

Lezione 10

File system

Sistemi Operativi (9 CFU), CdL Informatica, A. A. 2023/2024
Dipartimento di Scienze Fisiche, Informatiche e Matematiche
Università di Modena e Reggio Emilia

Quote of the day

(Meditate, gente, meditate...)

“Theory and practice sometimes clash. And when that happens, theory loses. Every single time.”

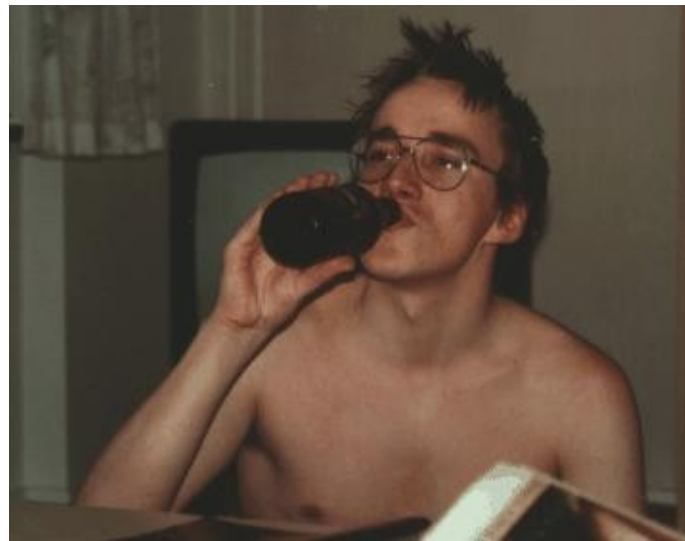
Linus Torvalds (1969-)

Programmatore

Autore di Linux

Autore di GIT

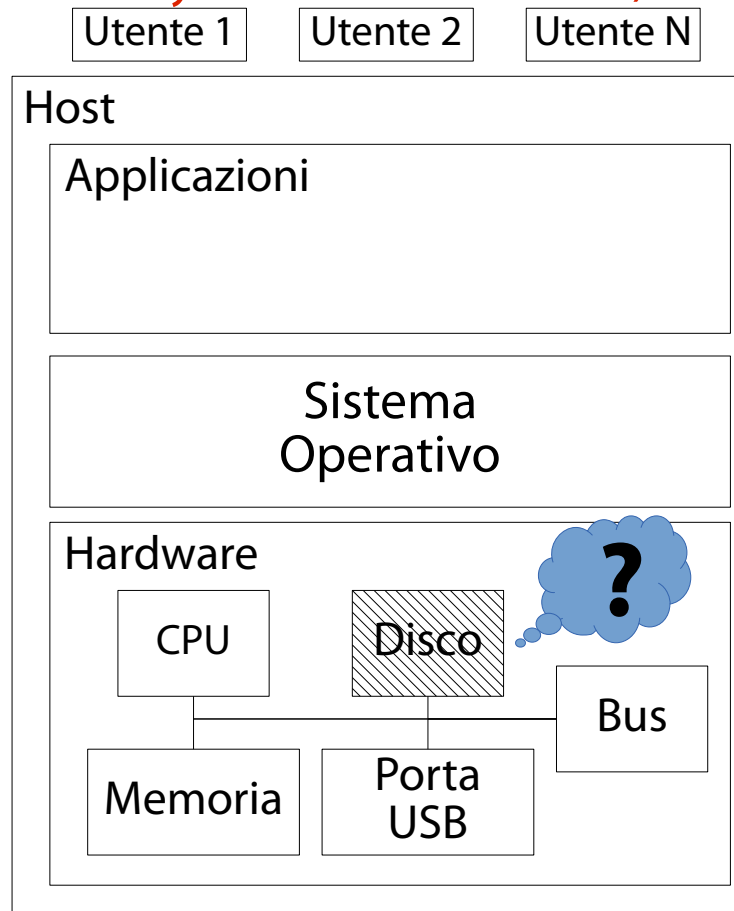
Autore di Subsurface



Lo scenario

(Lo studente vuole capire come preparare file system su dischi & Co.)

Uno studente in grado di usare la linea di comando vuole capire come preparare file system sui dispositivi di memorizzazione di massa.



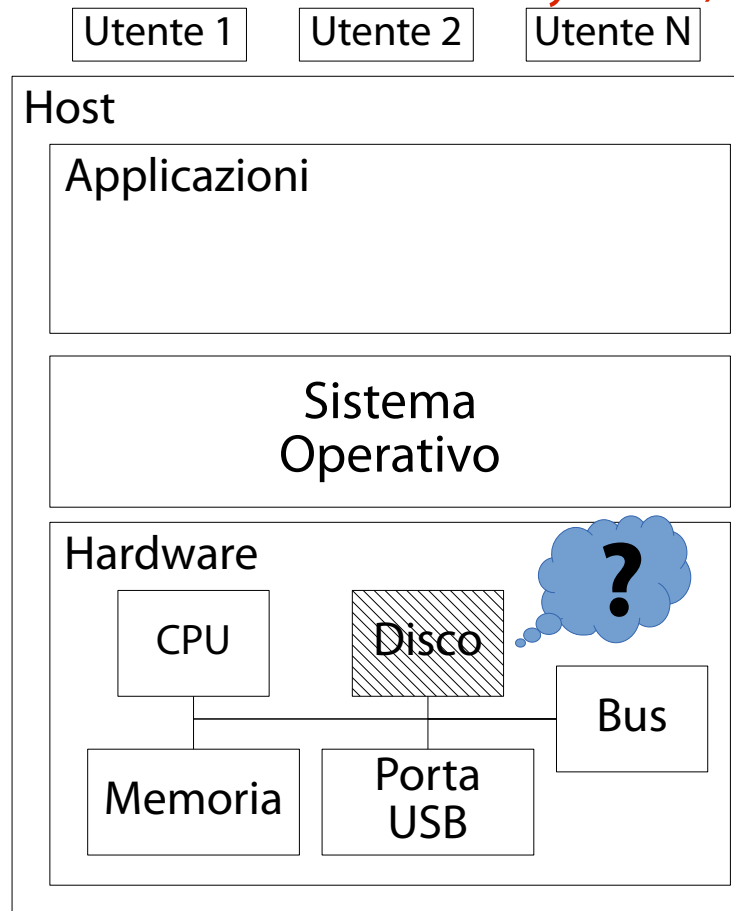
Interrogativi

(Come si prepara un dispositivo? Come si crea e si controlla un file system?)

Un dispositivo di memorizzazione di massa necessita di preparativi preliminari o può essere usato immediatamente?

Come viene creato un file system su un dispositivo?

È possibile controllare un file system?



INTRODUZIONE

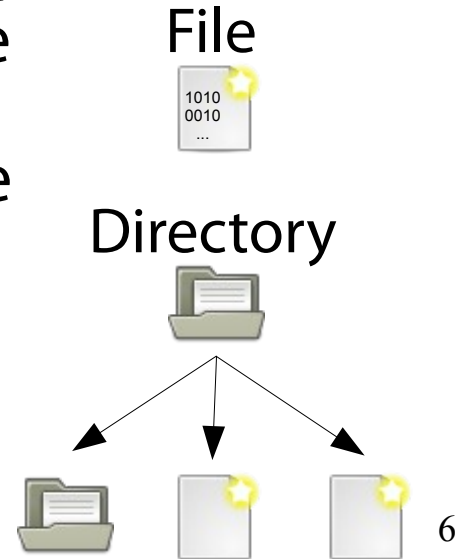
File e directory

(Repetita iuvant)

Nei SO moderni, il paradigma di uso di un dispositivo di memorizzazione ricorda la scrivania di un ufficio.

Fascicolo (file): contenitore di byte arbitrari, a cui un utente attribuisce un senso tramite le applicazioni.

Cartella (directory): raccoglitore di file e directory.



