell, posizionarsi nella dir che contiene l'attuale configurazione ed eseguire un altro make xconfig. Con l'apposito pulsante si può richiamare il suddetto kernel-2.2.16-i686.config e così si avranno due finestre di configurazione: la prima, per effettuare le dovute impostazioni, la seconda solo per leggere le impostazioni precedenti.

Qualora si trovi un'opzione non presente nella precedente versione, se non se ne comprende bene l'uso e se possibile, è preferibile sceglierne l'attivazione come modulo: così facendo, il codice corrispondente verrà caricato (automaticamente) solo se il proprio sistema necessita di quella particolare funzionalità.

Alla fine delle impostazioni, è necessario effettuare il salvataggio solo dalla prima finestra. Questo passo è molto delicato. Facendo una scelta non adeguata al proprio sistema, è facile che in uno degli step successivi si verifichi un errore. Nel caso, si deve tornare a questo step e modificare l'opzione sbagliata. In ogni caso, dato

che in ogni esecuzione viene generato il file `.config` nella stessa directory, è bene salvarlo anche con un altro nome per confronti successivi (es. `bakupXX.config` dove `XX` è un progressivo) e per avere tutte le configurazioni testate.

Passo 11 . Eseguite il comando

```
make -s dep
```

Passo 12 . A questo punto si potrebbe eseguire il comando

```
make -s clean
```

Non è indispensabile e, in fase di prove e di fretta, se ne può fare a meno. Valgono più o meno le stesse considerazioni del precedente `make mrproper`.

Nota: [e' necessario eseguire `make clean` se il kernel e' gia stato compilato e si e' installata una nuova patch]

Passo 13 . Eseguite

```
make -s bzImage
```

Questo comando genera il kernel vero e proprio.

Passo 14 . Eseguite

```
make -s modules
```

Questo comando genera i moduli scelti nel precedente passo 10.

Passo 15 . Eseguite

```
make -s modules_install
```

Questo comando copia nella directory `/lib/modules/2.4.2` i moduli generati.

Passo 16 . Eseguendo gli step dall'11 al 15, sul video possono apparire alcuni messaggi. Assicuratevi che nessuna delle ultime righe visualizzate segnali un errore: nel qual caso bisognerebbe modificare la configurazione (precedente punto 10).

Nota: [nota: questo e' principalmente dovuto al fatto che i kernel 2.4 sono relativamente recenti, e il meccanismo delle dependency non e' ancora stabilizzato come nei 2.2. E l'applicazione di patch complica le cose]

Passo 17 . Se tutto è andato bene, sarà stato generato il file `bzImage` nella dir `/usr/src/linux-2.4.2/arch/i386/boot`. Copiatelo nella directory `/boot`, rinominandolo con un nome che andrà poi utilizzato nel `lilo.conf` (vedere oltre). Per convenzione, e analogia alla versione precedente, è preferibile chiamarlo `vmlinuz-2.4.2` (con la `z`, non con la `x`).

Passo 18 . Copiate anche i files `System.map` e `vmlinux` (che sono stati generati nella `/usr/src/linux-2.4.2`) nella `/boot` rinominandoli in `System.map-2.4.2` e `vmlinux-2.4.2`. Invero `vmlinux` non sarebbe necessario, ma se si ha spazio a sufficienza è buona norma averne una copia in `/boot`.

Passo 19 . Poichè può accadere che occorran molte prove prima di trovare una giusta configurazione, è opportuno creare un file eseguibile, magari anche all'interno della /boot, assegnandogli un nome "parlante", tipo "installa-kernel-2.4.2", che abbia il compito di eseguire i passi ripetitivi riportati nei due precedenti step. Il contenuto di tale file potrebbe essere il seguente:

```
#!/bin/sh
VERSIONE=2.4.2
rm /boot/System.map
rm /boot/vmlinuz-$VERSIONE
rm /boot/vmlinux-$VERSIONE
cp /usr/src/linux-$VERSIONE/arch/i386/boot/bzImage /boot/vmlinuz-$VERSIONE
cp /usr/src/linux-$VERSIONE/System.map /boot/System.map-$VERSIONE
cp /usr/src/linux-$VERSIONE/vmlinux /boot/vmlinux-$VERSIONE
```

In questo modo, anzichè eseguire singolarmente le copie indicate nei passi 17 e 18, si può eseguire il file predetto, che provvede ad effettuare le copie necessarie.

Passo 20 . Nella /boot si trovano anche due "file" symlink (collegamenti simbolici) vmlinuz e System.map. Dato che nel lilo.conf (passo successivo) si fara' riferimento solo al vmlinuz-2.4.2 e, di conseguenza, al System.map-2.4.2, possiamo ignorare o anche cancellare questi files (il System.map verrà rigenerato automaticamente alla ripartenza e fatto puntare al System.map-XX.XX.XX opportuno, relativo al vmlinuz-XX.XX.XX attivato). Sempre nella /boot, si trovano anche due files: kernel.h-2.2.16 e module-info-2.2.16-22. Non vengono creati files corrispondenti della release che si sta installando (la 2.4.2), perciò potete lasciare questi files senza preoccuparvi del loro significato.

Passo 21 . A questo punto si deve modificare il filo /etc/lilo.conf per abilitare il sistema all'avvio sia col vecchio kernel 2.2.16-22 che col nuovo 2.4.2. Lasciando invariate le prime istruzioni, la parte finale potrebbe essere così:

```
default=linux-2.4.2
image=/boot/vmlinuz-2.2.16-22
    label=linux-2.2.16-22
    read-only
    root=/dev/hda8
image=/boot/vmlinuz-2.4.2
    label=linux-2.4.2
    read-only
    root=/dev/hda8
```

il rigo default=xxxx istruisce il lilo su quale kernel attivare automaticamente, dopo un certo numero di secondi di attesa, durante i quali è possibile scegliere la versione del kernel attivare.

Passo 22 . Fatto questo, nella solita shell immettete il comando

```
lilo -v
```

che, sulla scorta del lilo.conf, genererà il settore di boot per l'avvio del computer.

Passo 23 . Il nuovo kernel 2.4.2 richiede anche l'aggiornamento di alcuni pacchetti (consultate il /usr/src/linux-2.4.2/Documentation/Changes). Per il passaggio dalla 2.2.16-22 alla 2.4.2 è obbligatorio adeguare i pacchetti util-linux, modutils, ppp e isdn4k-utils. Non è questa la sede per trattare anche dell'aggiornamento dei predetti pacchetti.

Passo 24 . Fate ripartire il sistema e verificate che tutto funzioni. Se non funziona, rifate il boot con il vecchio kernel e ripartite dal punto 10.

Passo 25 . Se tutto e' andato liscio, complimenti, e benvenuti nel fantastico mondo del Linux 2.4.

Passo 26 . 26) Se avete compilato il kernel senza una o piu' delle patch che vi servivano, ripartite dal punto 8, senza ripetere l'applicazione di patch eventualmente gia' applicate.

[Indietro](#)

La sicurezza in MySql

[Partenza](#)

[Risali](#)

Guida Per Utonti

Capitolo 2. Documenti non ancora classificati

[Avanti](#)

Gli autori di GPU

[Indietro](#)

[Avanti](#)

Utilizzo dei winmodem Lucent

A cura di:

Claudio Cicali

<claudioc@firenze.linux.it>

Se avete comprato un portatile delle ultime generazioni è molto probabile che all'interno si trovi un winmodem (o soft-modem come dir si voglia). Avete sentito ovunque dire che i winmodem sotto linux non funzionano ma, beh, questo fortunatamente è solo *quasi sempre* vero. Se il modem all'interno del vostro portatile è prodotto dalla Lucent (www.lucent.com) ci sono buone probabilità di farlo funzionare.

Procurarsi il necessario

La prima cosa da fare è scaricarsi il modulo binario che la stessa lucent mette a disposizione <http://linmodems.org/linux568.zip>. Purtroppo, essendo solo in forma binaria e non sorgente, il modulo non può essere ricompilato e questo, come vedremo, pone dei discreti problemi.

Il primo problema è che questo modulo è compilato con la versione 2.2.12 del kernel di Linux e tentare di caricarlo con versioni successive del kernel impone l'utilizzo del flag `-f` (force) durante l'inserimento (tramite *insmod* o *modprobe*).

Con un normale **insmod** *nome del modulo* otterremo degli errori che indicano appunto che tale modulo è compilato per una versione più vecchia del kernel. Usando il flag `-f` imporremo comunque al kernel di caricare il modulo, che però continuerà ad avvertirci del problema.

Siccome l'informazione che serve per individuare con quale versione del kernel è stato compilato il modulo è all'interno del modulo stesso, si può operare un'operazione manuale sul file binario per ingannare il kernel sulla versione del modulo. Questa operazione, si badi bene, ha solo uno scopo "estetico" e serve solo per evitare i messaggi di avvertimento a fronte dell'uso del `-f`. Esiste uno script fatto apposta per questa operazione, e lo trovate qui: <http://www.test.dcnut.com/linmodem/fixscript>

A questo punto avrete il modulo inserito in memoria e pronto all'uso. Se provate ad utilizzare uno strumento per "colloquiare" con il vostro modem, come potrebbe essere minicom, vedrete che (eureka!) il vostro modem si lascia interrogare con i normali comandi AT e fa addirittura un numero telefonico !

Ma i problemi iniziano quando cercherete di utilizzare il vostro modem insieme al pppd per accedere ad Internet.

Arrivano i problemi

Se il vostro kernel è ancora un 2.2.12 o 13 o 14, non dovrete avere problemi, ma con kernel superiori, noterete che appena tenterete di avviare il pppd (a mano o con programmi che lo fanno per voi come il kppp o il gnome-ppp) questo morirà inaspettatamente (questa è la traduzione del messaggio che normalmente viene dato dai suddetti programmi). Andando a curiosare nei log di sistema (`/var/log/messages`) noterete queste

scritte (insieme ad altre...)

```
localhost kernel: Unable to handle kernel paging request at virtual address 60000000
localhost kernel: current->tss.cr3 = 0392f000, %cr3 = 0392f000
localhost kernel: *pde = 00000000
localhost kernel: Oops: 0002
localhost kernel: CPU: 0
localhost kernel: EIP: 0010:[<c80a27a1>]
localhost kernel: EFLAGS: 00010002
localhost kernel: eax: 00000fff ebx: 0000002d ecx: 0000000b edx: 00001000
localhost kernel: esi: c7872638 edi: 60000000 ebp: c7872400 esp: c3931ecc
localhost kernel: ds: 0018 es: 0018 ss: 0018
localhost kernel: Process pppd (pid: 722, process nr: 48, stackpage=c3931000)
localhost kernel: Stack: 00000001 c7872400 00000286 00000000 c8057e8b c3938000 00000000 c7872638
localhost kernel: 0000002d c7872400 c55c7c20 c7949ce0 c8057da8 c7872400 0000c021 c805a017
localhost kernel: c7872400 c7949ce0 c7872420 c7872400 c7872420 c55c7c20 00000018 c805a30b
localhost kernel: Call Trace: [ppp:__insmod_ppp_S.text_L15801+3639/15428]
```

Argh ! disastro ! Che e' successo ? Semplicemente il vostro pppd è "crashato" orribilmente. Morto. Defunto.

Il problema è che dal kernel 2.2.14 al 2.2.15 e' stato modificato il codice del modulo ppp, e questa modifica fa andare in crash il pppd se usato insieme al modulo della Lucent.

E adesso ?

