

Breve introduzione a Linux

V1.4

Matteo Ianeselli - LinuxTrent

12 novembre 2002

Capitolo 2

Preparazione all'installazione

2.1 Introduzione ai filesystem e alle partizioni.

Una tipica installazione di Linux per l'utilizzo come desktop può richiedere all'incirca un gigabyte di spazio libero sul disco rigido¹: ciò prevedendo di installare il sistema di base con l'ambiente di sviluppo *e gli applicativi più comuni*, oltre che a riservarsi dello spazio di swap per la memoria virtuale e ad avere dello spazio libero per i propri file (nell'ordine dei 300 megabyte).

Per farsi un'idea, l'installazione di *tutti* i pacchetti software presenti nella distribuzione RedHat 6.2 occupa 1,2GB di spazio, ai quale va aggiunto lo spazio libero per gli utenti e per la gestione del sistema.

2.1.1 Cos'è un filesystem?

Sistemi operativi diversi hanno di solito modi completamente diversi di organizzare i dati su disco, e talvolta accade pure che lo stesso sistema operativo usi modi diversi di organizzare i dati a seconda della natura dei dati stessi e del loro impiego. Il modo con cui essi sono organizzati prende il nome di *filesystem*.

Esiste un grande numero di tipi di filesystem diversi, destinati a scopi diversi. Ad esempio, nel solo ambito Windows esistono:

- FAT12, che viene usato sui floppy disk e sui dischetti da $3\frac{1}{2}$;
- FAT16, che una volta veniva usato su dischi rigidi, con la classica limitazione di 8+3 caratteri nel nome;
- FAT16 con nomi lunghi, in cui il limite di 8+3 caratteri viene portato a 255 caratteri;
- FAT32, analoga alle precedenti, ma con il quale è possibile gestire spazi di grandezza superiore ai 2 gigabyte.
- ISO9660 + estensioni Joliet, usato sui CD-ROM;
- NTFS, il filesystem usato da Windows NT, che memorizza informazioni aggiuntive sui file, quali permessi d'accesso, proprietario, etc. e che comunque organizza i dati in maniera più efficiente rispetto ai precedenti e con meno rischi di perdita di dati.

Nondimeno, Linux (essendo un sistema operativo della famiglia degli UNIX) ha le proprie esigenze in termini di organizzazione dei dati su disco, ed il tipo di filesystem da esso usato è solitamente EXT2 (che sta per EXTended filesystem, version 2)².

Normalmente, se una porzione dello spazio su disco è gestita utilizzando un determinato tipo di filesystem, non è possibile gestirla *contemporaneamente* usando un altro tipo di filesystem (perché

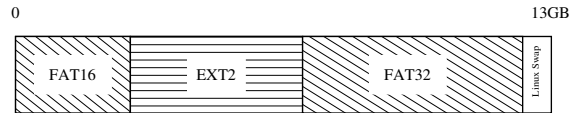
¹Le dimensioni sono puramente indicative, e variano radicalmente a seconda degli utilizzi: esistono distribuzioni di Linux che stanno su un singolo dischetto da $3\frac{1}{2}$ formattato in maniera particolare a 1.7 megabyte, come esistono distribuzioni che, con tutto installato, superano abbondantemente il gigabyte e mezzo, senza contare lo spazio di swap e lo spazio libero per i propri dati.

²*Solitamente* in quanto esistono diversi altri tipi di filesystem supportati da Linux che sono altrettanto adatti allo scopo, come ReiserFS.

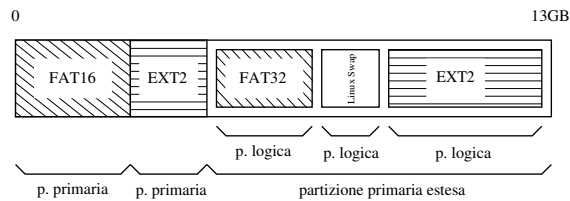
i modi di organizzare i dati sono diversi tra loro). Tuttavia, *porzioni diverse* dello spazio su disco possono essere gestite usando tipi di filesystem diversi tra loro. Ecco che qui entrano in gioco le partizioni di un disco rigido.