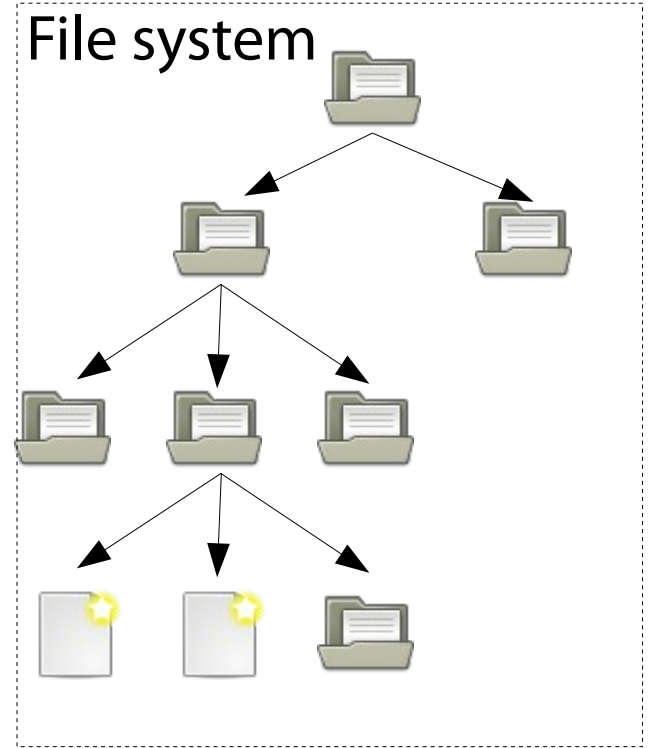
File system

(Contenitore di file e directory)

Un **file system** è una gerarchia di directory e file, ospitata su un dispositivo di memorizzazione.

Tipicamente, un disco rigido.

Rappresentazione: **albero** o **grafo orientato**.



Inizializzazione del file system

(Propedeutica all'organizzazione dei dati)

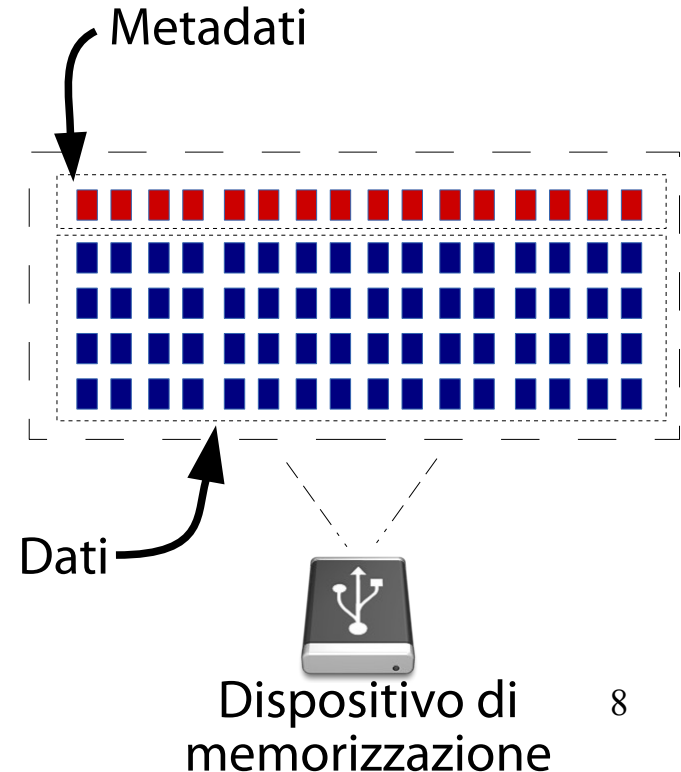
Inizialmente, un dispositivo è un insieme di blocchi dati, ciascuno di dimensione fissa.

Su tale insieme va organizzato un file system.

Formattazione: creazione di un file system vuoto su un dispositivo.

Area metadati (rosso): riservata alle informazioni sul file system.

Area dati (blu): riservata al contenuto dei file.

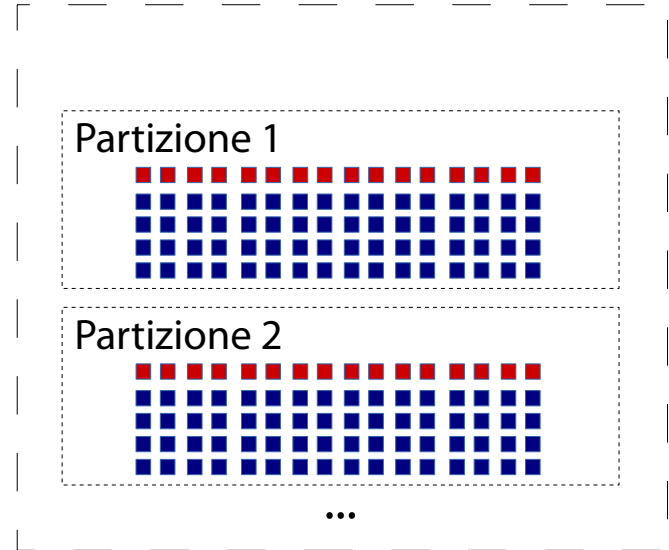


Disposizioni del file system

(All'interno di una singola partizione)

Alcuni dispositivi (i dischi rigidi) sono suddivisibili in porzioni fisse dette **partizioni**.

È possibile installare un file system diverso in ciascuna partizione.



Dispositivo di
memorizzazione

Disposizioni del file system

(Su partizioni distinte)

Alcuni dispositivi (i dischi rigidi) sono suddivisibili in porzioni fisse dette **partizioni**.

È possibile installare un file system su più partizioni distinte.

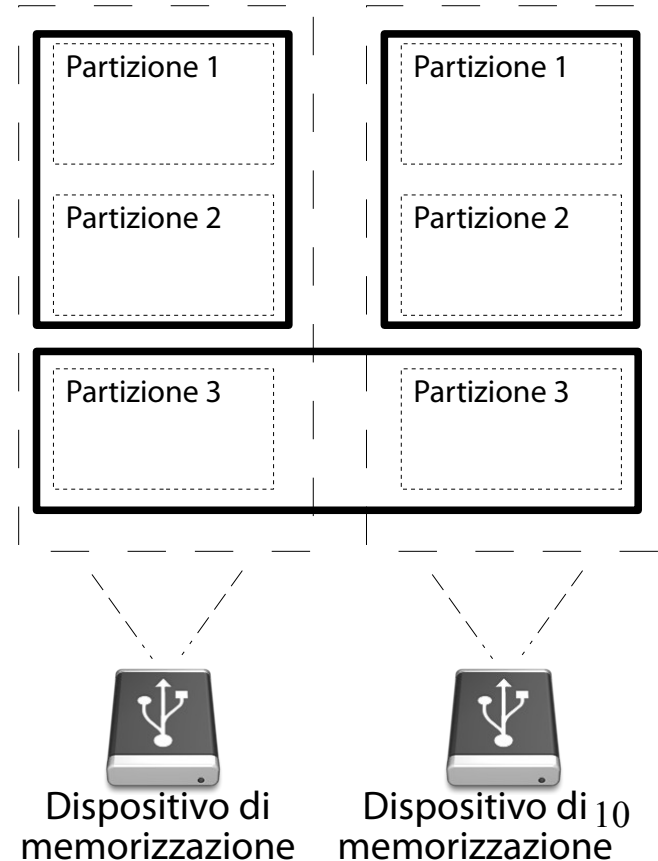
Sullo stesso dispositivo.

Su dispositivi diversi.

Esempi:

RAID.

Volumi logici (LVM).



Una avvertenza

(Altrimenti, si rischia di aprire il vaso di Pandora)

In questo corso introduttivo ci si occuperà della disposizione più semplice.

Un file system → al più una partizione.

Area metadati

(Che cosa contiene, in generale?)