Come risolvere i problemi

Beh, le soluzioni sono almeno due. Fortunatamente (viva il software libero !) abbiamo a disposizione il codice sorgente del ppp (/usr/src/linux/drivers/net/ppp.c) e questo ci permette di poterlo modificare in maniera da togliere quel (piccolo) aggiornamento che dà tanto fastidio al nostro modulo. Per essere esatti il file da modificare è /usr/src/linux/include/linux/tty.h. Se conoscete un po' il C la modifica da fare è questa: all'interno della struttura tty_struct, spostate la dichiarazione "**struct wait_queue *poll_wait;**" in fondo alla struttura stessa, subito dopo "**spinlock_t read_lock;**". Se non conoscete il C, entrate nel file incriminato e cercate "**tty_struct**". Adesso scendete fino a trovare la linea dove sta scritto: "**struct wait_queue *poll_wait;**". Prendete questa riga, togliendola dalla sua posizione originale, e spostatela subito prima della prossima parentesi graffa che troverete, ovvero subito dopo la riga dove sta scritto "**spinlock_t read_lock;**".

Salvate il file e ricompilate il kernel. Abbiate cura, come al solito, di usare il pppd in forma di modulo, non staticamente linkato all'interno del kernel.

A questo punto tutto dovrebbe funzionare.

Ma se siete pigri o non vi azzardate a fare queste modifiche, esiste un'altra soluzione, anzi due. La prima soluzione è quella di procurarvi un modulo ppp compilato con un kernel precedente al vostro (che naturalmente vi darà gli stessi problemi del modulo della Lucent. Usate il flag -f all'inserimento per risolverli, come detto all'inizio).

Infine, la seconda soluzione è trovare il modulo ppp.o per il vostro kernel compilato da qualcuno che ha fatto il lavoro sporco per voi (la modifica al tty.h). Ecco dove li potete trovare (sono tre diversi, uno per la versione 2.2.15, uno per la 16 e per la 17; si distinguono facilmente dal loro nome):

- [ppp.o.2.2.15](#)
- [ppp.o.2.2.16](#)
- [ppp.o.2.2.17](#)

Per installarli fate così: prima di tutto copiate il vostro modulo ppp.o attuale da qualche parte

```
mv /lib/modules/2.2.XX/net/ppp.o /lib/modules/2.2.XX/net/ppp.o.orig
```

e poi copiate al suo posto uno di quelli scaricati:

```
mv ppp.o.2.2.XX /lib/modules/2.2.XX/net/ppp.o
```

Nota per il kernel 2.4

Infine, un'ultima nota: non c'è nessun modo di far funzionare il modulo della lucent con il kernel della serie 2.4

Riferimenti

La maggior parte delle soluzioni mostrate su questo documento sono state prelevate dalle seguenti fonti:

HOW-TO sui winmodem: [linmodem-howto](#)

Altre informazioni sui winmodem: <http://www.heby.de/computer/ltmodem>

Sito ufficiale per lo sviluppo del supporto Linux ai winmodem: [linmodem.org](#)

[Indietro](#)

Utilizzo di CVS (concurrent
versioning system)

[Partenza](#)

[Risali](#)

[Avanti](#)
Configurare il ppp per connettersi al
proprio ISP

Guida Per Utonti

[Indietro](#)

Capitolo 2. Documenti non ancora classificati

[Avanti](#)

Utilizzo di CVS (concurrent versioning system)

A cura di:

Simone Piccardi

<piccardi@firenze.linux.it>

Il programma CVS è un sistema di archiviazione dei programmi che consente di tenere traccia di tutte le modifiche e di condividere un archivio comune per progetti portati avanti da diverse persone.

Introduzione

CVS è basato sul concetto di repository, un archivio in cui vengono riposti e da cui vengono presi i sorgenti dei programmi. L'archivio tiene traccia delle diverse versioni registrate; i programmatori inviano le modifiche usando una copia locale che hanno nella loro directory di lavoro.

CVS può gestire più di un progetto, esso organizza i progetti in *moduli* identificati dal nome della directory in cui sono messi i file relativi, ad esempio:

```
[piccardi@pcpamela ~]$ ls /usr/local/cvsroot/  
CVSROOT adidsp geometry muonacq pamela
```

in questo caso si hanno i moduli *adidsp geometry muonacq pamela*; un utente potrà recuperare tutti i file relativi al modulo *pamela* che andranno nella directory *pamela*; è anche possibile gestire una gerarchia di moduli, inseriti in un albero di directory.

CVS ha una directory base dove tiene gli archivi (che nel è appunto **/usr/local/cvsroot**) e i vari file amministrativi; ogni progetto deve fare riferimento ad essa; in un progetto attivo essa viene memorizzata nei file amministrativi locali del progetto (ad esempio per un utente che ha il progetto *muonacq* nella sua home directory sarà in **~/muonacq/CVS**) e non è necessario specificarla; in generale essa viene tenuta dalla variabile di shell **CVSROOT** (inizializzata allo startup), o specificata direttamente dall'utente con l'apposito switch **-d** (ex. **cvs -d /usr/local/cvsroot comando opzioni**).

Normalmente l'archivio si tiene su una macchina remota e vi si accede via rete. Per repository esterni esistono varie modalità di accesso, la via più semplice è quella dell'uso di *ssh* come rimpiazzo di *rsh* per l'accesso esterno; in tal caso la directory del repository si può accedere con la sintassi:

```
cvs -d :ext:utente@server:/usr/local/cvsroot comando
```

questo però comporta che sulla macchina remota *server* sia installato il server *ssh* ed esista l'utente *utente* con privilegi di accesso ai files; inoltre sulla macchina ospite deve esser stata definita la variabile di shell

CVS_RSH=ssh .

In questo modo ssh si collega alla macchina remota ed esegue il comando CVS con una connessione criptata dopo aver richiesto la password di *utente*.

Utilizzo

Per creare un repository a partire da un gruppo di programmi esistente basta dare il comando **cvs import project tag release** nella directory dei sorgenti; verrà creato nel repository il modulo di nome *project* a cui poi si potrà accedere con i successivi comandi; *tag* è una label di versione iniziale (ex. *1.1.1*) dato a tutto il blocco e *release* una label di versione data ai programmi.

Una volta creato il repository è d'uopo cancellare la directory e ripartire dal progetto appena creato. Per recuperare ex-novo tutti i file di un progetto il comando da dare è:

```
cvs [-d ...] checkout project [-r rel] [-D date]
```

che creerà la directory *project* nella directory corrente con l'ultima versione dei file archiviata; se non si vuole la versione più aggiornata, ma una versione precedente si può usare lo switch **-r** che scaricherà la versione identificata dall'etichetta specificata (vedi la parte seguente sul comando *tag*) o lo switch **-D** che scaricherà la versione più recente prima della data specificata.

Una volta scaricato il progetto è possibile evitare di specificare la directory base del repository con l'opzione **-d** qualora i comandi vengano dati all'interno della directory creata, essa viene tenuta nel file *CVS/Root*, e può essere cambiata (qualora si usino più repository) editando detto file.

Una volta creata la propria copia locale dei programmi, è possibile lavorare su di essi stando nella relativa directory, e apportare tutte le modifiche che si vogliono; due comandi permettono di schedulare la rimozione o l'aggiunta di file al repository:

```
cvs add file1.c
cvs remove file2.c
```

ma niente viene modificato nel repository fintanto che non viene dato il comando **commit**:

```
cvs commit [file]
```

(è possibile mandare le modifiche anche per il singolo file).

Questi comandi comunque non effettuano le modifiche se i file del repository nel frattempo sono stati modificati; in questo caso rilevano le differenze e restituiscono un *merging* delle versioni locale/globale nella directory di lavoro, che è compito del programmatore esaminare per eliminare eventuali contrasti.

Per esempio viene eseguito un commit su una versione già modificata da un altro sul repository, il programma segnalerà che c'è un conflitto e chiederà al ``committente" di intervenire sui files per i quali sono stati rilevati i conflitti .

Le sezioni di codice in conflitto sono separate come:

```
<<<<<<< Makefile
      $(CC) $(CFLAGS) -o pamacq pamacq.c -lm
=====
      $(CC) $(CFLAGS) -o pamacq pamacq.c
>>>>>>> 1.22
```

nel caso si c'e' stata una modifica sul file (mostrata nella parte superiore) incompatibile con quella fatta nel repository (mostrata nella parte inferiore). Prima di eseguire un *commit* occorre pertanto integrare le modifiche e risolvare il file; a questo punto il *commit* diventa possibile.

Comandi utili

Ulteriori comandi utili per controllare sono:

-

cvs update

scarica le eventuali modifiche dei file del repository e le applica ai file nella directory corrente. In caso di conflitti con modifiche locali effettua un merging con le caratteristiche esposte in precedenza.

-

cvs status

compara i file nella directory locale con quelli del repository e stampa a schermo le informazioni di stato di ciascuno di essi:

- ◆

Locally Modified: il file è stato modificato localmente

- ◆

Up to date: il file è identico

- ◆

Needs Patch il file è cambiato sul repository (va updatato)

- ◆

Needs Merge il file e' cambiato sul repository e localmente, le differenze vanno riunite prima del commit, in genere basta un **cv**s update , ma in caso di conflitti questi vanno sanati.

-

cvcs log

legge il log del modulo corrente; in questo modo si hanno le informazioni sullo stato del progetto all'interno del repository, per ciascun file vengono mostrati informazioni generali su:

- ◆ release attuale
 - ◆ presenza di eventuali ramificazioni
 - ◆ lista di etichette simboliche
 - ◆ i messaggi di commento salvati ad ogni *commit*
- **cvcs rtag Etichetta modulo**

permette di associare una etichetta alla corrente versione dei file nel progetto così come sono nel repository; in modo da poter ritornare a quello stato del software con il comando **cvcs checkout -rEtichetta modulo**.

[Indietro](#)

Breve manuale sull'uso del tool
make

[Partenza](#)

[Risali](#)

[Avanti](#)

Utilizzo dei winmodem Lucent

Guida Per Utonti

[Indietro](#)

Capitolo 2. Documenti non ancora classificati

[Avanti](#)

Breve manuale sull'uso del tool make

A cura di:

Simone Piccardi

[<piccardi@firenze.linux.it>](mailto:piccardi@firenze.linux.it)

Queste brevi note vogliono essere un'introduzione all'uso di un tool di costruzione dei programmi ampiamente usato nella comunità degli sviluppatori, che consentono una facile manutenzione dei programmi, e una procedura di compilazione dei medesimi rapida ed efficiente.

Introduzione: cosa e' *make*

Il comando **make** serve per effettuare una compilazione intelligente di tutti i file relativi a un programma (subroutines e routine principale) ricompilando solo i file necessari.

Con **make** si possono definire i simboli del preprocessore C che consentono la compilazione condizionale dei programmi (anche in Fortran); è pertanto possibile gestire la ricompilazione dei programmi con diverse configurazioni con la modifica di un unico file.

La sintassi normale del comando (quella che si usa quasi sempre, per le opzioni vedere la man page) è semplicemente **make**. Questo comando esegue le istruzioni contenute in un file standard (usualmente *Makefile*, o *makefile* nella directory corrente).

Il formato normale dei comandi contenuti in un *Makefile* è:

```
bersaglio: dipendenza1 dipendenza2 ...
          regola1
          regola2
          ...
```

dove lo spazio all'inizio deve essere un *TAB*, il bersaglio e le dipendenze nomi di files e le regole comandi di shell.

Il concetto di base è che se uno dei file di dipendenza è più nuovo del file bersaglio (sulla base della sua data di modifica) il bersaglio viene ricostruito usando le regole elencate sotto.

Il comando **make** ricostruisce di default il primo bersaglio che viene trovato nella scansione del *Makefile*, se in un *Makefile* sono contenuti più bersagli indipendenti, si può farne ricostruire un'altro che non sia il primo con il comando esplicito *make bersaglio*.