2.1.2 Cosa sono le partizioni (di un disco rigido)?

In breve, partizionare un disco rigido è l'operazione con cui si suddivide lo stesso in parti più piccole, dette appunto *partizioni*, che possono essere immaginate come tanti piccoli dischi rigidi separati. Ciò permette a sistemi operativi diversi (e che utilizzano criteri diversi per organizzare i dati sul disco) di convivere pacificamente, senza che un sistema operativo vada a sovrascrivere o a danneggiare i dati dell'altro sistema.



Originariamente, lo standard nel mondo dei PC IBM compatibili prevedeva che un disco rigido fosse suddivisibile in al massimo 4 partizioni (dette *primarie*). Ben presto, ciò si rivelò del tutto insufficiente, per cui venne reso possibile suddividere ulteriormente queste 4 partizioni in un numero arbitrario di sotto-partizioni (dette *logiche*). Una partizione primaria destinata a contenere partizioni logiche viene detta *partizione estesa*.



Il layout delle partizioni è conservato sul disco rigido stesso, in una piccola area chiamata *partition table*. In essa, oltre ad altre informazioni, viene conservata per ogni partizione il punto d'inizio, il punto di fine, e un identificatore che permette di stabilire velocemente quale filesystem contiene la partizione. L'identificatore è un byte (un numero da 0 a 255, o da 00 a FF in esadecimale). Ecco un elenco sintetico degli identificatori esistenti:

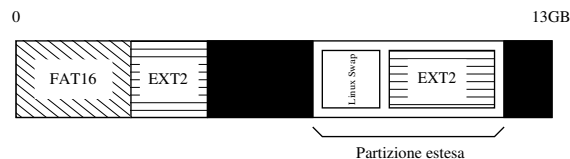
01 FAT12	61 SpeedStor
02 XENIX root	63 GNU HURD or SysV
03 XENIX usr	64 Novell Netware 286
04 FAT16 <32M	65 Novell Netware 386
05 Extended	70 DiskSecure Multi-Boo
06 FAT16	75 PC/IX
07 HPFS/NTFS	80 Old Minix
08 AIX	81 Minix / old Linux
09 AIX bootable	82 Linux swap
0A OS/2 Boot Manager	83 Linux
0B Win95 FAT32	84 OS/2 hidden C: drive
0C Win95 FAT32 (LBA)	85 Linux extended
0E Win95 FAT16 (LBA)	86 NTFS volume set
0F Win95 Ext'd (LBA)	87 NTFS volume set
10 OPUS	93 Amoeba
11 Hidden FAT12	94 Amoeba BBT
12 Compaq diagnostics	A0 IBM Thinkpad hiberna
14 Hidden FAT16 <32M	A5 BSD/386
16 Hidden FAT16	A6 OpenBSD
17 Hidden HPFS/NTFS	A7 NeXTSTEP
18 AST Windows swapfile	B7 BSDI fs
1B Hidden Win95 FAT32	B8 BSDI swap
1C Hidden Win95 FAT32 (C1 DRDOS/sec (FAT-12)
1E Hidden Win95 FAT16 (C4 DRDOS/sec (FAT-16 <

24 NEC DOS	C6 DRDOS/sec (FAT-16)
3C PartitionMagic recov	C7 Syrix
40 Venix 80286	DB CP/M / CTOS / ...
41 PPC PReP Boot	E1 DOS access
42 SFS	E3 DOS R/O
4D QNX4.x	E4 SpeedStor
4E QNX4.x 2nd part	EB BeOS fs
4F QNX4.x 3rd part	F1 SpeedStor
50 OnTrack DM	F4 SpeedStor
51 OnTrack DM6 Aux1	F2 DOS secondary
52 CP/M	FD Linux raid autodetec
53 OnTrack DM6 Aux3	FE LANstep
54 OnTrackDM6	FF BBT
55 EZ-Drive	
56 Golden Bow	
5C Priam Edisk	

Notare in particolare gli identificatori 05 e 85 che identificano le partizioni estese, ovvero le partizioni primarie destinate a contenere partizioni logiche. Linux, dal canto suo, fa uso dei tipi 82 Linux Swap e 83 Linux (EXT2).