L'area metadati contiene la struttura del file system.

Grandezza.

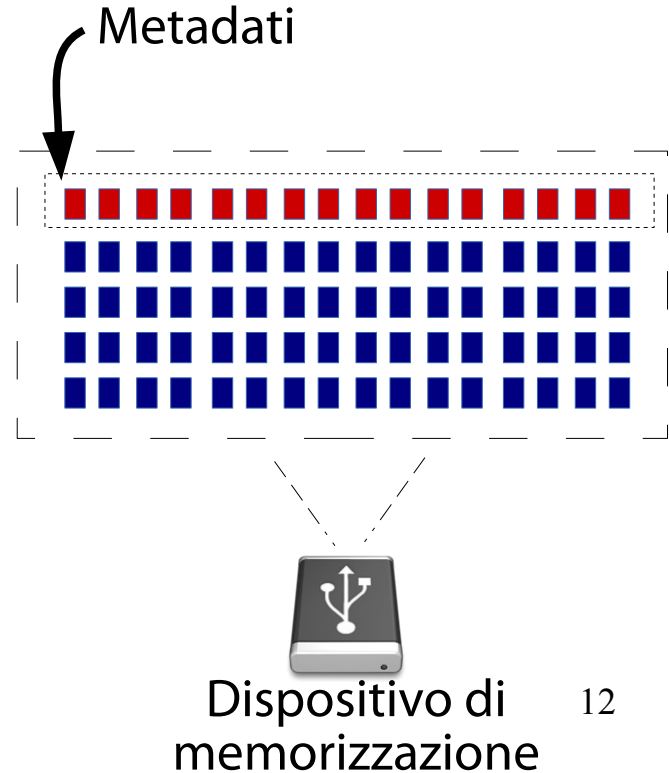
Tabella dei blocchi in uso.

Tabella dei blocchi liberi.

Strutture descrittive dei file.

Strutture descrittive dei directory.

Ciascun file system organizza in modo indipendente la propria struttura.



AGGIUNTA DI UN DISCO

Scenario

(Come si crea un disco rigido virtuale?)

Scenario: l'utente ha appena comprato un nuovo dispositivo di memorizzazione di massa (ad es., un disco rigido) e vuole aggiungerlo al computer.

Interrogativi:

Come si crea un disco rigido virtuale?

Creazione di un disco rigido “virtuale”

(Per chi non avesse sottomano un disco rigido non formattato...)

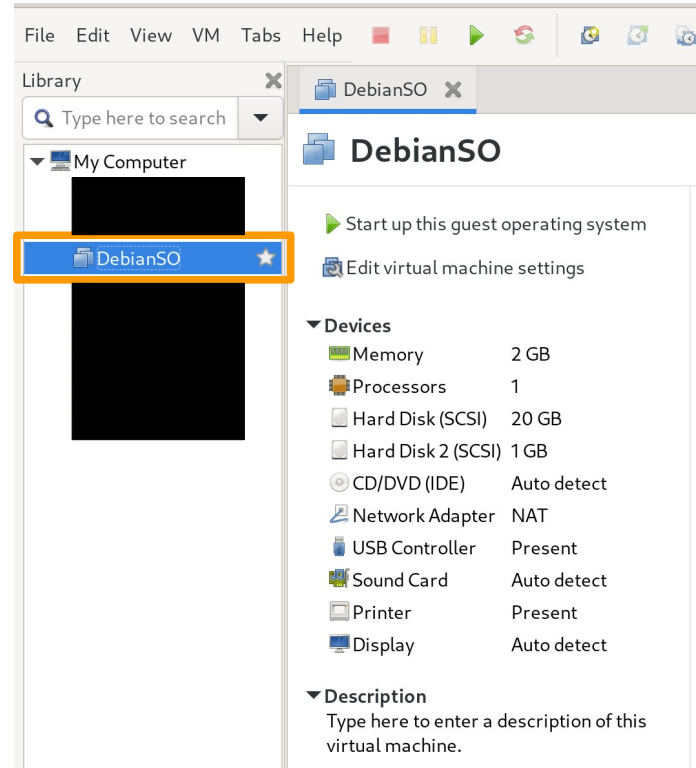
Per il resto della lezione si rende necessario l’inserimento di un disco rigido in un PC.

Non è necessario dotarsi di hardware vero (PC e disco). È possibile (e consigliato) lavorare con una macchina virtuale, su cui sarà inserito un secondo disco rigido virtuale.

Selezione del sistema guest

(A cui si vuole aggiungere un secondo disco rigido)

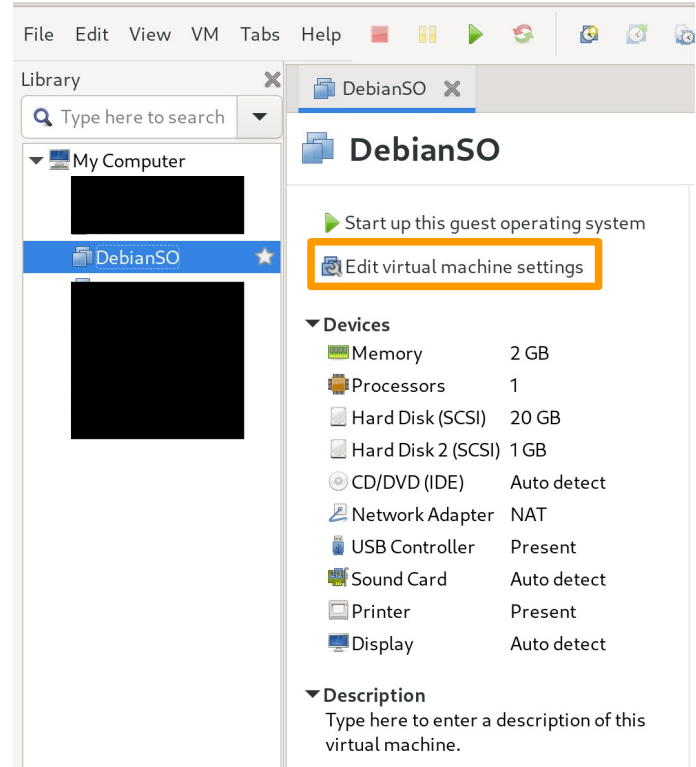
Partendo dall'elenco delle macchine virtuali, si selezioni il sistema guest a cui si vuole inserire un secondo disco rigido.



Selezione delle impostazioni

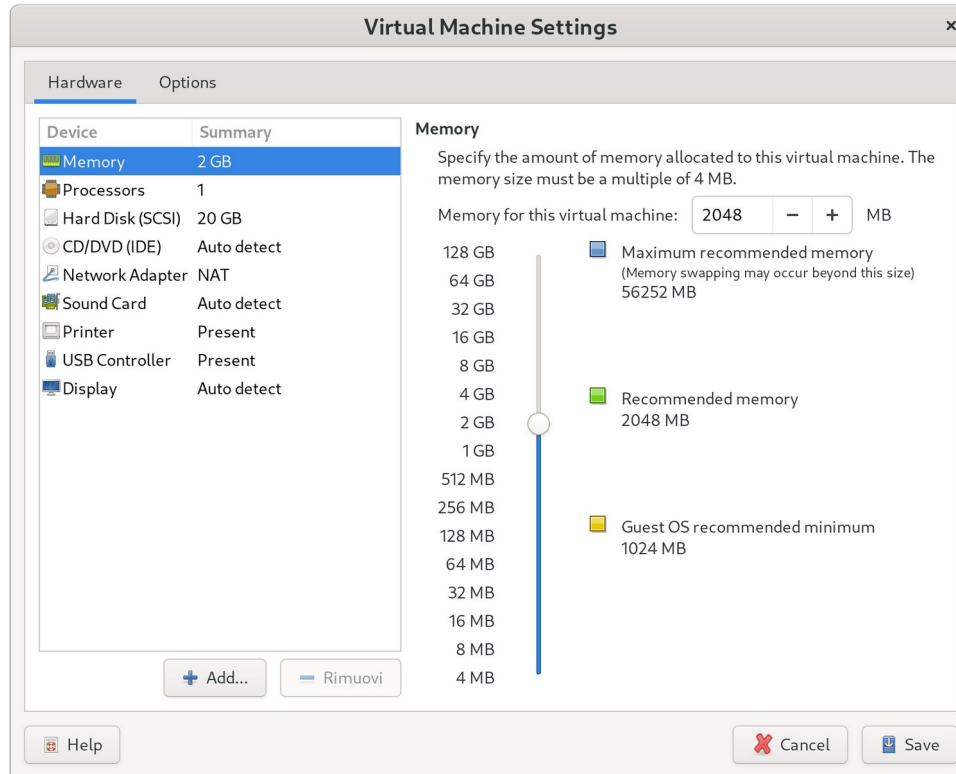
(Per poter aggiungere un disco virtuale)

Si selezioni il menu “Edit virtual machine settings”.