Si tenga presente che le dipendenze stesse possono essere dichiarate come bersagli dipendenti da altri file; in questo modo è possibile creare una catena di ricostruzioni.

In esempio comune di quello che si fa è mettere come primo bersaglio il programma principale che si vuole usare, e come dipendenze tutte gli oggetti delle sue subroutine, con i quali deve essere linkato; a loro volta questi oggetti sono bersagli che hanno come dipendenza i relativi sorgenti. In questo modo il cambiamento di una sola delle subroutine comporta solo la ricompilazione della medesima e del programma finale.

Utilizzo

Make mette a disposizione una serie molto complesse di opzioni e di regole standard predefinite e sottintese, che permettono una gestione estremamente rapida e concisa di progetti anche molto complessi; per questo piuttosto che fare una replica del manuale preferisco commentare un esempio di makefile, quello usato per ricompilare i programmi di analisi dei dati dei test su fascio del tracciatore di Pamela:

```
#-----
#
# Makefile for a Linux System:
# use GNU FORTRAN compiler g77
# Makefile done for tracker test data
#
#-----
# Fortran flags
FC=g77
FFLAGS= -fvxt -fno-automatic -Wall -O6 -DPC # -DDEBUG
CC=gcc
CFLAGS= -Wall -O6
CFLADJ=-c #-DDEBUG
#
# FC          Fortran compiler for standard rules
# FFLAGS      Fortran flags for standard rules
# CC          C Compiler for standard rules
# CFLAGS      C compiler flags for standard rules
LIBS= -L/cern/pro/lib -lkernlib -lpacklib -lgraflib -lmathlib
OBJ=cnoise.o fit2.o pedsig.o loop.o badstrp.o cutcn.o readevnt.o \
erasepedvar.o readinit.o dumpval.o writeinit.o

riduzione: riduzione.F $(OBJ) commondef.f readfile.o
        $(FC) $(FFLAGS) -o riduzione riduzione.F readfile.o $(OBJ) $(LIBS)

readfile.o: readfile.c
        $(CC) $(CFLAGS) -o readfile.o readfile.c

$(OBJ): commondef.f

.PHONY : clean
clean:
        rm -f *.o
        rm -f *~
        rm -f riduzione
        rm -f *.rz
        rm -f output
```

Anzitutto i commenti: ogni linea che inizia con un *#* è un commento e non viene presa in considerazione. Con *make* possono essere definite delle variabili, da potersi riusare a piacimento, per leggibilità si tende a definirle tutte maiuscole, nell'esempio ne sono definite varie:

```

FC=g77
FFLAGS= -fvxt -fno-automatic -Wall -O6 -DPC # -DDEBUG
CC=gcc
CFLAGS= -Wall -O6
CFLADJ=-c #-DDEBUG
...
LIBS= -L/cern/pro/lib -lkernlib -lpacklib -lgraflib -lmathlib
OBJ=cnoise.o fit2.o pedsig.o loop.o badstrp.o cutcn.o readevnt.o \

```

La sintassi è **NOME=**, alcuni nomi però hanno un significato speciale (nel caso **FC**, **FLAGS**, **CC**, **CFLAGS**) in quanto sono usati da *make* nelle cosiddette *regole implicite* (su cui torneremo dopo).

Nel caso specifico, vedi anche i commenti, abbiamo definito i comandi di compilazione da usare per il C e il Fortran, e i rispettivi flag, una variabile che contiene il path e la lista delle librerie del CERN e una variabile con una lista di file oggetto.

Per richiamare una variabile si usa la sintassi **\$(NOME)**, ad esempio nel makefile abbiamo usato:

```
$(FC) $(FFLAGS) -o riduzione riduzione.F readfile.o $(OBJ) $(LIBS)
```

e questo significa che la regola verrà trattata come se avessimo scritto esplicitamente i valori delle variabili.

Veniamo ora alla parte principale del makefile che esegue la costruzione del programma:

```

riduzione: riduzione.F $(OBJ) commondef.f readfile.o
    $(FC) $(FFLAGS) -o riduzione riduzione.F readfile.o $(OBJ) $(LIBS)
readfile.o: readfile.c
    $(CC) $(CFLAGS) -o readfile.o readfile.c
$(OBJ): commondef.f

```

Il primo bersaglio del makefile, che definisce il bersaglio di default, è il programma di riduzione dei dati; esso dipende dal suo sorgente da tutti gli oggetti definiti dalla variabile **OBJ**, dal file di definizioni *commondef.f* e dalla routine C *readfile.o*; si noti il *.F* del sorgente, che significa che il file prima di essere compilato viene fatto passare attraverso il preprocessore C (cosa che non avviene per i *.f*) che permette di usare i comandi di compilazione condizionale del preprocessore C con la relativa sintassi. Sotto segue il comando di compilazione che sfrutta le variabili definite in precedenza per specificare quale compilatore e flags usare e specifica di nuovo gli oggetti e le librerie.

Il secondo bersaglio definisce le regole per la compilazione della routine in C; essa dipende solo dal suo sorgente. Si noti che per la compilazione vengono usate le variabili relative al compilatore C. Si noti anche che se questa regola viene usata, allora lo sarà anche la precedente, dato che *riduzione* dipende da *readfile.o.o*.

Il terzo bersaglio è apparentemente incomprensibile dato che vi compare solo il riferimento alla variabile **OBJ** con una sola dipendenza e nessuna regola; essa però mostra le possibilità (oltre che la complessità) di *make* connesse alla presenza di quelle regole implicite a cui avevamo accennato.

Anzitutto una peculiarità di *make* è che si possono anche usare più bersagli per una stessa regola (nell'esempio quelli contenuti nella variabile **OBJ** che viene espansa in una lista); in questo caso la regola di costruzione sarà applicata a ciascuno che si potrà citare nella regola stessa facendo riferimento con la variabile automatica: **\$@**. L'esempio usato per la nostra costruzione però sembra non avere neanche la regola di costruzione.

Questa mancanza sia di regola che di dipendenze (ad esempio dai vari sorgenti) illustra le capacità di funzionamento automatico di *make*. Infatti è facile immaginarsi che un oggetto dipenda da un sorgente, e che per ottenere l'oggetto si debba compilare quest'ultimo.

Make sa tutto questo per cui quando un bersaglio è un oggetto (cioè ha un nome tipo *qualcosa.o*) non è necessario specificare il sorgente, ma il programma lo va a cercare nella directory corrente (ma è possibile pure dirgli di cercarlo altrove, vedi il manuale). Nel caso specifico allora si è messo come dipendenza solo il file delle definizioni che viene incluso in ogni subroutine. Inoltre come dicevamo in genere per costruire un oggetto si deve compilarne il sorgente; *make* sa anche questo e sulla base dell'estensione del sorgente trovato (che nel caso sarà un *qualcosa.f*) applica la regola implicita. In questo caso la regola è quella di chiamare il compilatore fortran applicato al file oggetto e al relativo sorgente, questo viene fatto usando la variabile **FC** che è una delle variabili standard usata dalle regole implicite (come **CC** nel caso di file *.c*); per una maggiore flessibilità poi la regola standard usa anche la variabile **FFLAGS** per specificare, a scelta dell'utente che non ha che da definirla, quali flag di compilazione usare (nella documentazione sono riportate tutte le regole implicite e le relative variabili usate).

In questo modo è stato possibile usare una sola riga per indicare la serie di dipendenze e relative compilazioni delle singole subroutine; inoltre con l'uso della variabile **OBJ** l'aggiunta di una nuova eventuale routine *nuova.f* comporta solo l'aggiunta di *nuova.o* alla definizione di **OBJ**.

[Indietro](#)

Documenti non ancora classificati

[Partenza](#)

[Risali](#)

[Avanti](#)

Utilizzo di CVS (concurrent
versioning system)

Guida Per Utonti

[Indietro](#)

Capitolo 2. Documenti non ancora classificati

[Avanti](#)

La sicurezza in MySql

A cura di:

Claudio Cicali

<claudioc@firenze.linux.it>

L'amministrazione dei privilegi in MySql può essere molto raffinata e complessa permettendo di specificare i permessi di qualsiasi tipo di operazione sui database basandosi sui concetti di user, password e host dalla quale avviene la connessione.

Utenti e permessi

Può essere specificato, per esempio, che l'utente "user01" può solo connettersi dalla macchina locale e che questi può effettuare solo delle SELECT soltanto sul database specificato. Qualsiasi altra operazione non gli sarà permessa.

Esiste un utente predefinito "root" il quale può compiere qualsiasi operazione su qualsiasi database. Per semplificare l'amministrazione del sistema, subito dopo l'installazione di MySql, l'utente root non ha password.

E' tuttavia consigliabile associare una password a tale utente *appena possibile* !

Pur essendo piuttosto raffinata, l'amministrazione dei privilegi è abbastanza semplice, una volta capito i concetti che ne stanno alla base. Tutte le volte che occorre fare una certa operazione su un database (sia essa una SELECT o una UPDATE) occorre specificare dal client al server a quale database ci vogliamo collegare, con quale utente e password e da quale host. Se utilizziamo il client da riga comando (mysql.exe) lo potremo specificare con il seguente comando:

```
[claudioc@melele:~]$ mysql --user=user01 --password=mypasswd  
--host=193.122.123.11 db1
```

In questo modo chiediamo al server di connettersi al database "db1" come utente "user01" e password "mypasswd". La connessione viene richiesta al server che risiede sulla macchina "193.122.123.11" (se l'host non fosse specificato, si assume che il server stia girando sulla stessa macchina del client)

Prima di permettere l'accesso al database, il server mysql controllerà: – che l'utente esista sul suo database degli utenti – che la password specificata sia corretta – che quell'utente possa connettersi da remoto – che quell'utente possa connettersi al database db1. Inoltre, tutte le volte che l'utente tenterà una operazione sul database (come una SELECT su una tabella, per esempio), il server mysql controllerà che tale utente ne abbia l'autorità. Appena una di queste condizioni non fosse verificata, il server rifiuterà o la connessione o l'operazione.

Aggiungere utenti e permessi

Il comando per aggiungere utenti e permessi al sistema e' "GRANT".

Facciamo l'esempio in cui sul nostro sistema esista il database "DB1".

Connettiamoci al server mysql (che assumeremo girare sulla stessa macchina del client) tramite il client da riga di comando e colleghiamoci al database di sistema, mysql. Su questo database vengono registrati tutti gli utenti e tutti i privilegi di MySQL

```
[claudioc@melele:~]$ mysql -u root mysql
```

Supponiamo di volere inserire l'utente "phpuser01", e di dover garantire a questo la possibilita' di effettuare SELECT, UPDATE e DELETE su tutte le tabelle del database "db1". Come limitazione esigeremo che tale utente si potra' connettere solo dalla macchina sul quale sta girando il server MySQL. All'utente associeremo la password "mypasswd"

```
> GRANT SELECT,UPDATE,DELETE ON db1.* TO phpuser01@localhost IDENTIFIED BY 'mypasswd' ;
```

Fatto !

Nel caso delle procedure web, il discorso e' simile. In pratica il php, ad esempio, dovra' semplicemente essere censito come user al MySQL, applicando tutte le restrizioni del caso. Se non ci sono particolari indicazioni, la GRANT dell'esempio puo' gia' essere sufficiente.

Una volta censito in questo modo, tutte le procedure PHP che avranno bisogno di utilizzare i servizi del MySQL server, si conetteranno come quell'utente e il server MySQL provvederà ad applicare tutte le restrizioni del caso.

Da notare che, specificando che l'utente "phpuser01" puo' connettersi solo da localhost, nessuno (al di fuori dell'utente "root" del quale pero' avremo specificato una robusta password...) potra' connettersi al nostro database, e dunque fare potenziali danni, da una macchina remota.