Questo identificatore permette ai vari sistemi operativi di farsi velocemente un'idea del contenuto di una partizione, e quindi di stabilire da subito quali tra esse sono in grado di gestire e quali invece no. In questo modo, un sistema operativo diverso da Linux può facilmente ignorare le partizioni di tipo EXT2, evitando di presentarle erroneamente all'utente.

Diversamente da quello che si potrebbe credere, non è obbligatorio utilizzare tutto lo spazio su disco assegnandolo a una qualche partizione.



Lo spazio vuoto semplicemente *non viene utilizzato*. Ciò torna particolarmente utile quando si vogliono installare più sistemi operativi sullo stesso disco rigido e le utility di partizionamento degli stessi hanno funzionalità limitate (come nel caso di `fdisk` per Windows).

Il layout delle partizioni può essere variato in qualsiasi momento, tramite le utility descritte in seguito, tuttavia ciò generalmente comporta la perdita dei dati in esse contenuti (a meno che non venga usato un partizionatore *non distruttivo*, che vedremo in seguito). Cambiare il layout delle partizioni di fatto consiste semplicemente nel definire nuovi confini delle stesse, ma ciò non comporta automaticamente uno spostamento dei dati, o l'adeguamento alle nuove dimensioni, o la conversione da un tipo di filesystem a un altro. Ciò deve essere realizzato con utility specifiche.

Ad esempio, cambiando l'identificatore su una partizione da FAT32 a EXT2 non si convertono assolutamente i dati da un'organizzazione di tipo FAT32 a un'organizzazione di tipo EXT2, ma semplicemente *si sta suggerendo un'informazione sbagliata al sistema*, cosa che ha senso in previsione di formattare quella partizione con un tipo di filesystem diverso³.

La formattazione consiste semplicemente nel preparare un filesystem vuoto di un certo tipo all'interno di una certa partizione. Nel mondo Windows avviene tramite il comando `format` (che solitamente prepara un filesystem adeguato in base alla versione di Windows in uso), mentre nel mondo Linux ciò avviene con utility specifiche a seconda del tipo di filesystem desiderato (per EXT2 è `mke2fs`).

2.1.3 Utility per gestire le partizioni

Esistono diverse utility che permettono di gestire delle partizioni. Ecco alcune tra le più usate:

³Beninteso, una volta installato, Linux è in grado di interagire con i dati presenti su partizioni con numerosi filesystem. Nondimeno, le peculiarità dei sistemi UNIX richiedono informazioni aggiuntive sui file e sulle directory, e quindi solo una minoranza di tutti i filesystem supportati sono in grado di contenere il sistema, mentre gli altri sono utilizzabili ad esempio per lo scambio di dati.