La finestra delle impostazioni

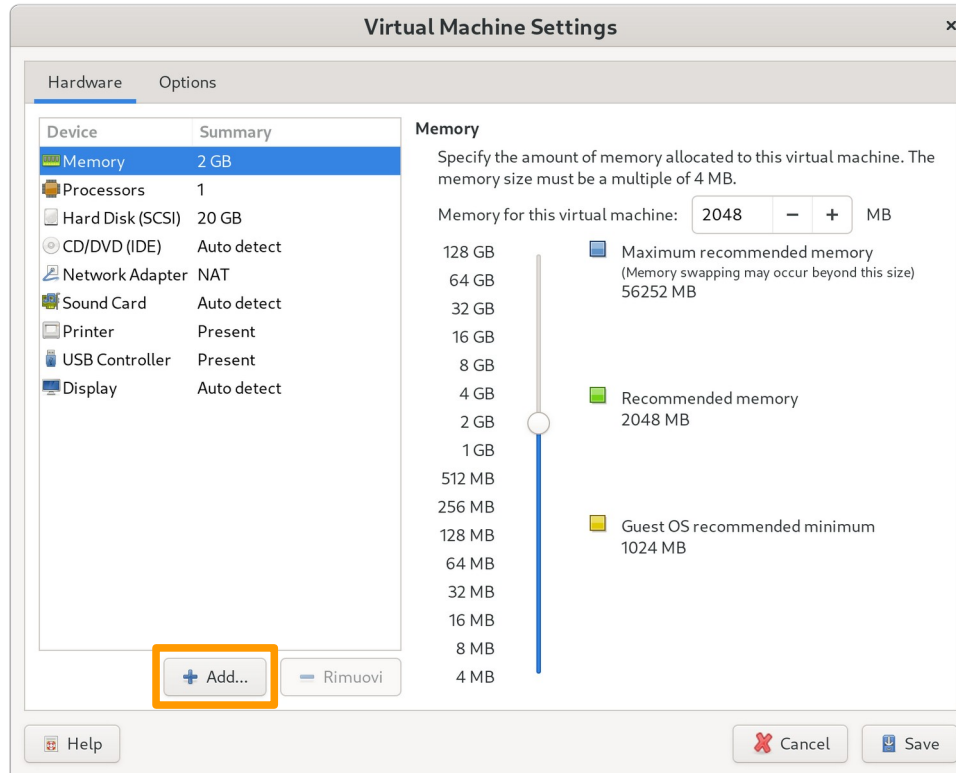
(Dovrebbe comparire)



Aggiunta di una periferica

(Virtuale)

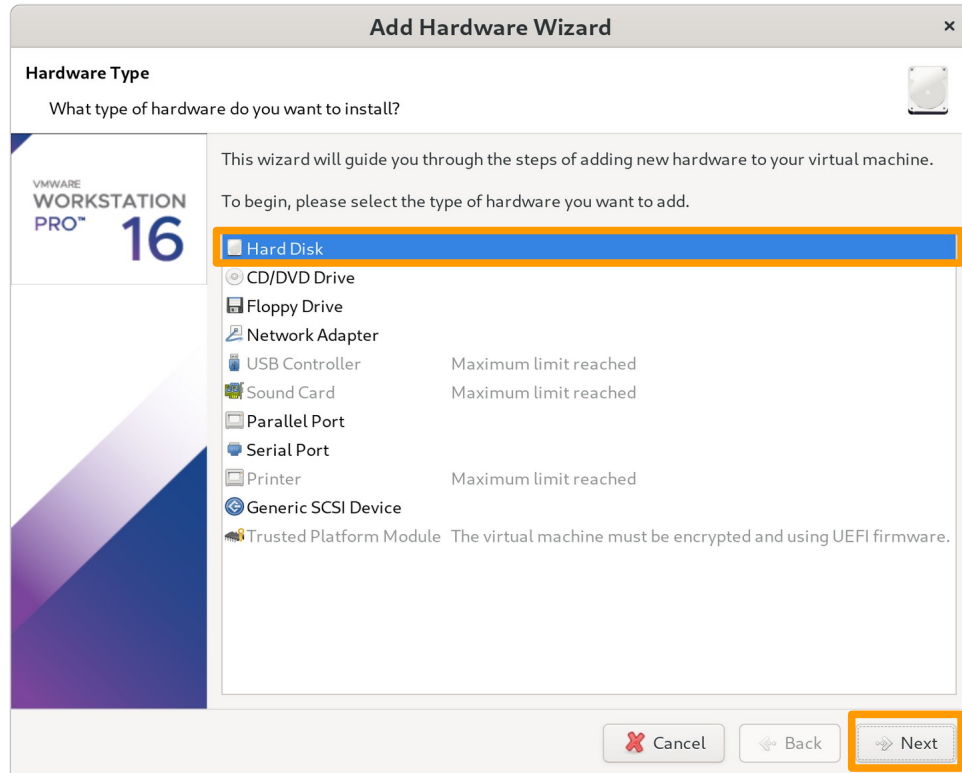
Si preme il pulsante “Add...” per aggiungere una periferica.



Aggiunta di un disco rigido virtuale

(Tramite apposito wizard)

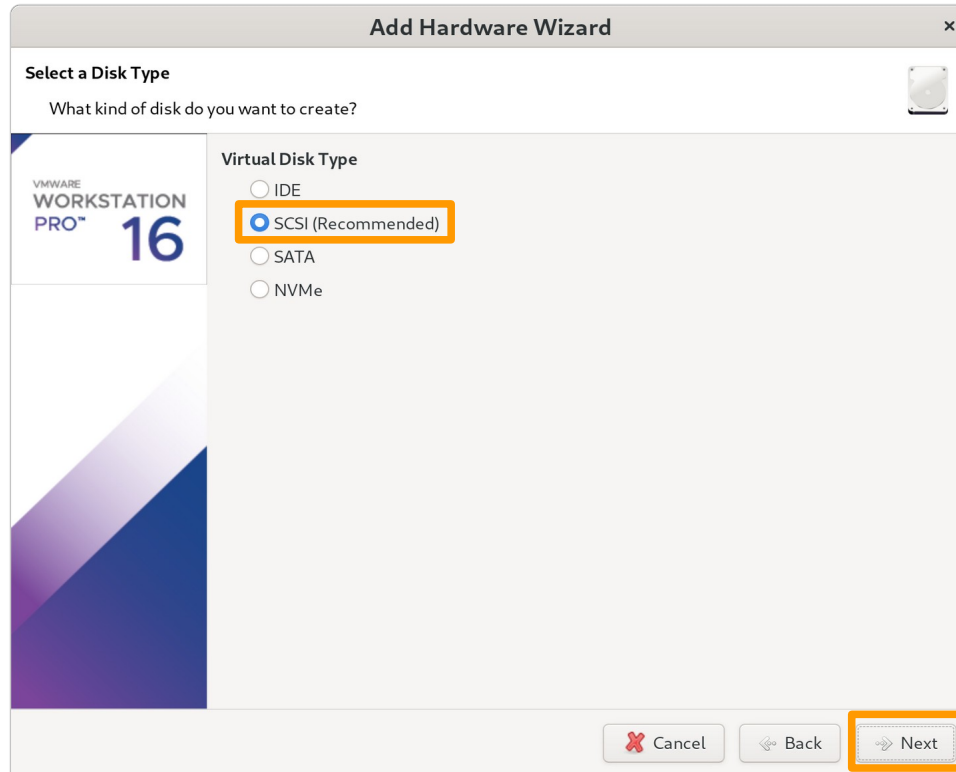
Si seleziona la periferica “Hard Disk” e si preme il pulsante “Next”.