Questo non significa che gli utenti del sito non possano navigare ! Il php, infatti, girerà in locale sulla macchina dove risiede sia il web server che il MySQL server. Solo lui avrà necessità di effettuare connessioni al server per creare le pagine dinamicamente, non l'utente web.

Modificare le password

Anche in questo caso MySQL ci fornisce un apposito comando: PASSWORD

Abbiamo appena installato mysql ed abbiamo deciso che e' giunto il momento di cambiare la password di root. Come facciamo ?

Collegiamoci al nostro server MySQL usando ancora il nostro client testuale e colleghiamoci come root al database "di sistema", mysql.

```
[claudioc@melele:~]$ mysql -u root mysql
```

Appena possibile digitiamo il comando:

```
> SET PASSWORD FOR root=PASSWORD('newpassword');
```

Ecco fatto. Da adesso, per collegarsi come utente "root", dovra' essere specificata anche la password 'newpassword'.

Un metodo piu' veloce potrebbe essere quello di utilizzare l'altro programma di utilita' fornito con MySQL: mysqladmin. In questo caso, per cambiare la password all'utente root sara' sufficiente il comando:

```
[claudioc@melele:~]$ mysqladmin -u root password new_password
```

Un piccolo approfondimento

Essendo "mysql" il database di sistema, tutte le sue tabelle contengono i dati riferiti agli utenti e ai loro permessi. Le tabelle interessanti, in questo caso, sono "db" e "user". Nella prima sono presenti i permessi degli utenti sui vari database, nella seconda gli utenti e la loro password. La password e' comunque salvata in maniera crittografata.

Attenzione ! Se si vuol lavorare direttamente su queste tabelle, una volta capitate la (semplice) struttura, ricordarsi sempre di eseguire il comando "FLUSH PRIVILEGES", non necessario se si usa il comando "GRANT".

[Indietro](#)

Usare fetchmail

[Partenza](#)

[Risali](#)

[Avanti](#)

Installazione e configurazione del
kernel 2.4.x

Guida Per Utonti

[Indietro](#)

Capitolo 2. Documenti non ancora classificati

[Avanti](#)

Configurare il ppp per connettersi al proprio ISP

A cura di:

Maurizio Mollicone
<maurmoll@tin.it>

Per poter eseguire un connessione ad Internet tramite un provider abbiamo bisogno del pppd, incluso in tutte le distribuzioni. Vedremo come utilizzare il pppd e chat per comunicare col modem e chiamare il provider semplicemente editando due file.

Il kernel e il supporto di rete

Per prima cosa si deve verificare che il supporto per il TCP/IP e il PPP siano presenti nel kernel. Provate a digitare il comando:

```
ugetsu:~# grep -i "TCP/IP" /var/log/messages
```

Se ottenete una riga simile a questa:

```
Nov 12 14:19:07 ugetsu kernel: NET4: Linux TCP/IP 1.0 for NET4.0
```

allora il supporto per il TCP/IP compilato nel kernel. Successivamente controllate che nel kernel sia compilato il supporto per il ppp sia staticamente che come modulo. Digitate il comando:

```
ugetsu:~# dmesg |less
```

e controllate che, nei messaggi di boot, appaiano informazioni sul ppp simili a queste:

```
PPP: version 2.3.7 (demand dialling)  
TCP compression code copyright 1989 Regents of the University of California  
PPP line discipline registered.
```

Generalmente i kernel precompilati delle varie distribuzioni includono sempre il supporto per il ppp, magari come modulo.

A questo punto ci si deve assicurare di avere il modem correttamente configurato, inoltre bisogna conoscere la porta seriale a cui è collegato:

•

DOS com1 = Linux /dev/ttyS0

- DOS com2 = Linux /dev/ttyS1

Se si utilizza un modem esterno con velocità superiore a 14400 Baud, allora la porta seriale deve essere capace di gestire il flusso di dati prodotto dal modem. Questo richiede che la seriale abbia un UART (Universal Asynchronous Receiver Transmitter) a 16550A. Queste informazioni possono essere ricavate digitando:

```
ugetsu:~# setserial -a /dev/ttyS1
```

Se la macchina o la scheda sono vecchie, allora è possibile che la porta seriale abbia solo 8250 UART. Questo può causare problemi con modem ad alte velocità. Per configurare correttamente il modem bisogna leggere attentamente il manuale, anche se molti modem hanno delle configurazioni standard di fabbrica che lavorano bene con il ppp. Attenzione ai cosiddetti winmodem (a cui è dedicato un capitolo a parte! :) Su internet è possibile reperire delle configurazioni d'esempio per una grande varietà di modem all'indirizzo: www.in.net/info/modems/index.html Per cambiare configurazioni o effettuare controlli si può utilizzare un qualunque programma di comunicazione come minicom o seyon.

Informazioni sull'ISP

A questo punto abbiamo bisogno di alcune informazioni riguardanti l'ISP.

- Il numero di telefono per accedere al servizio. Senza questo numero non possiamo fare niente!!!
- Numero IP statico o dinamico? Se il server utilizza numeri IP statici allora bisogna sapere questo numero, e se fornisce una sottorete di numeri validi anche la maschera di rete. Molti ISP forniscono indirizzi dinamici. Ad ogni modo anche se si utilizza l'IP statico, molti server PPP non lasceranno mai scegliere all'utente quale IP usare per motivi di sicurezza.
- Quali sono i numeri IP dei Domain Name Servers del provider? Dovrebbero essere almeno 2 anche se ne basta uno solo.
- Il server richiede il metodo PAP o CHAP? In tal caso servono l' "id" e il "secret", cioè il nome utente e la password.
- Il server fa partire automaticamente il ppp o c'è bisogno di qualche comando particolare?
- Il server è Microsoft Windows NT? Se sí, utilizza il sistema MS PAP/CHAP?

Una volta scoperte tutte queste informazioni possiamo passare a configurare la connessione.

I file di connessione del ppp

Il ppp utilizza alcuni file per leggere le opzioni e i comandi da impartire al modem. Questi sono:

- il file `options`, generalmente in `/etc/ppp/` che contiene le opzioni da passare al comando `pppd`;
- il `chatfile`, contenente i parametri per la chiamata.

Il `pppd` viene chiamato con le opzioni descritte nel file `options` e a sua volta chiama `chat`, un programma che permette di inviare le stringhe descritte nel `chatscript` al modem.

Una configurazione standard del file `/etc/ppp/options` potrebbe essere la seguente:

```
lock
debug
defaultroute
noipdefault
modem
/dev/ttySx
115200
crtsets
passive
asynmap num
```

L'opzione `modem`, in realtà, è superflua: utilizza le linee di controllo del modem ed è il comportamento di default mentre `debug` serve per ottenere informazioni su come sta andando la negoziazione della connessione. `defaultroute` abilita il `pppd` ad aggiungere un instradamento di default che viene eliminato quando cade la connessione ppp. Disabilita il comportamento di default che è quello di determinare l'IP locale dal nome dell'host. In questo modo il pari deve specificare l'IP locale durante la negoziazione. La riga `/dev/ttySx` specifica la porta a cui è collegato il modem e quella successiva è la velocità in baud. `crtsets` indica l'utilizzo del controllo del flusso hardware, cioè sarà il modem a gestire il flusso di dati che passa sulla porta seriale, mentre `asynmap` specifica quali caratteri di controllo non possono essere accettati sulla linea seriale. Chiaramente non è detto che qualcosa del genere funzioni per tutte le connessioni, e tutti gli ISP, in ogni caso con le informazioni dateci dal provider e la dettagliatissima man page del ppp non ci dovrebbero essere problemi.

Per quanto riguarda il `chatscript` ecco un esempio che può andare bene per gran parte delle configurazioni:

```
TIMEOUT 60
ABORT ERROR
ABORT BUSY
ABORT "NO CARRIER"
ABORT "NO DIALTONE"
" " "ATZ"
OK "ATDTXXXXXXXXXX"
```

```
TIMEOUT 75
CONNECT
```

Vediamo in dettagli cosa fa questo script. La prima riga imposta il timeout del pppd nel caso non riesca a trovare la linea, mentre le successive quattro terminano il processo nel caso ci siano errori, la linea sia occupata o comunque non si riesca a stabilire una comunicazione tramite il modem. Quindi viene passata al modem la stringa di inizializzazione e il numero da chiamare. Importante: per questa parte occorre far riferimento al manuale del modem! Alcuni modem sono tarati per le linee americane e richiedono una stringa leggermente diversa per chiamare in Italia. Si può provare ad inserire la stringa

```
OK "ATX3DTXXXXXXXXX"
```

per chiamare il numero del provider. Volendo controllare il volume del modem basta sostituire alla stringa di inizializzazione:

```
" " "ATZ"
```

la stringa:

```
" " "ATMO"
```

In questo modo il volume del modem è azzerato. Infine viene impostato il timeout per la risposta del server e quindi si aspetta la connessione. Una nota riguardo ai vari server: alcuni sono abbastanza "intelligenti", e, se usano PAP, non c'è bisogno di nulla, se non del file pap-secrets. Altri, invece, presentano un terminale, con i classici prompt username e password. In tal caso basta aggiungere dopo CONNECT le seguenti righe:

```
ogin:--ogin:   nome_utente
assword:      password_utente
```

In questo modo diciamo al chatscrip di aspettare una stringa e di rispondergli con la corretta informazione. Inoltre può accadere di dover avviare il ppp sul server da soli, in tal caso dobbiamo farci dire dal provider qual è questo comando. Comunque tutti i provider che oggi ci offrono abbonamenti e connessioni gratuite non utilizzano più sistemi del genere, ma utilizzano PAP/CHAP, quindi dobbiamo solo preoccuparci di settare correttamente il pap-secret/chat-secret.

L'ultimo file che dobbiamo modificare è /etc/resolv.conf. Qui vanno inseriti gli indirizzi IP del DNS. Generalmente sono due, ma se ne utilizza solo uno. Il file resolv.conf ha questa struttura:

```
search dominio.del.provider
```

```
nameserver w.x.y.z
nameserver a.b.c.d
```

Dopo aver inserito queste informazioni siamo pronti finalmente per navigare! Basta lanciare il comando (per adesso come root):

```
/usr/sbin/pppd -detach connect "/usr/sbin/chat -vf /etc/ppp/chatsript"
```

dove chatsript è il proprio file chat. A questo punto, se sullo schermo appare qualcosa del genere vuol dire che siamo in rete!

Dato che solo l'utente root ha diritto di eseguire il comando, possiamo scegliere se diventare superutente, connetterci, e poi navigare come utente non privilegiato, oppure permettere che un utente qualsiasi possa invocare il pppd. In questo caso per prima cosa controlliamo se esiste un gruppo per gli utenti del ppp. Sotto Redhat questo gruppo è pppusers, sotto Debian dip, e aggiungiamo l'utente a quel gruppo, modificando /etc/group. Quindi settiamo il suid bit del pppd e modifichiamo il gruppo di appartenenza:

```
ugetsu:~# chown root.dip /usr/sbin/pppd
```

```
ugetsu:~# chmod +s /usr/sbin/pppd
```

A questo punto anche l'utente non privilegiato è in grado di eseguire e stabilire una connessione con il provider. La scelta di avere il pppd settato con il bit suid è poco pulita ma per sistemi casalinghi è accettabile.

Riferimenti

Per ulteriori dettagli ci sono le pagine di manuale di pppd e chat, mentre su <http://www.pluto.linux.it/ildp/howto> ci sono le traduzioni italiane del Modem-HOWTO e del PPP-HOWTO.