- **fdisk** (di Windows): è un programma testuale che permette di creare, cancellare e modificare il layout delle partizioni. Non è di grande utilità in questo contesto, in quanto permette di gestire unicamente partizioni di tipo Windows, e al massimo una partizione estesa (benchè Windows stesso non sembri avere problemi nell'usare partizioni logiche contenute in più partizioni estese). Viene fornito a corredo di tutte le versioni di Windows.
- **fdisk** (di Linux): è un programma analogo a quello esistente per Windows, ma funzionante sotto Linux e in grado di gestire in maniera trasparente tutti i tipi di partizione elencati in precedenza (l'elenco è stato appunto ricavato da fdisk per Linux). Viene fornito a corredo di tutte le distribuzioni di Linux, e solitamente è disponibile in fase di installazione.
- **cdisk**: è analogo all'fdisk di Linux, solo che si presenta con un'interfaccia a caratteri usabile con i tasti cursore. Viene fornito a corredo con le varie distribuzioni di Linux. Solitamente non è disponibile in fase di installazione;
- **Disk Druid**: altra variante dell'fdisk di Linux, che si presenta con un'interfaccia a caratteri usabile con i tasti cursore e i tasti funzione, ma con alcune funzionalità in meno. Viene solitamente impiegato dal programma di installazione della distribuzione RedHat come alternativa al solito fdisk;
- **fips**: è un programma che permette di ridimensionare una partizione contenente un filesystem di tipo FAT16 o FAT32, senza distruggere dati in essa contenuti. Viene generalmente usato per rimpicciolire una partizione FAT16 (o 32) esistente, in modo da fare posto ad altre partizioni (quelle destinate a Linux). Necessita che la partizione in questione sia stata deframmentata (con defrag o simili), in quanto fips esige che l'area da liberare sia già sgombra dai dati;
- **GNU PartEd**: è un programma testuale che permette di definire delle partizioni, ma anche di ridimensionarle senza perdere i dati in esse contenuti. È disponibile anche sotto forma di micro-distribuzione di Linux in un solo dischetto da $3\frac{1}{2}$, e quindi non necessita che vi sia già installato un sistema operativo per funzionare;
- **Partition Magic** (per Windows): è un programma proprietario che permette di gestire partizioni di vario genere (anche quelle di tipo EXT2 usate da Linux), e in più consente di ridimensionare e spostare tali partizioni senza perdere i dati in esse contenuti (se tutto va bene). A differenza di fips, non necessita che la partizione sia stata deframmentata, ed è in grado inoltre di ridimensionare anche partizioni EXT2.

2.2 Preparare lo spazio su disco

Da quanto detto in precedenza dovrebbe essere ormai chiaro che Linux ha bisogno del proprio spazio su disco, e precisamente di *almeno* due partizioni (tre, se si presenta il problema dei 1024 cilindri descritto nella sezione che segue).

- Una prima partizione, da marcare con l'identificatore 83, è destinata a **contenere il sistema stesso, il kernel e l'area di lavoro per gli utenti**. Dato che inizialmente non si può prevedere con esattezza di quali applicativi si farà uso e di quanto spazio avremo bisogno (stiamo usando un nuovo sistema operativo, dopotutto), è ragionevole destinare a questa partizione uno spazio di qualche gigabyte.
- Una seconda partizione, da marcare con l'identificatore 82, è invece destinata a essere usata **per la memoria virtuale**, ovvero come area di scambio in cui conservare temporaneamente le informazioni che eccedono la quantità di memoria fisica presente sul PC. Tale partizione viene detta **partizione di swap**. Anche in questo caso le dimensioni di tale area sono fortemente legate al tipo di applicazioni che si vuole usare. Una regola di massima consiste nel riservare tanto spazio quanto il doppio della memoria fisica presente sul PC⁴.

⁴Ad esempio, con un vecchio PC con soli 32MB di memoria è il caso di preparare una partizione di swap da 64MB. Analogamente, con 64MB di memoria, si consiglia una partizione di swap da 128MB. Con quantità di memoria superiori si consiglia invece di riservare uno spazio di swap *pari* alla quantità di memoria fisica, per cui con 128MB di memoria va ancora bene riservare 128MB di swap. In ogni caso, si possono preparare in seguito fino a 16 partizioni di swap residenti anche su altri dischi rigidi, o si possono usare anche file di swap, per cui non è il caso di preoccuparsi più di tanto all'inizio.

Se abbiamo un disco rigido “vergine”, potremo fare questa cosa in tutta comodità direttamente in fase di installazione. Le varie distribuzioni di Linux permettono di utilizzare del software di partizionamento in questa fase.

Se abbiamo un disco rigido su cui già risiede un sistema operativo e non abbiamo spazio libero per creare queste due partizioni, dovremo provvedere prima a crearlo, ridimensionando le partizioni esistenti, oppure rinunciando ad una o più partizioni esistenti sufficientemente grandi, al posto della quale andremo invece a creare le nostre due partizioni per Linux.