Scelta tipologia del disco

(SCSI, SATA, IDE, NVMe)

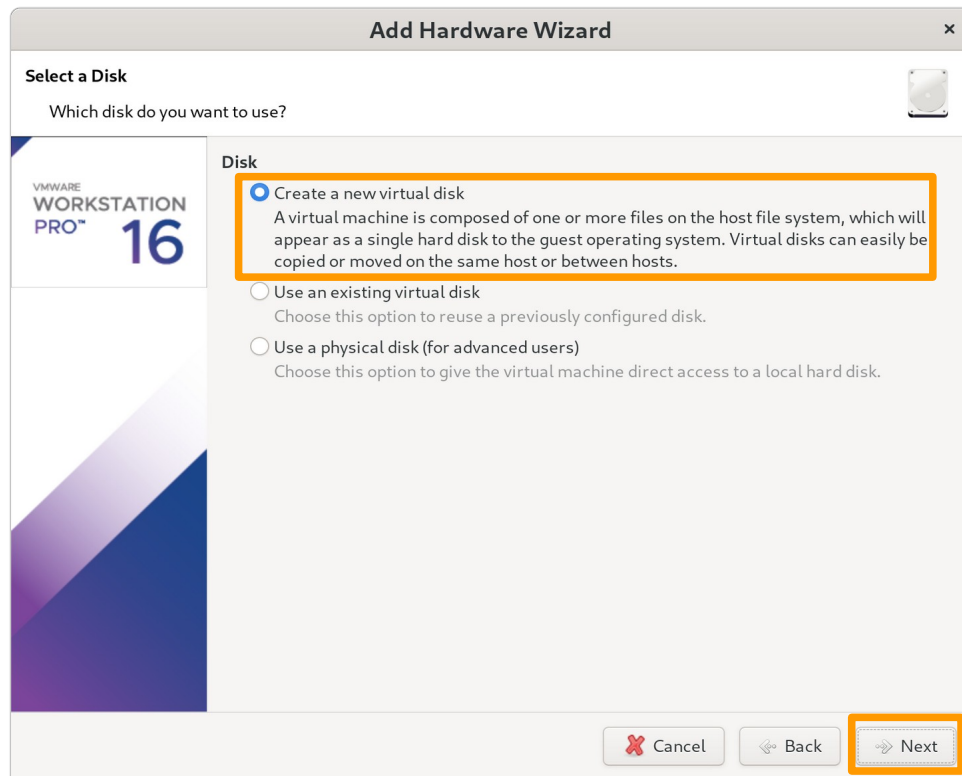
Si seleziona un disco di tipo "SCSI" e si preme il pulsante "Next".



Creazione disco nuovo

(Se ne potrebbe scegliere anche uno esistente virtuale o fisico, volendo)

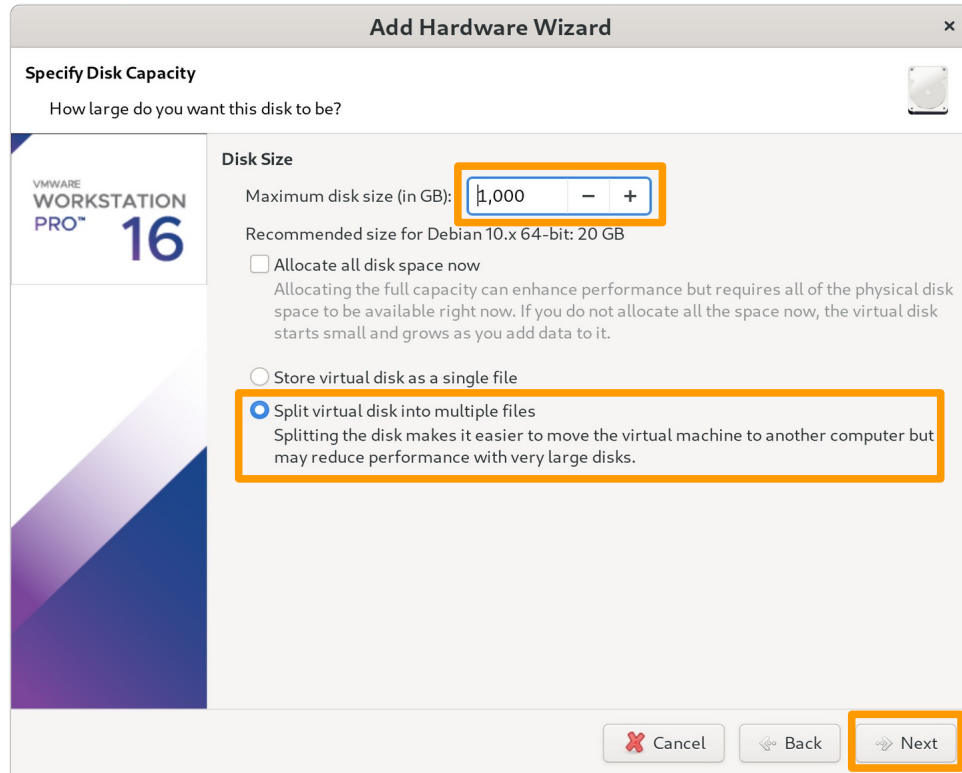
Si seleziona la voce "Create a new virtual disk" e si preme "Next".



Dimensione e modalità allocazione

(Un file singolo o tanti file da 2GB l'uno; allocazione dinamica o statica)

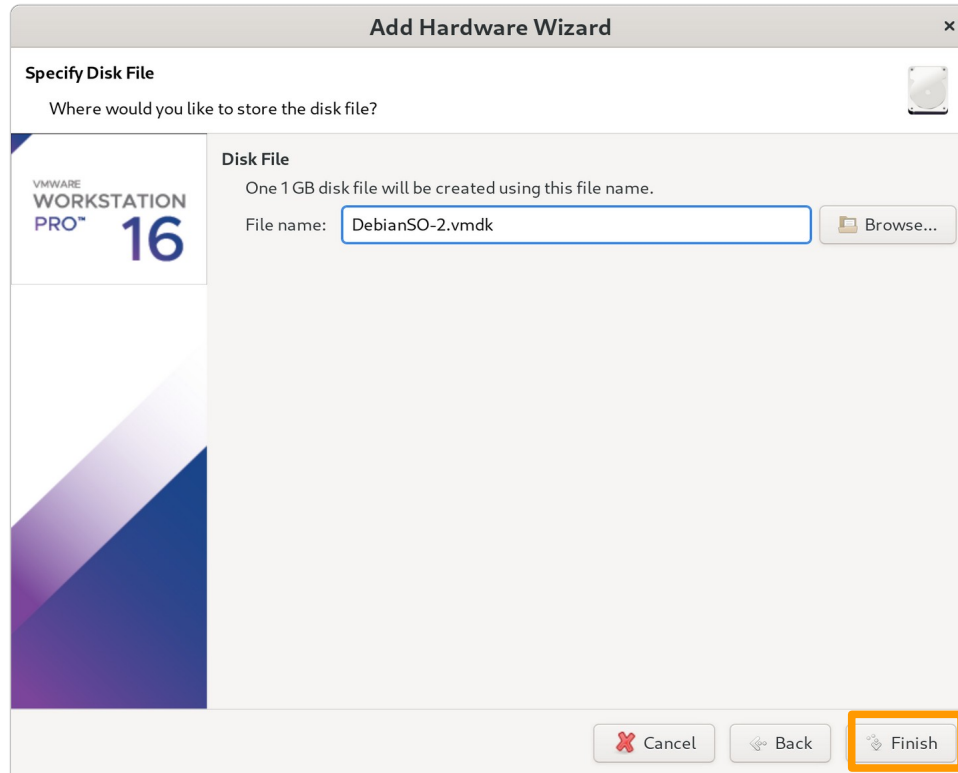
Si crea un disco di 1 GB diviso in più file, e si preme "Next".