[Indietro](#)

Utilizzo dei winmodem Lucent

[Partenza](#)

[Risali](#)

[Avanti](#)

Installazione di Debian GNU/Linux
2.2 (Potato)

Guida Per Utonti

[Indietro](#)

Capitolo 2. Documenti non ancora classificati

[Avanti](#)

Come installare ReiserFS.

A cura di:

Simone Piccardi

[<piccardi@firenze.linux.it>](mailto:piccardi@firenze.linux.it)

Le seguenti note spiegano come installare ReiserFS. Si danno per note alcune conoscenze basilari come saper ricompilare il kernel, o scompattare e installare programmi a partire dai sorgenti. Si dà inoltre per scontato che si sia già effettuato l'upgrade al kernel 2.4.x (e risolti i relativi problemi di upgrade dei software necessari) o si sia comunque in grado di farlo (vedi ... quando l'avrò scritto)

Installazione

Dal kernel 2.4.1 reiserfs è stato inserito nel kernel standard, pertanto non è più necessario applicare le patch, la procedura pertanto si è notevolmente semplificata.

Il primo passo è quello di procurarsi i sorgenti del kernel, scompattarli e fare partire il programma di configurazione (make menuconfig da terminale, make xconfig sotto X).

Per poter accedere alla compilazione di reiserfs occorre anzitutto abilitare le opzioni di sviluppo:

```
Code Maturity Level --->
[*] Promp for development and/or incomplete code/drivers
```

a questo punto nel menù:

```
File Systems --->
```

saranno divenute accessibili le opzioni:

```
<M> Reiserfs support
  [ ] Have reiserfs do extra internal checking
```

(si sono ripresi testi di menuconfig).

Si tratta pertanto di attivare reiserfs inserendolo come modulo e lasciare non selezionata l'altra opzione.

Se si vuole utilizzare reiserfs per la partizione di root occorre compilarlo nel kernel, usarlo per il boot è comunque sconsigliato, in quanto lilo non dovrebbe ancora supportarlo.

Nel caso in questione si farà conto di avere selezionato reiserfs come modulo.

Fatto questo basta ricompilare kernel e moduli ed installare il tutto.

Il secondo passo è quello di procurarsi i reiserfs-utils, compilarli ed installarli. La versione più aggiornata che ho trovato è:

```
reiserfs_utils-3.6.25-fsck-3.x.0b.tar.gz
```

scompattato questo i comandi:

```
make  
make install
```

compilano ed installano il tutto.

Note per l'uso con debian potato

aggiungere in sources.list:

```
deb-src http://http.us.debian.org/debian unstable main contrib non-free
```

dopo di che compilazione ed installazione possono essere fatti con:

```
apt-get update  
apt-get source reiserfsprogs --compile  
dpkg -i reiserfsprogs_3.x.0a-1_i386.deb
```

che rispettivamente producono e installano il .deb (si fa riferimento alla versione al momento della stesura del presente documento).

Note per l'installazione del kernel 2.4.x

Queste sono note semplificate, non è detto funzionino correttamente su qualunque macchina, dovrebbero andar bene nella maggior parte dei casi (se non altro per me han funzionato).

Per l'installazione non basta ricompilare il kernel, occorre pure aggiornare una serie di pacchetti, il cui elenco è in:

linux/Documentation/Changes

La procedura di ricompilazione è la stessa, ma sono state cambiate (e create di nuove) molte opzioni rispetto alla serie 2.2.x. In particolare i cambiamenti più significativi sono:

Guida Per Utonti

Processor type and features	---	permette di selezionare anche le CPU AMD abilita il riconoscimento per > 2Gb ram
General setup	---	ridotto, molte parti sono state spostate
Memory Technology Devices (MTD)	---	nuovo menù per la gestione di device di memoria (per l'embedded)
Parallel port support	---	separato il supporto per la parallela
Plug and Play configuration	---	comprende il PNP per le schede ISA
Block devices	---	resta qui solo la configurazione dei formati degli eseguibili e di alcuni device come floppy, loopback, NBD, paride etc.
Multi-device support (RAID ...)	---	supporto per il raid software e l'LVM
...		
Networking options	---	usa iptables al posto di ipchain
ATA/IDE/MFM/RLL support	---	nuova sezione per i dispositivi IDE
...		
IEEE 1394 (FireWire) support	---	supporto per la firewire
...		
File systems	---	c'è anche il reiserfs (dal 2.4.1)
Input core support	---	per le tastiere mouse etc su USB
Character devices	---	supporto per agp e le DRI

La sola differenza sostanziale nella procedura di ricompilazione sta nella scelta delle opzioni, tutto il resto (compilazione, installazione con lilo o quel che si vuole) è identico al 2.2.x e lo tralascio.

Il passo successivo per potere usare il nuovo kernel è aggiornare alcuni pacchetti, la maggior parte sono già presenti su tutte le distribuzioni recenti, quelli che più probabilmente mancheranno sono:

modutils e2fsprogs (e quelli per la pcmcia, che non ho e non conosco)

ed a questi mi limito.

Le versioni correnti si possono ottenere usando lo specchietto riportato in linux/Documentation/Changes che riporta anche dove trovare le versioni che servono e le cose da fare (lettura raccomandata!!).

In ogni caso le nuove versioni dei pacchetti sunnominati si trovano su:

<ftp://ftp.kernel.org/pub/linux/utils/kernel/modutils/v2.4/>

<ftp://download.sourceforge.net/pub/sourceforge/e2fsprogs/e2fsprogs-1.19.tar.gz>

Tralascio i dettagli su come installarli, ne esistono degli rpm, ed è comunque sempre possibile ricompilarli, mi limito a coprire solo il caso della loro installazione su una Debian potato che non debba essere upgradata a woody.

Installare, se non c'è già, il pacchetto debhelper. Aggiungere in sources.list:

```
deb-src http://http.us.debian.org/debian unstable main contrib non-free
```

dopo di che l'installazione si effettua con:

```
apt-get update
apt-get source modutils --compile
apt-get source e2fsprogs --compile
dpkg -i modutils_2.4.1-2_i386.deb
dpkg -i e2fsprogs_1.19-3_i386.deb
```

In questo modo al reboot i moduli funzioneranno correttamente. Questa installazione copre il minimo indispensabile, il devfs deve essere disattivato, fstab aggiornato per l'uso della shared memory. Il sistema non è stato testato intensivamente, non si assumono responsabilità su niente ...

[Indietro](#)

Installazione di Debian GNU/Linux
2.2 (Potato)

[Partenza](#)

[Risali](#)

Guida Per Utonti

Capitolo 2. Documenti non ancora classificati

[Avanti](#)

Introduzione ai comandi
fondamentali di SQL

[Indietro](#)

[Avanti](#)

Introduzione ai comandi fondamentali di SQL

A cura di:

Claudio Cicali

<claudioc@firenze.linux.it>

Uno dei punti di forza del linguaggio SQL (Structured Query Language) è sicuramente la sua relativa semplicità sintattica la quale, unita alla sua estrema flessibilità, lo rende semplice da imparare ma nello stesso tempo potentissimo. Un altro vantaggio di SQL sta' nella sua `universalita'. Una volta imparatone perlomeno le basi, si può essere sufficientemente sicuri di poter `dialogare' con qualsiasi database moderno.

Scopo di questo documento

Questo breve documento vuol essere di aiuto a coloro che per la prima volta vogliono provare ad interagire con un database, ma che sono intimoriti dalla lettura di grossi tomi sull'argomento. Come ho detto la sintassi intrinseca dell'SQL e' piuttosto semplice, ma `dietro' all'SQL ci deve essere sempre un database. Capire a cosa serve un database, come i dati siano e debbano essere strutturati al suo interno, cosa sia la normalizzazione, o quali siano le differenze tra un database e l'altro, non e' compito di questo documento. Sicuramente tale compito sarà comunque di chi vuole usare in maniera `seria' un database e dovrà farlo utilizzando una letteratura piu' specifica ed esaustiva.

Inoltre, pur essendo l'SQL un linguaggio standard ANSI, ne esistono implementazioni piu' o meno standard da parte dei vari DBMS (Database Management System, come Oracle, MySql, SQLServer, Postgresql), soprattutto per quanto riguarda funzionalità avanzate. Cercherò dunque di rimanere sempre sull'SQL standard, quello cioè capito da tutti i DBMS.

Convenzioni di questo documento

Nel descrivere la sintassi dei vari comandi, utilizzerò la convenzione di scrivere in maiuscolo il comando e le altre parole chiave del SQL, mentre in minuscolo, racchiuso tra parentesi quadre quello che deve essere specificato dal programmatore. SQL non è comunque sensibile al minuscolo o al maiuscolo per quanto riguarda le sue parole chiave. Per i nomi dei campi e delle tabelle è necessario rispettare invece la convenzione usata al momento di creare la tabella e i suoi campi. Taluni DBMS lo tengono in considerazione, altri no.

Righe e colonne

Un database è come un contenitore. All'interno di questo contenitore ci possono essere diversi oggetti. L'oggetto più importante per noi adesso è la tabella. Le tabelle contengono i dati, logicamente suddivisi in righe e colonne, come in una matrice. La colonna identifica il dato, mentre nella riga vi è il contenuto vero e proprio. Estrarre una riga da una tabella di un database, equivarrà a dire: estrarre il valore delle colonne di una determinata riga di una tabella. Con un esempio sarà tutto più chiaro:

Poniamo il caso di dover salvare nel mio database le informazioni relative ai numeri telefonici dei miei conoscenti. Per questa operazione avrò bisogno di definire dunque una tabella (che potrò chiamare per esempio ``ElencoTelefonico`) composta dalle seguenti colonne:

```
``Nome", ``Cognome", ``Telefono_fisso", ``Telefono_mobile"
```

A questo punto, tutte le volte che inserirò all'interno della tabella quei dati relativi ad un mio conoscente, inserirò una riga (detta anche record) nella tabella. Nello stesso modo, quando dovrò reperire i dati di un conoscente, estrarrò dalla tabella la riga corrispondente al suo Nome e Cognome.

Nota: talvolta viene usato il termine ``campo" come sinonimo di ``colonna". Nell'esempio dunque, ``Cognome" è un campo della tabella ``ElencoTelefonico".

Quale comando per una determinata operazione ?

SQL in realtà è l'insieme di due sotto-linguaggi. Uno è il DDL (Data Definition Language) mentre l'altro è il DML (Data Manipulation Language). Il DDL è utilizzato per la creazione degli oggetti all'interno dei database (tabelle, indici, viste, procedure, etc) e comprende i comandi come CREATE, ALTER etc. Il DML invece comprende i comandi per interagire con i dati che sono all'interno del database, dunque per estrarli, modificarli o cancellarli. In questo documento affronterò solo i comandi DML, in quanto per parlare del DDL dovremmo scendere in troppi dettagli su come è costruito un database, su cosa sono gli indici, le viste etc. Inoltre, sebbene DML sia abbastanza standard, ogni DBMS ha poi le sue regole specifiche per la creazione degli oggetti. Queste regole specifiche riguardano certi tipi particolari di dato, certe ottimizzazioni da effettuare sugli oggetti al momento della creazione, la definizione degli spazi di allocazione necessari a questi oggetti, etc. Consiglio dunque di leggere il manuale del proprio database per creare le tabelle e gli oggetti ad esse associate.

Per richiedere dati dal database useremo il comando **SELECT**

Per inserire dati nel database useremo il comando **INSERT**

Per modificare dati dal database useremo il comando **UPDATE**

Per cancellare dati dal database useremo il comando **DELETE**

Nei paragrafi successivi, analizzeremo ognuno dei sopracitati comandi.