Notare che se stiamo già usando tutte e 4 le partizioni primarie senza che nessuna di esse sia una partizione estesa, dobbiamo per forza cederne almeno una, al fine di creare una partizione estesa in cui potremo poi creare tutte le partizioni logiche di cui abbiamo bisogno. A differenza dei sistemi Windows, che devono necessariamente essere installati in una partizione primaria, Linux può risiedere anche in partizioni estese senza bisogno di alcun accorgimento particolare.

2.2.1 Attenzione ai 1024 cilindri!

Può esserci un problema relativo al caricamento iniziale di un qualunque sistema operativo su PC. Normalmente, subito dopo l'accensione della macchina, il BIOS carica da disco rigido un programma di qualche kilobyte che si occupa a sua volta di caricare il kernel (il nucleo) del sistema operativo e di mandarlo in esecuzione. Per fare questo, il programma in questione (detto **boot-loader**) utilizza le funzionalità messe a disposizione dal BIOS per leggere i dati che costituiscono il nucleo del sistema operativo. Il problema nasce in quanto vi sono dei limiti sull'area del disco a cui si può accedere tramite le funzionalità del BIOS. Ciò non costituisce un problema in seguito, in quanto è poi il sistema operativo a gestire direttamente gli accessi al disco, senza avvalersi del BIOS, tuttavia pone un confine ben preciso in fase di caricamento iniziale.

Questo problema solitamente non si avverte quando sulla macchina è installato un solo sistema operativo, in quanto avviene quasi sempre che la partizione che contiene il kernel sia situata interamente entro questo limite. Tuttavia, quando andiamo a installare Linux insieme a un altro sistema operativo, dobbiamo tenere conto di questo fatto, poichè pone qualche vincolo su come è possibile organizzare le partizioni.

BIOS non proprio recenti pongono questo limite a 20GB, o addirittura intorno agli 8GB del disco rigido. BIOS ancora più vecchi hanno questo limite intorno ai 4GB. In ogni caso, si tratta di un limite fissato ai primi 1024 cilindri del disco⁵. È importante che la partizione destinata a contenere il file con il kernel di Linux sia *interamente* al di sotto di questo limite.

Va anche detto che i BIOS recenti hanno delle estensioni che consentono di superare il limite dei 1024 cilindri, e le distribuzioni recenti se ne possono avvantaggiare.

Una soluzione che **funziona in tutti i casi** e che non comporta alcun problema dal punto di vista di Linux consiste nel riservare lo spazio per una piccola partizione (10-15 megabyte dovrebbero essere più che sufficienti per qualsiasi esigenza) destinata unicamente a contenere il file relativo al kernel.

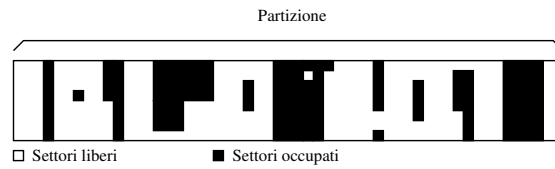
Se non riuscissimo in alcun modo a recuperare dello spazio prima del cilindro 1024, è pur sempre possibile avviare Linux con un dischetto da $3\frac{1}{2}$, o da Windows tramite il programma `loadlin`.

2.2.2 Usare fips per recuperare lo spazio che ci serve

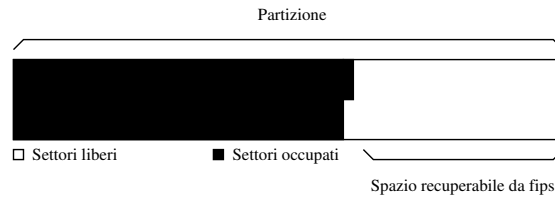
`fips` è un programma disponibile sotto forma di software libero per MS-DOS (e Windows in modalità solo MS-DOS) che permette di ridimensionare una partizione contenente un filesystem di tipo FAT16 o FAT32. Tale partizione va innanzitutto deframmentata utilizzando l'apposita utility `defrag` di Windows, in modo da compattare tutti i dati all'inizio o alla fine della partizione. Ciò è importante in quanto `fips` è in grado di recuperare lo spazio unicamente se su di esso non sono presenti dati.