Dimensione e modalità allocazione

(Un file singolo o tanti file da 2GB l'uno; allocazione dinamica o statica)

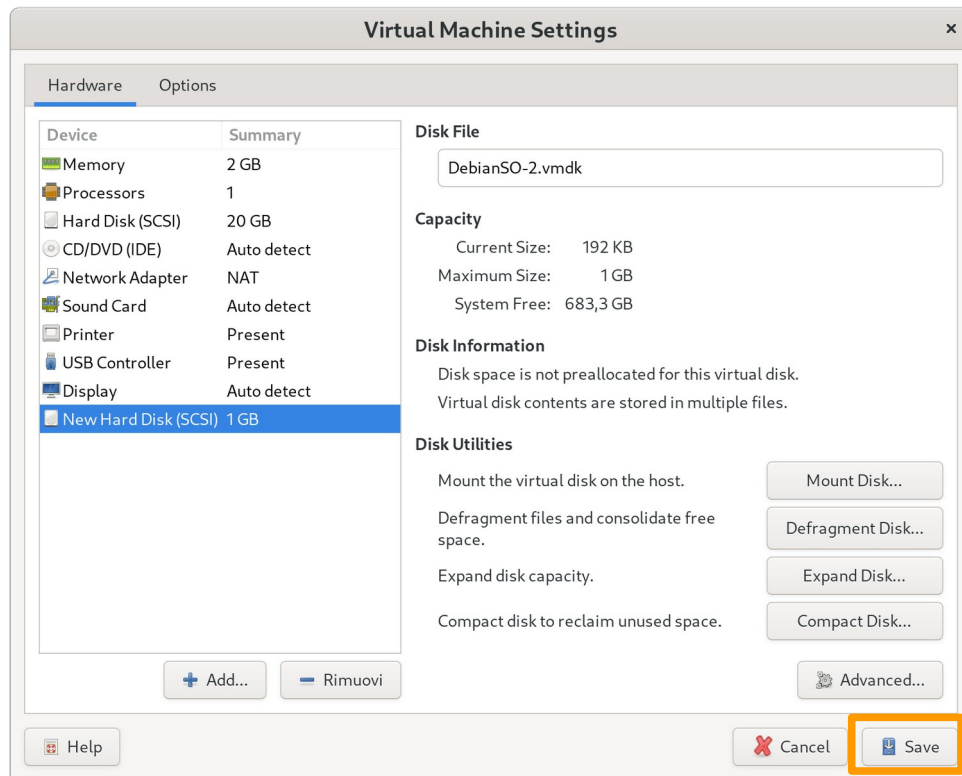
Viene proposto un nome di file. Lo si accetta premendo "Next".



Salvataggio impostazioni

(E il disco è allocato!)

Si salvano le impostazioni premendo il pulsante "Save".



Esercizio 1 (3 min.)

Create un nuovo disco fisso virtuale con le seguenti caratteristiche:

- dimensione pari ad 1 GB;
- allocazione dinamica;
- più file da 2 GB l'uno.

Agganciate il disco fisso al controller SCSI del sistema guest in uso.

PREPARAZIONE ALL'USO

Scenario

(Come si prepara un dispositivo all'uso?)

Scenario: l'utente ha appena comprato un nuovo dispositivo di memorizzazione di massa (ad es., un disco rigido) e vuole prepararlo all'uso.

Interrogativi:

Come si prepara un disco rigido all'uso?

Partizionamento del disco rigido

(Si usa il comando **fdisk**)

Prima di poter essere formattato, il disco rigido va partizionato.

In GNU/Linux, l'impostazione della tabella delle partizioni avviene tramite il comando **fdisk**.

Ne esistono anche altri (**cfdisk**, **gnome-disks**, ...).

Il comando **fdisk**

(Gestisce la tabella delle partizioni di un disco rigido)

Il comando **fdisk** gestisce la tabella delle partizioni di un disco rigido. La sintassi è semplice:

```
fdisk [opzioni] file_dispositivo
```

L'interfaccia fornita da **fdisk** è interattiva. Se si usano le opzioni da linea di comando, è non interattiva.

Domanda

(Spontanea)

Come si identifica il file speciale del disco rigido appena inserito?

Si sa che il file risiede nella directory `/dev`. Tuttavia, il comando

```
ls -l /dev/
```

fornisce tanti file. Qual è quello corretto?

Studio del problema

(Tramite le pagine di manuale)

Fortunatamente vengono in aiuto le pagine di manuale, tramite il comando **apropos**.

MEMENTO: nella Sezione 4 del manuale sono contenute tutte le pagine di documentazione relative ai file speciali.

Nel dubbio, si esegua il comando **man man** per recuperare l'elenco dei capitoli del manuale.

Si stampino tutte le pagine di manuale nella Sezione 4 del manuale:

```
apropos -s 4 -r '.*'
```


Individuazione delle pagine corrette

("Potenzialmente" corrette, in realtà)

Quali sono le pagine di manuale dei dischi?

`hd(4)` – MFM/IDE hard disk devices

`sd(4)` – driver for SCSI disk drives

Leggendole, si scopre che i file speciali dei dischi sono scritti in un formato ben preciso.

`man 4 hd`

`man 4 sd`

Il formato dei file speciali: disco intero

(`/dev/[hs]d[a-z]`)

I file speciali che rappresentano interi dischi rigidi hanno il formato seguente.

`/dev/[tipo]d[id_disco]`

Tipo: **h** (dischi IDE), **s** (dischi SCSI o SATA).

ID disco: una lettera univoca associata al disco, partendo dalla **a**.

Il formato dei file speciali: partizione

(`/dev/[hs]d[a-z][0-9]+`)

I file speciali che rappresentano singole partizioni dei dischi rigidi hanno il formato seguente.

`/dev/[tipo]d[id_disco][id_partizione]`

Tipo: **h** (dischi IDE), **s** (dischi SCSI o SATA).

ID disco: una lettera univoca associata al disco, partendo dalla **a**.

ID partizione: un numero intero univoco associato al disco, partendo dall'**1**.

Alcuni esempi

(Nella speranza che siano comprensibili)

/dev/sda: intero primo disco SATA.

/dev/sda1: prima partizione del primo disco SATA.

/dev/hdb2: seconda partizione del secondo disco IDE.

Individuazione del disco rigido nuovo

(Uno dei tanti metodi possibili)

Uno dei tanti modi per elencare il disco rigido nuovo è quello di elencare la directory `/dev` alla ricerca dei file speciali relativi.

```
ls -l /dev/[hs]d*
```

```
studente@debian:~$ ls -l /dev/[hs]d*
brw-rw---- 1 root disk 8,  0 nov 11 21:47 /dev/sda
brw-rw---- 1 root disk 8,  1 nov 11 21:47 /dev/sda1
brw-rw---- 1 root disk 8,  2 nov 11 21:47 /dev/sda2
brw-rw---- 1 root disk 8,  5 nov 11 21:47 /dev/sda5
brw-rw---- 1 root disk 8, 16 nov 11 21:47 /dev/sdb
studente@debian:~$
```

`/dev/sda`
`/dev/sda1`
`/dev/sda2`
`/dev/sda5`
`/dev/sdb`

Primo
disco SATA

Partizioni
primo disco SATA

Secondo
disco SATA

Risultato dell'indagine

(Il primo disco è partizionato, il secondo no)

Sono presenti due dischi SATA e nessuno di tipo IDE.