SELECT

Il comando select è sicuramente il più usato tra tutti quelli SQL. Il suo scopo è reperire i dati da una tabella o da più tabelle di un database. E' possibile definire quali campi specifici della tabella/e si vuole estrarre e secondo quali regole o filtri. Vediamone la sintassi di base:

```
SELECT [elenco dei campi] FROM [nome della tabella]
```

All'esecuzione del comando il DBMS ci fornirà l'elenco dei valori dei campi che abbiamo richiesto, presenti nella tabella specificata. Ovvero, estrarrà tutti i record della tabella. Se non specifichiamo nessun filtro di estrazione, ci verranno ritornati TUTTE le righe della tabella. Se vogliamo ottenere tutte le colonne della tabella non è necessario specificare ogni singolo campo della stessa. Sarà sufficiente inserire un asterisco

(`*`) al posto di [elenco campi]. Per esempio:

```
SELECT * FROM [nome della tabella]
```

Vediamo adesso come fare per ottenere solo un determinato sottoinsieme di tutte le righe della tabella. Quello che ci serve è un metodo per specificare una regola e fare in modo che il DBMS ritorni solo le righe che la soddisfano. La regola viene specificata con la parola chiave ``WHERE``:

```
SELECT [elenco dei campi] FROM [nome della tabella] WHERE [regola]
```

Nella [regola] dovremo specificare il modo secondo il quale il DBMS possa decidere quali tra tutte le righe della tabella vogliamo ottenere. Se per esempio nella tabella ``ElencoTelefonico`` volessimo ottenere solo i dati del conoscente di cognome ``Rossi`` dovremmo scrivere:

```
SELECT * FROM ElencoTelefonico WHERE Cognome = ``Rossi``
```

In questo modo, dato che il DBMS userà la regola specificata per estrarre le righe dalla tabella, solo quella (o quelle, se conosciamo più persone con cognome Rossi) che soddisfa la regola verrà estratta.

Volendo ottenere solo il numero di telefono del nostro conoscente Rossi dovremmo dare il seguente comando:

```
SELECT Telefono_fisso, Telefono_mobile FROM ElencoTelefonico WHERE  
Cognome = ``Rossi``
```

Si noti nell'esempio come l'elenco dei campi sia separato dalla virgola. E' possibile inoltre costruire clausole di where più sofisticato utilizzando le espressioni logiche booleane, utilizzando i classici operatori AND e OR. Ad esempio, estraiamo dalla nostra tabella tutti i dati relativi al cognome Cicali e Rossi:

```
SELECT * FROM ElencoTelefonico WHERE Cognome = ``Rossi`` AND Cognome =  
``Cicali``
```

INSERT

Il comando insert è il comando per inserire dati nel database. Ogni volta che viene eseguito, il comando inserisce una ed una sola riga nella tabella.

```
INSERT INTO [nome della tabella] ([elenco dei campi da valorizzare]) VALUES  
([valore dei campi])
```

L'elenco dei campi da valorizzare e i loro valori devono essere chiusi tra parentesi tonde e separati da una virgola, come per la select. Se si vogliono valorizzare tutti i campi della tabella, l'elenco dei campi da valorizzare non è obbligatorio. Il DBMS in quel caso capirà quello che vogliamo fare e provvederà automaticamente ad inserire il giusto valore nella giusta colonna. E' comunque buona norma specificare sempre l'elenco dei campi, anche nel caso si debba valorizzarli tutti. Ecco un esempio di inserimento nella tabella ElencoTelefonico:

```
INSERT INTO ElencoTelefonico (Nome, Cognome, Telefono_fisso,  
Telefono_mobile) VALUES (``Claudio``, ``Cicali``, ``0623939920``,
```

```
``0332444373'' )
```

Nel caso, per esempio, non avessimo subito a disposizione i numeri di telefono del nostro conoscente, ma volessimo comunque inserire una riga a lui corrispondente, potremmo scrivere:

```
INSERT INTO ElencoTelefonico (Nome, Cognome) VALUES (``Claudio'',  
``Cicali'')
```

Successivamente (vedremo tra poco il comando apposito, UPDATE) potremo modificare il contenuto di quella riga per aggiungere il valore dei campi che adesso abbiamo lasciato vuoti (Telefono_fisso e Telefono_mobile).

UPDATE

Con il comando update sarà possibile modificare il contenuto delle colonne di una o più righe della nostra tabella. E' un comando che deve essere utilizzato con la massima attenzione e ne spiegherò i motivi. Intanto, ecco la sintassi nella sua forma più semplice:

```
UPDATE [nome della tabella] SET [campo] = [valore]
```

Se si esegue questo comando, TUTTE le colonne [campo] di TUTTE le righe della tabella assumeranno il valore [valore]. Naturalmente è necessario un modo per specificare che l'aggiornamento deve avvenire solo su un determinato insieme di righe. Ancora una volta la parola chiave è WHERE:

```
UPDATE [nome della tabella] SET [campo] = [valore] WHERE [regola]
```

In questo modo, come per la select, il DBMS aggiornerà solo le righe che soddisfano la regola. Come accennavo pocanzi, la update si può rivelare uno strumento piuttosto pericoloso. In SQL non esiste, per intendersi, un comando di "undo" sull'operazione appena effettuata (in realtà su certi DBMS esiste, ma non nel senso che sto cercando adesso di spiegare...). Se si sbaglia la regola di estrazione di una select, poco male, si cambia la clausola della where e si ritenta. Ma nella update non si può ritentare ! Se si sbaglia la clausola della where o dimentichiamo di metterla potremmo combinare un disastro, modificando righe che non dovevano essere modificate. E senza la possibilità di tornare indietro. La soluzione è solo un consiglio: prima di lanciare una update, verificate sempre la clausola di WHERE con una select. Vi eviterete delle grosse grane. Per esempio: supponete di avere una tabella con 10000 righe e della quale dovete modificare il cognome di tutte le persone che hanno il cognome Cicali e il numero civico 32 o 2. Voi scrivete:

```
UPDATE Anagrafe SET Cognome = `Rossi' WHERE Cognome = `Cicali' and  
NumCiv = 32 or NumCiv = 2
```

e lanciate il comando. Avete combinato un grosso pasticcio ! In realtà il DBMS avrà fatto ben più di quello che avevate richiesto. Esso ha cambiato sì il cognome di Cicali dove questo aveva il numero civico 32, ma ha anche cambiato il cognome di tutti coloro che hanno il numero civico 2 ! A meno di non avere un backup recente del vostro database, siete nei pasticci. Eppure, prima di eseguire quel comando sarebbe bastato un:

```
SELECT * from anagrafe WHERE Cognome = `Cicali' and NumCiv = 32 or  
NumCiv = 2
```

per accorgevi del malefico errore. Il comando avrebbe dovuto essere scritto così:

```
UPDATE Anagrafe SET Cognome = `Rossi` WHERE Cognome = `Cicali` and
(NumCiv = 32 or NumCiv = 2)
```

sarebbe bastata solo una coppia di stupide parentesi !

Per tornare al nostro esempio dell'elenco telefonico, vediamo come possiamo aggiornare il record (o riga) inserito con l'esempio della insert. Andiamo ad inserire i dati mancanti del nostro conoscente ``Claudio``Cicali``

```
UPDATE ElencoTelefonico SET Telefono_fisso = ``055224343``,
Telefono_mobile=''0332343393'' WHERE Nome = ``Claudio`` AND `Cognome` =
``Cicali``
```

Ecco fatto. Notare come le coppie di colonna/valore siano divise da una virgola.

DELETE

Infine ecco il comando per cancellare le righe che non interessano più dalla nostra tabella. Se il comando UPDATE è pericoloso, questo lo è ancora di più. Per esso valgono gli stessi avvertimenti della update con in più la nota che se sbagliando con la update si ottengono delle modifiche non volute, sbagliando con la delete otterrete dell CANCELLAZIONI non volute. Non so se mi spiego. Ecco la sintassi:

```
DELETE [nome della tabella] WHERE [regola]
```

Con il comando delete non si agisce mai a livello di colonna, ma sempre di riga. Ovvero, ogni delete cancella dalla tabella tutte le righe che soddisfano la clausola. Come avrete notato, ho aggiunto subito la parola chiave WHERE in quanto, pur potendo utilizzare la delete anche senza (con il risultato di svuotare completamente la tabella), il 99% delle delete vengono comunque sempre ``controllate`` da un'apposita clausola where. Vediamo, nell'esempio, come cancellare tutte le righe del nostro elenco telefonico riguardanti Claudio Cicali:

```
DELETE ElencoTelefonico WHERE Nome = ``Claudio`` AND `Cognome` =
``Cicali``
```

[Indietro](#)

Come installare ReiserFS.

[Partenza](#)

[Risali](#)

Guida Per Utonti

[Avanti](#)

Usare fetchmail

[Indietro](#)

[Avanti](#)

Appendice A. Gli autori di GPU

La Guida Per Utonti è stata scritta da:

Autori.

Claudio Cicali, < claudioc@firenze.linux.it >

Gian-Luca Dei Rossi, < gideir@inwind.it >

Simone Piccardi, < piccardi@firenze.linux.it >

Maurizio Mollicone, < maurmoll@tin.it >

Umberto Genovese, < umbegen@tin.it >

Cosimo Vagarini, < cosimo@firenze.linux.it >

Sergio Ballestrero, < [???](#) >

TheOld, < [???](#) >

[Indietro](#)

[Partenza](#)

[Avanti](#)

Installazione e configurazione del
kernel 2.4.x

GNU Free Documentation License

Guida Per Utonti

[Indietro](#)

Appendice C. GNU Free Documentation License (traduzione italiana)

Questa è semplicemente una traduzione della licenza che accompagna questo documento, e non ha valore legale. Riferirsi alla versione originale in lingua inglese, acclusa in questo documento, per ogni questione di rilievo.

Versione 1.1, Marzo 2000

Copyright © 2000

Free Software Foundation, Inc.
9 Temple Place, Suite 330,
Boston,
MA
02111-1307
USA

Chiunque può copiare e distribuire copie letterali di questo documento di licenza, ma non ne è permessa la modifica.

0. PREAMBOLO

Lo scopo di questa licenza è di rendere un manuale, un testo o altri documenti scritti "liberi" nel senso di assicurare a tutti la libertà effettiva di copiarli e redistribuirli, con o senza modifiche, a fini di lucro o no. In secondo luogo questa licenza prevede per autori ed editori il modo per ottenere il giusto riconoscimento del proprio lavoro, preservandoli dall'essere considerati responsabili per modifiche apportate da altri.

Questa licenza è un "copyleft": ciò vuol dire che i lavori che derivano dal documento originale devono essere ugualmente liberi. È il complemento alla GNU General Public License, che è una licenza di tipo "copyleft" pensata per il software libero.

Abbiamo progettato questa licenza al fine di applicarla alla documentazione del software libero, perché il software libero ha bisogno di documentazione libera: un programma libero dovrebbe accompagnarsi a manuali che forniscano la stessa libertà del software. Ma questa licenza non è limitata alla documentazione del software; può essere utilizzata per ogni testo che tratti un qualsiasi argomento e al di là dell'avvenuta pubblicazione cartacea. Raccomandiamo principalmente questa licenza per opere che abbiano fini didattici o per manuali di consultazione.

1. APPLICABILITÀ E DEFINIZIONI

Questa licenza si applica a qualsiasi manuale o altra opera che contenga una nota messa dal detentore del copyright che dica che si può distribuire nei termini di questa licenza. Con "[Documento](#)", in seguito ci si riferisce a qualsiasi manuale o opera. Ogni fruitore è un destinatario della licenza e viene indicato con "voi".

Una "[versione modificata](#)" di un documento è ogni opera contenente il documento stesso o parte di esso, sia riprodotto alla lettera che con modifiche, oppure traduzioni in un'altra lingua.