È importante che anche eventuali file nascosti o di sistema vengano compattati all'inizio o alla fine della partizione, altrimenti non si riuscirebbe a recuperare lo spazio.

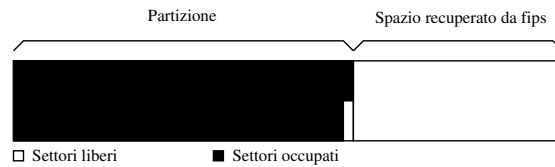
⁵Il metodo tradizionale nei PC IBM compatibili di indicare un punto preciso su un disco consiste nel fornire una terna di valori che indicano rispettivamente il cilindro, la testina e il settore. Questa terna ormai non ha alcuna attinenza con la realtà fisica del disco rigido, ma viene ancora usata per motivi di compatibilità. Iniziano comunque a prendere piede delle estensioni alle funzionalità dei BIOS moderni che permetteranno di dimenticarsi una volta per sempre di questo problema.



Come si può vedere nella figura, vi è molto spazio libero nella partizione, tuttavia i dati vanno spostati all'inizio o alla fine, in modo che poi fips sia in grado di recuperare lo spazio.



Dopo aver eseguito `fips` e aver risposto alle sue domande, ci ritroviamo con la partizione originale più piccola e una nuova partizione che contiene lo spazio vuoto.



Avviando in seguito `fdisk`, basterà eliminare la nuova partizione contenente vuota e creare nello spazio liberato le partizioni che ci servono.

Capitolo 3

Installazione

A questo punto abbiamo lo spazio libero su disco pronto per installare Linux. Creeremo le partizioni necessarie durante l'installazione.

3.1 Avviare il programma di installazione

La prima cosa da fare consiste nell'avviare il programma di installazione. Le moderne distribuzioni di Linux possono essere avviate direttamente da CDROM. Per fare questo è sufficiente avviare il programma di controllo del BIOS del PC, solitamente premendo il tasto Canc (o Del) sulla tastiera immediatamente dopo l'accensione del computer. Tra le (tante) opzioni disponibili vi è quella che permette di impostare l'ordine con cui il BIOS va alla ricerca dei dispositivi da cui effettuare l'avvio della macchina. È quindi sufficiente selezionare il lettore CDROM, salvare le impostazioni e riavviare la macchina con il CDROM inserito.

3.1.1 Se il computer non permette l'avvio da CDROM?

È possibile avviare Linux anche utilizzando un comune dischetto da $3\frac{1}{2}$, che solitamente viene incluso nelle distribuzioni commerciali, o che comunque è presente sul CDROM sotto forma di un file contenente una copia byte per byte del dischetto in questione. Sul CDROM è presente anche l'utility per MS-DOS chiamata RAWRITE.EXE, che permette di ricreare tale dischetto a partire dal file.

3.2 Uso di fdisk per partizionare i dischi

Una delle prime fasi dell'installazione di Linux consiste appunto nella definizione delle partizioni e nella loro formattazione, in modo che esse possano poi andare a ospitare il sistema. Spesso i programmi di installazione offrono come alternativa al loro software di partizionamento la possibilità di eseguire fdisk di Linux. L'interfaccia da esso offerta è testuale, ma non è affatto difficile da usare. fdisk per Linux si presenta inizialmente in questo modo:

```
Command (m for help):
```

Digitando m seguito da Invio come suggerito si ottiene un aiuto sui comandi disponibili:

```
Command (m for help): m
```

```
Command action
```

- a toggle a bootable flag
- b edit bsd disklabel
- c toggle the dos compatibility flag
- d delete a partition
- l list known partition types
- m print this menu
- n add a new partition

```

o  create a new empty DOS partition table
p  print the partition table
q  quit without saving changes
s  create a new empty Sun disklabel
t  change a partition's system id
u  change display/entry units
v  verify the partition table
w  write table to disk and exit
x  extra functionality (experts only)