Il primo disco è partizionato nel modo seguente:

- una partizione primaria (**/dev/sda1**).

- una partizione estesa (**/dev/sda2**), contenitore di partizioni logiche.

- una partizione logica (**/dev/sda5**).

Il secondo disco non è partizionato.

→ È il nuovo disco appena inserito.

Invocazione di **fdisk**

(Occhio alle trappole!)

Per partizionare il disco rigido appena inserito, è necessario invocare **fdisk** nel modo seguente.

```
fdisk /dev/sdb
```

OCCHIO! Si partiziona sempre l'intero dispositivo, mai una singola partizione!

Troubleshooting 1/2

(fdisk non è nel percorso di un utente normale Debian)

È probabile che il comando appena immesso non funzioni su Debian. Perché?

fdisk è un comando di amministrazione che si trova nella directory **/sbin**. In Debian, un utente normale non ha **/sbin** nel percorso.

Digitando **/sbin/fdisk** il comando viene eseguito anche da utente normale.

Troubleshooting 2/2

(fdisk richiede diritti di amministratore)

È probabile che il comando appena immesso non funzioni su Debian. Perché?

fdisk richiede diritti di amministratore per poter aggiornare la tabella delle partizioni.

Partirebbe pure senza diritti di amministratore.

Quando si aggiorna la tabella delle partizioni si ottiene il famigerato *Permission denied*.

Invocazione corretta di fdisk

(Una operazione tutto sommato semplice)

Si invochi **fdisk** con le credenziali di **root**:

fdisk /dev/sdb

Dovrebbe presentarsi il menu seguente.

```
root@debian:~# fdisk /dev/sdb
```

```
Welcome to fdisk (util-linux 2.33.1).
```

```
Changes will remain in memory only, until you decide to write them.  
Be careful before using the write command.
```

```
Device does not contain a recognized partition table.  
Created a new DOS disklabel with disk identifier 0x22783302.
```

```
Command (m for help): █
```

Invocazione dell'help in linea

(Si preme il tasto **m** e si poi **<INVIO>**)

Si preme il tasto **m** e si preme **<INVIO>**.

Si dovrebbe ottenere un elenco delle operazioni disponibili.

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

Help:

DOS (MBR)

- a toggle a bootable flag
- b edit nested BSD disklabel
- c toggle the dos compatibility flag

Generic

- d delete a partition
- F list free unpartitioned space
- l list known partition types
- n add a new partition
- p print the partition table
- t change a partition type
- v verify the partition table
- i print information about a partition

Misc

- m print this menu
- u change display/entry units

Stampa tabella delle partizioni

(Si preme il tasto **p** e si poi **<INVIO>**)

Si preme il tasto **p** e si preme **<INVIO>**.

Si dovrebbe ottenere la tabella delle partizioni attuale.

Command (m for help): m

Help:

DOS (MBR)

- a toggle a bootable flag
- b edit nested BSD disklabel
- c toggle the dos compatibility flag

Generic

- d delete a partition
- F list free unpartitioned space
- l list known partition types
- n add a new partition
- p print the partition table**
- t change a partition type
- v verify the partition table
- i print information about a partition

Misc

- m print this menu
- u change display/entry units

La tabella delle partizioni

(È inizialmente vuota)

Inizialmente la tabella delle partizioni è vuota.
Sono stampate solo informazioni relative alla geometria del disco.

Numero di byte totali, numero di settori, dimensione di un settore, dimensione I/O minimo, tipo di tabella.

```
Command (m for help): p
Disk /dev/sdb: 1 GiB, 1073741824 bytes, 2097152 sectors
Disk model: VMware Virtual S
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x22783302
```

Aggiunta di una nuova partizione

(Si preme il tasto **n** e si poi **<INVIO>**)

Si preme il tasto **n** e si preme **<INVIO>**.

Si aggiunge una nuova partizione al disco rigido.

Command (m for help): m

Help:

DOS (MBR)

- a toggle a bootable flag
- b edit nested BSD disklabel
- c toggle the dos compatibility flag

Generic

- d delete a partition
- F list free unpartitioned space
- l list known partition types
- n add a new partition**
- p print the partition table
- t change a partition type
- v verify the partition table
- i print information about a partition

Misc

- m print this menu
- u change display/entry units

Configurazione della partizione

(Partizione primaria sull'intero disco)

Si inserisce una partizione primaria (tasto **p**) con identificatore e dimensioni di default (tasto **INVIO**).

Partizione primaria → Command (m for help): n
ID di partizione → Partition type
Settore iniziale → p primary (0 primary, 0 extended, 4 free)
Settore finale → e extended (container for logical partitions)
→ Select (default p): p
→ Partition number (1-4, default 1): 1
→ First sector (2048-2097151, default 2048):
→ Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-2097151, default 2097151):
Created a new partition 1 of type 'Linux' and of size 1023 MiB.

Selezione tipo della nuova partizione

(Si preme il tasto **t** e si poi **<INVIO>**)

Si preme il tasto **t** e si preme **<INVIO>**.

Si seleziona la tipologia di tabella delle partizioni desiderata.

Command (m for help): m

Help:

DOS (MBR)

- a toggle a bootable flag
- b edit nested BSD disklabel
- c toggle the dos compatibility flag

Generic

- d delete a partition
- F list free unpartitioned space
- l list known partition types
- n add a new partition
- p print the partition table
- t change a partition type**
- v verify the partition table
- i print information about a partition

Misc

- m print this menu
- u change display/entry units

Selezione delle voci disponibili

(Si preme il tasto **L** e si poi **<INVIO>**)

Si preme il tasto **L** e si preme **<INVIO>**.

Sono stampate le tipologie di partizionamento supportate da **fdisk**, insieme all'identificatore relativo in esadecimale.

```
Command (m for help): t
Selected partition 1
Hex code (type L to list all codes): L
```

0	Empty	24	NEC DOS	81	Minix / old Lin	bf	Solaris
1	FAT12	27	Hidden NTFS Win	82	Linux swap / So	c1	DRDOS/sec (FAT-
2	XENIX root	39	Plan 9	83	Linux	c4	DRDOS/sec (FAT-
3	XENIX usr	3c	PartitionMagic	84	OS/2 hidden or	c6	DRDOS/sec (FAT-
4	FAT16 <32M	40	Venix 80286	85	Linux extended	c7	Syrinx
5	Extended	41	PPC PReP Boot	86	NTFS volume set	da	Non-FS data
6	FAT16	42	SFS	87	NTFS volume set	db	CP/M / CTOS / .

Individuazione della voce corretta

(Per file system supportati da GNU/Linux → "83 Linux")

Per i file system nativi di GNU/Linux, la voce corretta è "83 Linux". Il codice esadecimale è 83.

Si immetta tale codice nel prompt di **fdisk** e si preme il tasto **<INVIO>**.

```
Command (m for help): t
Selected partition 1
Hex code (type L to list all codes): L
```

0	Empty	24	NEC DOS	81	Minix / old Lin	bf	Solaris
1	FAT12	27	Hidden NTFS Win	82	Linux swap / So	c1	DRDOS/sec (FAT-
2	XENIX root	39	Plan 9	83	Linux	c4	DRDOS/sec (FAT-
3	XENIX usr	3c	PartitionMagic	84	OS/2 hidden or	c6	DRDOS/sec (FAT-
4	FAT16 <32M	40	Venix 80286	85	Linux extended	c7	Syrinx
5	Extended	41	PPC PReP Boot	86	NTFS volume set	da	Non-FS data
6	FAT16	42	SFS	87	NTFS volume set	db	CP/M / CTOS / .