Una "[sezione secondaria](#)" è un'appendice cui si fa riferimento o una premessa del [documento](#) e riguarda esclusivamente il rapporto dell'editore o dell'autore del [documento](#) con l'argomento generale del [documento](#) stesso (o argomenti affini) e non contiene nulla che possa essere compreso nell'argomento principale. (Per esempio, se il [documento](#) è in parte un manuale di matematica, una [sezione secondaria](#) non può contenere spiegazioni di matematica). Il rapporto con l'argomento può essere un tema collegato storicamente con il soggetto principale o con soggetti affini, o essere costituito da argomentazioni legali, commerciali, filosofiche, etiche o politiche pertinenti.

Le "[sezioni non modificabili](#)" sono alcune [sezioni secondarie](#) i cui titoli sono esplicitamente dichiarati essere [sezioni non modificabili](#), nella nota che indica che il [documento](#) è realizzato sotto questa licenza.

I "[testi copertina](#)" sono dei brevi brani di testo che sono elencati nella nota che indica che il documento è realizzato sotto questa licenza.

Una copia "[trasparente](#)" del [documento](#) indica una copia leggibile da un calcolatore, codificata in un formato le cui specifiche sono disponibili pubblicamente, i cui contenuti possono essere visti e modificati direttamente, ora e in futuro, con generici editor di testi o (per immagini composte da pixel) con generici editor di immagini o (per i disegni) con qualche editor di disegni ampiamente diffuso, e la copia deve essere adatta al trattamento per la formattazione o per la conversione in una varietà di formati atti alla successiva formattazione. Una copia fatta in un altro formato di file [trasparente](#) il cui markup è stato progettato per intralciare o scoraggiare modifiche future da parte dei lettori non è [trasparente](#). Una copia che non è trasparente è "opaca".

Esempi di formati adatti per copie [trasparenti](#) sono l'ASCII puro senza markup, il formato di input per Texinfo, il formato di input per LaTeX, SGML o XML accoppiati ad una DTD pubblica e disponibile, e semplice HTML conforme agli standard e progettato per essere modificato manualmente. Formati opachi sono PostScript, PDF, formati proprietari che possono essere letti e modificati solo con word processor proprietari, SGML o XML per cui non è in genere disponibile la DTD o gli strumenti per il trattamento, e HTML generato automaticamente da qualche word processor per il solo output.

La "[pagina del titolo](#)" di un libro stampato indica la pagina del titolo stessa, più qualche pagina seguente per quanto necessario a contenere in modo leggibile, il materiale che la licenza prevede che compaia nella pagina del titolo. Per opere in formati in cui non sia contemplata esplicitamente la pagina del titolo, con "[pagina del titolo](#)" si intende il testo prossimo al titolo dell'opera, precedente l'inizio del corpo del testo.

2. COPIE ALLA LETTERA

Si può copiare e distribuire il [documento](#) con l'ausilio di qualsiasi mezzo, per fini di lucro e non, fornendo per tutte le copie questa licenza, le note sul copyright e l'avviso che questa licenza si applica al [documento](#), e che non si aggiungono altre condizioni al di fuori di quelle della licenza stessa. Non si possono usare misure tecniche per impedire o controllare la lettura o la produzione di copie successive alle copie che si producono o distribuiscono. Però si possono ricavare compensi per le copie fornite. Se si distribuiscono un numero sufficiente di copie si devono seguire anche le condizioni della [sezione 3](#).

Si possono anche prestare copie e con le stesse condizioni sopra menzionate possono essere utilizzate in pubblico.

3. COPIARE IN NOTEVOLI QUANTITÀ

Se si pubblicano a mezzo stampa più di 100 copie del [documento](#), e la nota della licenza indica che esistono uno o più [testi copertina](#), si devono includere nelle copie, in modo chiaro e leggibile, tutti i [testi copertina](#) indicati: il testo della prima di copertina in prima di copertina e il testo di quarta di copertina in quarta di copertina. Ambedue devono identificare l'editore che pubblica il documento. La prima di copertina deve presentare il titolo completo con tutte le parole che lo compongono egualmente visibili ed evidenti. Si può aggiungere altro materiale alle copertine. Il copiare con modifiche limitate alle sole copertine, purché si preservino il titolo e le altre condizioni viste in precedenza, è considerato alla stregua di copiare alla lettera.

Se il testo richiesto per le copertine è troppo voluminoso per essere riprodotto in modo leggibile, se ne può mettere una prima parte per quanto ragionevolmente può stare in copertina, e continuare nelle pagine immediatamente seguenti.

Se si pubblicano o distribuiscono copie [opache](#) del [documento](#) in numero superiore a 100, si deve anche includere una copia [trasparente](#) leggibile da un calcolatore per ogni copia o menzionare per ogni copia [opaca](#) un indirizzo di una rete di calcolatori pubblicamente accessibile in cui vi sia una copia [trasparente](#) completa del [documento](#), spogliato di materiale aggiuntivo, e a cui si possa accedere anonimamente e gratuitamente per scaricare il documento usando i protocolli standard e pubblici generalmente usati. Se si adotta l'ultima opzione, si deve prestare la giusta attenzione, nel momento in cui si inizia la distribuzione in quantità elevata di copie [opache](#), ad assicurarsi che la copia [trasparente](#) rimanga accessibile all'indirizzo stabilito fino ad almeno un anno di distanza dall'ultima distribuzione (direttamente o attraverso rivenditori) di quell'edizione al pubblico.

È caldamente consigliato, benché non obbligatorio, contattare l'autore del [documento](#) prima di distribuirne un numero considerevole di copie, per metterlo in grado di fornire una versione aggiornata del [documento](#).

4. MODIFICHE

Si possono copiare e distribuire [versioni modificate](#) del [documento](#) rispettando le condizioni delle precedenti sezioni [2](#) e [3](#), purché la [versione modificata](#) sia realizzata seguendo scrupolosamente questa stessa licenza, con la [versione modificata](#) che svolga il ruolo del "[documento](#)", così da estendere la licenza sulla distribuzione e la modifica a chiunque ne possieda una copia. Inoltre nelle versioni modificate si deve:

- A. Usare nella [pagina del titolo](#) (e nelle copertine se ce ne sono) un titolo diverso da quello del [documento](#), e da quelli di versioni precedenti (che devono essere elencati nella sezione storia del [documento](#) ove presenti). Si può usare lo stesso titolo di una versione precedente se l'editore di quella versione originale ne ha dato il permesso.
- B. Elencare nella [pagina del titolo](#), come autori, una o più persone o gruppi responsabili in qualità di autori delle modifiche nella [versione modificata](#), insieme ad almeno cinque fra i principali autori del documento (tutti gli autori principali se sono meno di cinque).
- C. Dichiarare nella [pagina del titolo](#) il nome dell'editore della [versione modificata](#) in qualità di editore.
- D. Conservare tutte le note sul copyright del [documento originale](#).

Aggiungere un'appropriata licenza per le modifiche di seguito alle altre licenze sui copyright.

- F. Includere immediatamente dopo la nota di copyright, un avviso di licenza che dia pubblicamente il permesso di usare la [versione modificata](#) nei termini di questa licenza, nella forma mostrata nell'addendum alla fine di questo testo.
- G. Preservare in questo avviso di licenza l'intera lista di [sezioni non modificabili](#) e [testi copertina](#) richieste come previsto dalla licenza del [documento](#).
- H. Includere una copia non modificata di questa licenza.
- I. Conservare la sezione intitolata "Storia", e il suo titolo, e aggiungere a questa un elemento che riporti al minimo il titolo, l'anno, i nuovi autori, e gli editori della [versione modificata](#) come figurano nella [pagina del titolo](#). Se non ci sono sezioni intitolate "Storia" nel [documento](#), createne una che riporti il titolo, gli autori, gli editori del [documento](#) come figurano nella [pagina del titolo](#), quindi aggiungete un elemento che descriva la [versione modificata](#) come detto in precedenza.
- J. Conservare l'indirizzo in rete riportato nel [documento](#), se c'è, al fine del pubblico accesso ad una copia [trasparente](#), e possibilmente l'indirizzo in rete per le precedenti versioni su cui ci si è basati. Questi possono essere collocati nella sezione "Storia". Si può omettere un indirizzo di rete per un'opera pubblicata almeno quattro anni prima del [documento stesso](#), o se l'originario editore della versione cui ci si riferisce ne dà il permesso.
- K. In ogni sezione di "Ringraziamenti" o "Dediche", si conservino il titolo, il senso, il tono della sezione stessa.
- L. Si conservino inalterate le [sezioni non modificabili](#) del [documento](#), nei propri testi e nei propri titoli. I numeri della sezione o equivalenti non sono considerati parte del titolo della sezione.
- M. Si cancelli ogni sezione intitolata "Riconoscimenti". Solo questa sezione può non essere inclusa nella [versione modificata](#).
- N. Non si modifichi il titolo di sezioni esistenti come "miglioria" o per creare confusione con i titoli di [sezioni non modificabili](#).

Se la [versione modificata](#) comprende nuove sezioni di primaria importanza o appendici che ricadono in "[sezioni secondarie](#)", e non contengono materiale copiato dal documento, si ha facoltà di rendere non modificabili quante sezioni si voglia. Per fare ciò si aggiunga il loro titolo alla lista delle [sezioni immutabili](#) nella nota di copyright della [versione modificata](#). Questi titoli devono essere

diversi dai titoli di ogni altra sezione.

Si può aggiungere una sezione intitolata "Riconoscimenti", a patto che non contenga altro che le approvazioni alla [versione modificata](#) prodotte da vari soggetti—per esempio, affermazioni di revisione o che il testo è stato approvato da una organizzazione come la definizione normativa di uno standard.

Si può aggiungere un brano fino a cinque parole come [Testo Copertina](#), e un brano fino a 25 parole come [Testo di Retro Copertina](#), alla fine dell'elenco dei [Testi Copertina](#) nella [versione modificata](#). Solamente un brano del [Testo Copertina](#) e uno del [Testo di Retro Copertina](#) possono essere aggiunti (anche con adattamenti) da ciascuna persona o organizzazione. Se il [documento](#) include già un testo copertina per la stessa copertina, precedentemente aggiunto o adattato da voi o dalla stessa organizzazione nel nome della quale si agisce, non se ne può aggiungere un altro, ma si può rimpiazzare il vecchio ottenendo l'esplicita autorizzazione dall'editore precedente che aveva aggiunto il testo copertina.

L'autore/i e l'editore/i del ["documento"](#) non ottengono da questa licenza il permesso di usare i propri nomi per pubblicizzare la versione modificata o rivendicare l'approvazione di ogni [versione modificata](#).

5. UNIONE DI DOCUMENTI

Si può unire il [documento](#) con altri realizzati sotto questa licenza, seguendo i termini definiti nella precedente [sezione 4](#) per le versioni modificate, a patto che si includa l'insieme di tutte le [Sezioni Invarianti](#) di tutti i documenti originali, senza modifiche, e si elenchino tutte come [Sezioni Invarianti](#) della sintesi di documenti nella licenza della stessa.

Nella sintesi è necessaria una sola copia di questa licenza, e multiple [sezioni invarianti](#) possono essere rimpiazzate da una singola copia se identiche. Se ci sono multiple [Sezioni Invarianti](#) con lo stesso nome ma contenuti differenti, si renda unico il titolo di ciascuna sezione aggiungendovi alla fine e fra parentesi, il nome dell'autore o editore della sezione, se noti, o altrimenti un numero distintivo. Si facciano gli stessi aggiustamenti ai titoli delle sezioni nell'elenco delle [Sezioni Invarianti](#) nella nota di copyright della sintesi.