```

Command (m for help):

Vediamo in dettaglio i comandi principali (per i rimanenti, non necessari al nostro scopo, si rimanda alla documentazione di fdisk):

- **a** permette di stabilire da quale partizione va effettuato il caricamento del sistema operativo subito dopo l'accensione della macchina;
- **d** permette di cancellare la definizione di una partizione;
- **l** visualizza un elenco degli identificatori noti (è lo stesso elenco di pagina 10);
- **m** visualizza la schermata d'aiuto;
- **n** permette di creare una nuova partizione;
- **p** visualizza il layout corrente delle partizioni;
- **q** esce senza attuare nessuna delle modifiche;
- **t** permette di modificare l'identificatore associato ad una partizione;
- **v** verifica la consistenza della tabella delle partizioni;
- **w** scrive le modifiche ed esce.

Come si può dedurre da quanto sopra, le modifiche non vengono riportate su disco fin quando non si utilizza il comando **w**.

3.3 Assegnazione delle partizioni

Come vedremo più in dettaglio nel prossimo capitolo, in Linux i vari dispositivi (lettore di dischetti, CDROM, partizioni) sono tutti inglobati in un unico albero di directory. Ciò significa che il contenuto di un partizione va agganciato in qualche punto dell'albero delle directory. In particolar modo, una partizione sarà destinata a contenere la radice di tutto l'albero delle directory, indicata semplicemente con una barra obliqua (/).¹

In fase di installazione dobbiamo specificare a quale directory deve corrispondere ogni partizione che abbiamo creato.

In Linux, una partizione viene identificata con le regole che seguono:

- interi dischi rigidi EIDE sono rappresentati tramite i nomi `/dev/hda` (master sul controller primario), `/dev/hdb` (slave sul controller primario), `/dev/hdc` (master sul controller secondario), `/dev/hdd` (slave sul controller secondario), etc.
- aggiungendo al nome del disco rigido un numero da 1 a 4 si indica una delle quattro partizioni primarie. Ad esempio, `/dev/hda2` indica la seconda partizione primaria sul disco rigido che è master sul primo controller EIDE.

¹A differenza del mondo Windows e MS-DOS, il carattere usato per separare i nomi delle directory nei sistemi UNIX è una barra obliqua (/) e non una barra obliqua rovescia (\).

- aggiungendo al nome del disco rigido un numero da 5 a 63, si indica una partizione logica. `/dev/hda5` indica la prima partizione logica, `/dev/hda6` la seconda, etc.
- interi dischi rigidi SCSI (o anche dispositivi rimovibili SCSI) sono rappresentati tramite i nomi `/dev/sda`, `/dev/sdb`, `/dev/sdc`, etc. La corrispondenza tra nomi e dispositivi è data dall'ordine con cui il sistema rileva i dispositivi stessi.
- aggiungendo ad essi un numero da 1 a 4 si indicano le partizioni primarie. Ad esempio, `/dev/sdb1` indica la prima partizione primaria sul secondo dispositivo SCSI;
- Aggiungendo ad essi un numero da 5 a 15 si indicano le partizioni logiche, in maniera analoga a quanto avviene per i dischi EIDE, solo che il numero è più limitato. Ad esempio: `/dev/sdb8`.

Le partizioni che necessitano di essere assegnate sono solo quelle che andranno a contenere file e directory. La partizione di swap (o le partizioni di swap) non sono quindi interessate.

Generalmente, si procederà ad assegnare la partizione più grande a `/` (la radice del filesystem), Se abbiamo preparato anche una piccola partizione per il problema dei 1024 cilindri, la assegneremo alla directory `/boot`.

Il programma di installazione provvederà a generare dei file di configurazione opportuni in modo che subito dopo l'avvio di Linux vengano ripristinati tali agganci (vedi anche 4.2.6.1 a pagina 42).