Nella sintesi si devono unire le varie sezioni intitolate "storia" nei vari documenti originali di partenza per formare una unica sezione intitolata "storia"; allo stesso modo si unisca ogni sezione intitolata "Ringraziamenti", e ogni sezione intitolata "Dediche". Si devono eliminare tutte le sezioni intitolate "Riconoscimenti".

6. RACCOLTE DI DOCUMENTI

Si può produrre una raccolta che consista del [documento](#) e di altri realizzati sotto questa licenza; e rimpiazzare le singole copie di questa licenza nei vari documenti con una sola inclusa nella raccolta, solamente se si seguono le regole fissate da questa licenza per le copie alla lettera come se si applicassero a ciascun documento.

Si può estrarre un singolo documento da una raccolta e distribuirlo individualmente sotto questa licenza, solo se si inserisce una copia di questa licenza nel documento estratto e se si seguono tutte le altre regole fissate da questa licenza per le copie alla lettera del documento.

7. RACCOGLIERE INSIEME A LAVORI INDIPENDENTI

Una raccolta del documento o sue derivazioni con altri [documenti](#) o lavori separati o indipendenti, all'interno di o a formare un archivio o un supporto per la distribuzione, non è una ["versione"](#)

[modificata](#) del [documento](#) nella sua interezza, se non ci sono copyright per l'intera raccolta. Ciascuna raccolta si chiama allora "aggregato" e questa licenza non si applica agli altri lavori contenuti in essa che ne sono parte, per il solo fatto di essere raccolti insieme, qualora non siano però loro stessi lavori derivati dal [documento](#).

Se le esigenze del [Testo Copertina](#) della [sezione 3](#) sono applicabili a queste copie del [documento](#) allora, se il [documento](#) è inferiore ad un quarto dell'intero aggregato i [Testi Copertina](#) del [documento](#) possono essere piazzati in copertine che delimitano solo il [documento](#) all'interno dell'aggregato. Altrimenti devono apparire nella copertina dell'intero aggregato.

8. TRANSLATION

La traduzione è considerata un tipo di modifica, e di conseguenza si possono distribuire traduzioni del [documento](#) seguendo i termini della [sezione 4](#). Rimpiazzare [sezioni non modificabili](#) con traduzioni richiede un particolare permesso da parte dei detentori del diritto d'autore, ma si possono includere traduzioni di una o più [sezioni non modificabili](#) in aggiunta alle versioni originali di queste sezioni immutabili. Si può fornire una traduzione della presente licenza a patto che si includa anche l'originale versione inglese di questa licenza. In caso di discordanza fra la traduzione e l'originale inglese di questa licenza la versione originale inglese prevale sempre.

9. TERMINI

Non si può applicare un'altra licenza al [documento](#), copiarlo, modificarlo, o distribuirlo al di fuori dei termini espressamente previsti da questa licenza. Ogni altro tentativo di applicare un'altra licenza al [documento](#), copiarlo, modificarlo, o distribuirlo è deprecato e pone fine automaticamente ai diritti previsti da questa licenza. Comunque, per quanti abbiano ricevuto copie o abbiano diritti coperti da questa licenza, essi non ne cessano se si rimane perfettamente coerenti con quanto previsto dalla stessa.

10. REVISIONI FUTURE DI QUESTA LICENZA

La [Free Software Foundation](#) può pubblicare nuove, rivedute versioni della Gnu Free Documentation License volta per volta. Qualche nuova versione potrebbe essere simile nello spirito alla versione attuale ma differire in dettagli per affrontare nuovi problemi e concetti. Si veda <http://www.gnu.org/copyleft/>.

Ad ogni versione della licenza viene dato un numero che distingue la versione stessa. Se il [documento](#) specifica che si riferisce ad una versione particolare della licenza contraddistinta dal numero o "ogni versione successiva", si ha la possibilità di seguire termini e condizioni sia della versione specificata che di ogni versione successiva pubblicata (non come bozza) dalla Free Software Foundation. Se il [documento](#) non specifica un numero di versione particolare di questa licenza, si può scegliere ogni versione pubblicata (non come bozza) dalla Free Software Foundation.

Come usare questa licenza per i vostri documenti

Per applicare questa licenza ad un documento che si è scritto, si includa una copia della licenza nel documento e si inserisca il seguente avviso di copyright appena dopo la pagina del titolo:

Copyright (c) ANNO VOSTRO NOME.

È garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le [Sezioni Non Modificabili](#) ELENCARNE I TITOLI, con i [Testi Copertina](#) ELENCO, e con i [Testi di Retro](#) Copertina ELENCO. Una copia della licenza è acclusa nella sezione intitolata "GNU Free Documentation License".

Guida Per Utonti

Se non ci sono [Sezioni non Modificabili](#), si scriva "senza Sezioni non Modificabili" invece di dire quali sono non modificabili. Se non c'è [Testo Copertina](#), si scriva "nessun Testo Copertina" invece di "il testo Copertina è ELENCO"; e allo stesso modo si operi per il [Testo di Retro Copertina](#).

Se il vostro documento contiene esempi non banali di programma in codice sorgente si raccomanda di realizzare gli esempi contemporaneamente applicandovi anche una licenza di software libero di vostra scelta, come ad esempio la [GNU General Public License](#), al fine di permetterne l'uso come software libero.

[Indietro](#)

[Partenza](#)

GNU Free Documentation License

Guida Per Utonti

[Avanti](#)

[Indietro](#)

Appendice B. GNU Free Documentation License

Version 1.1, March 2000

Copyright © 2000

Free Software Foundation, Inc.
59 Temple Place, Suite 330,
Boston,
MA
02111-1307
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "[Document](#)", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "[Modified Version](#)" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "[Secondary Section](#)" is a named appendix or a front-matter section of the [Document](#) that deals exclusively with the relationship of the publishers or authors of the [Document](#) to the [Document's](#) overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the [Document](#) is in part a textbook of mathematics, a [Secondary Section](#) may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "[Invariant Sections](#)" are certain [Secondary Sections](#) whose titles are designated, as being those of [Invariant Sections](#), in the notice that says that the [Document](#) is released under this License.

The "[Cover Texts](#)" are certain short passages of text that are listed, as [Front-Cover Texts](#) or [Back-Cover Texts](#), in the notice that says that the [Document](#) is released under this License.

A "[Transparent](#)" copy of the [Document](#) means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise [Transparent](#) file format whose markup has been designed to thwart or discourage subsequent modification by readers is not [Transparent](#). A copy that is not "[Transparent](#)" is called "Opaque".

Examples of suitable formats for [Transparent](#) copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "[Title Page](#)" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "[Title Page](#)" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

2. VERBATIM COPYING

You may copy and distribute the [Document](#) in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the [Document](#) are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in [section 3](#).

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies of the [Document](#) numbering more than 100, and the [Document's](#) license notice requires [Cover Texts](#), you must enclose the copies in covers that carry, clearly and legibly, all these [Cover Texts](#): [Front-Cover Texts](#) on the front cover, and [Back-Cover Texts](#) on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the [Document](#) and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute [Opaque](#) copies of the [Document](#) numbering more than 100, you must either include a machine-readable [Transparent](#) copy along with each [Opaque](#) copy, or state in or with each [Opaque](#) copy a publicly-accessible computer-network location containing a complete [Transparent](#) copy of the [Document](#), free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of [Opaque](#) copies in quantity, to ensure that this [Transparent](#) copy will remain thus accessible at the stated location until at least one year after the last time you distribute an [Opaque](#) copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the [Document](#) well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the [Document](#).

4. MODIFICATIONS

You may copy and distribute a [Modified Version](#) of the [Document](#) under the conditions of sections [2](#) and [3](#) above, provided that you release the [Modified Version](#) under precisely this License, with the [Modified Version](#) filling the role of the [Document](#), thus licensing distribution and modification of the [Modified Version](#) to whoever possesses a copy of it. In addition, you must do these things in the [Modified Version](#):

- A. Use in the [Title Page](#) (and on the covers, if any) a title distinct from that of the [Document](#), and from those of previous versions (which should, if there were any, be listed in the History section of the [Document](#)). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the [Title Page](#), as authors, one or more persons or entities responsible for authorship of the modifications in the [Modified Version](#), together with at least five of the principal authors of the [Document](#) (all of its principal authors, if it has less than five).
- C. State on the [Title Page](#) the name of the publisher of the [Modified Version](#), as the publisher.
- D. Preserve all the copyright notices of the [Document](#).
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the [Modified Version](#) under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of [Invariant Sections](#) and required [Cover Texts](#) given in the [Document's](#) license notice.
- H.

Include an unaltered copy of this License.

- I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the [Modified Version](#) as given on the [Title Page](#). If there is no section entitled "History" in the [Document](#), create one stating the title, year, authors, and publisher of the [Document](#) as given on its [Title Page](#), then add an item describing the [Modified Version](#) as stated in the previous sentence.
- J. Preserve the network location, if any, given in the [Document](#) for public access to a [Transparent](#) copy of the [Document](#), and likewise the network locations given in the [Document](#) for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the [Document](#) itself, or if the original publisher of the version it refers to gives permission.
- K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the [Invariant Sections](#) of the [Document](#), unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section entitled "Endorsements". Such a section may not be included in the [Modified Version](#).
- N. Do not retitle any existing section as "Endorsements" or to conflict in title with any [Invariant Section](#).

If the [Modified Version](#) includes new front-matter sections or appendices that qualify as [Secondary Sections](#) and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of [Invariant Sections](#) in the [Modified Version's](#) license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your [Modified Version](#) by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a [Front-Cover Text](#), and a passage of up to 25 words as a [Back-Cover Text](#), to the end of the list of [Cover Texts](#) in the [Modified Version](#). Only one passage of [Front-Cover Text](#) and one of [Back-Cover Text](#) may be added by (or through arrangements made by) any one entity. If the [Document](#) already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the [Document](#) do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any [Modified Version](#) .

5. COMBINING DOCUMENTS

You may combine the [Document](#) with other documents released under this License, under the terms defined in [section 4](#) above for modified versions, provided that you include in the combination all of the [Invariant Sections](#) of all of the original documents, unmodified, and list them all as [Invariant Sections](#) of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical [Invariant Sections](#) may be replaced with a single copy. If there are multiple [Invariant Sections](#) with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of [Invariant Sections](#) in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the [Document](#) and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the [Document](#) or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a [Modified Version](#) of the [Document](#), provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the [Document](#) , on account of their being thus compiled, if they are not themselves derivative works of the [Document](#). If the [Cover Text](#) requirement of [section 3](#) is applicable to these copies of the [Document](#), then if the [Document](#) is less than one quarter of the entire aggregate, the [Document's Cover Texts](#) may be placed on covers that surround only the [Document](#) within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the [Document](#) under the terms of [section 4](#). Replacing [Invariant Sections](#) with translations requires special permission from their copyright holders, but you may include translations of some or all [Invariant Sections](#) in addition to the original versions of these [Invariant Sections](#). You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the [Document](#) except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the [Document](#) is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The [Free Software Foundation](#) may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the [Document](#) specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the [Document](#) does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

Addendum

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright © YEAR YOUR NAME.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the [Invariant Sections](#) being LIST THEIR TITLES, with the [Front-Cover Texts](#) being LIST, and with the [Back-Cover Texts](#) being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no [Invariant Sections](#), write "with no Invariant Sections" instead of saying which ones are invariant. If you have no [Front-Cover Texts](#), write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for [Back-Cover Texts](#).

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the [GNU General Public License](#), to permit their use in free software.

[Indietro](#)

Gli autori di GPU

[Partenza](#)

[Avanti](#)

GNU Free Documentation License
(traduzione italiana)