Immissione della voce corretta

(Si digiti il codice esadecimale ora individuato)

Si immetta tale codice nel prompt di **fdisk** e si prema il tasto **<INVIO>**.

```
Hex code (type L to list all codes): 83  
Changed type of partition 'Linux' to 'Linux'.
```

Stampa tabella delle partizioni

(Per verificare la correttezza dell'operazione appena svolta)

Si preme il tasto **p** e si preme **<INVIO>**.

La tabella delle partizioni dovrebbe contenere la partizione primaria appena creata.

```
Command (m for help): p
Disk /dev/sdb: 1 GiB, 1073741824 bytes, 2097152 sectors
Disk model: VMware Virtual S
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x22783302
```

Device	Boot	Start	End	Sectors	Size	Id	Type
/dev/sdb1		2048	2097151	2095104	1023M	83	Linux

Aggiornamento tabella delle partizioni

(La tabella è stata modificata in memoria, non sul disco rigido!)

La tabella delle partizioni è stata modificata in RAM, non sul disco rigido!

Per aggiornarla sul disco rigido è possibile usare l'opzione **w** (oppure **wq**) per uscire dal programma).

Si digiti la sequenza **wq** e si preme **<INVIO>**.

```
Command (m for help): wq
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

Esercizio 2 (3 min.)

Partizionate nel modo seguente il disco fisso creato nell'Esercizio 1:

- una sola partizione primaria;
- dimensione della partizione pari all'intero disco;
- tipologia di partizione "Linux".

Creazione di un file system

(Si usa la famiglia di comandi **mkfs**)

Prima di poter essere utilizzato, un dispositivo di memorizzazione di massa va preparato all'uso.

In GNU/Linux, la creazione di un file system avviene tramite il comando **mkfs**.

mkfs: "make file system".

Comando wrapper che invoca il comando giusto di formattazione.

I comandi di formattazione hanno nomi del tipo:

mkfs.tipo_fs

dove **tipo_fs** è il nome del file system.

Invocazione di **mkfs**

(Valgono le stesse considerazioni svolte per **fdisk**!)

Il comando **mkfs** è di semplice uso.

```
mkfs -t tipo_fs file_speciale_dispositivo
```

Valgono le stesse considerazioni svolte in precedenza per il comando **fdisk**!

- Non è nel percorso di un utente normale.

- Va eseguito da utente amministratore.

- Va individuato il file speciale della partizione.

Individuazione di un file system

(Una ulteriore scocciatura, purtroppo necessaria)

Bisogna inoltre individuare il tipo di file system che si vuole installare sul disco rigido.

Quali file system sono disponibili nel sistema?

Un modo rozzo ma efficace per individuarli consiste nell'elencare i comandi di formattazione presenti nella directory **/sbin**.

```
ls -l /sbin/mkfs.*
```

BIG FAT WARNING

(Bisognerebbe anche verificare il supporto fornito dal kernel!)

Ad esserre pignoli, questo controllo non sarebbe sufficiente.

Bisognerebbe anche verificare il supporto fornito dal nucleo per un dato file system.

Non basta installare le applicazioni di gestione se il nucleo non le supporta.

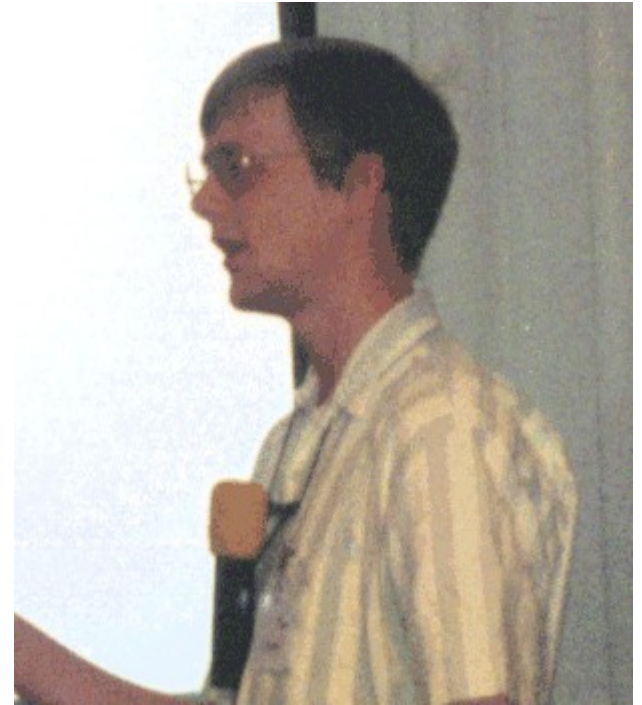
In questa introduzione si suppone che i file system siano supportati dal nucleo ed equipaggiati con le applicazioni di gestione.

Scelta di un file system

(EXT3, uno dei file system più popolari su GNU/Linux)

Nel resto della lezione si illustreranno le principali operazioni di preparazione e gestione di un file system Linux: EXT3.

Stephen Tweedie (2001).



Stephen Tweedie (1969 -)

Creazione di un file system EXT3

(Occhio al file speciale della partizione! Non vi sbagliate!)

Per creare un file system di tipo EXT3 sul nuovo disco rigido si esegue il comando seguente da amministratore:

```
mkfs -t ext3 /dev/sdb1
```

Tale comando scrive diverse strutture di controllo all'interno del file, in posizioni ben specifiche.

Il risultato è un file system che potrà essere utilizzato in seguito.

Esercizio 3 (1 min.)

Create un file system di tipo EXT3 nella partizione primaria del nuovo disco rigido.

AGGANCIO DI UN FILE SYSTEM

Scenario

(Come si associa/sgancia un file system ad/da una directory?)

Scenario: l'utente ha appena partizionato un disco rigido e formattato una delle sue partizioni. Ora vorrebbe attivare il file system e usarlo.

Interrogativi:

Come si "attiva" un file system?

Come si "disattiva" un file system dopo l'uso?

Aggancio di un nuovo file system

(File system mount)

Un file system, prima di essere utilizzato deve essere associato ad un dispositivo di memorizzazione secondaria;
deve essere agganciato ad un file system esistente, usando una directory come punto di attacco.

Tali due operazioni prendono il nome di **aggancio (mount)**.

La directory radice del file system prende il nome di **punto di aggancio (mount point)**.

File system di root

(Viene agganciato all'avvio del SO)

Affinché lo schema proposto funzioni, almeno un file system deve essere sempre agganciato ad una directory. Tale file system prende il nome di **file system di root** (**root file system**) con le seguenti caratteristiche.

- È associato alla directory /.

- Contiene almeno un comando di percorso `/sbin/init` che implementa il meccanismo di avvio e gestione dei servizi.

- L'associazione avviene all'avvio del SO.

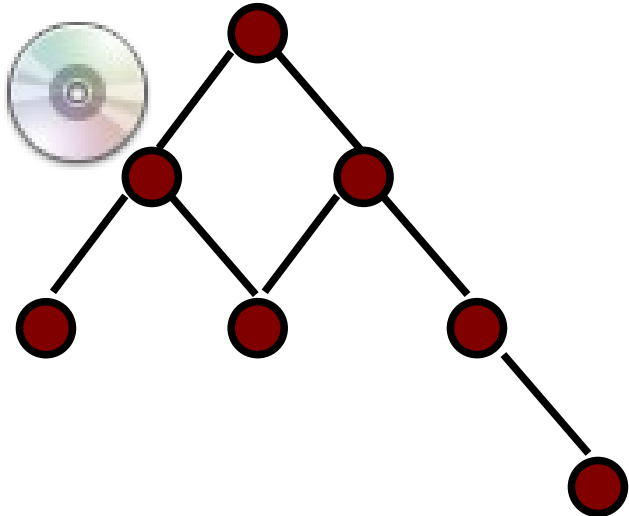
Aggancio file system: prima

(Due alberi staccati)

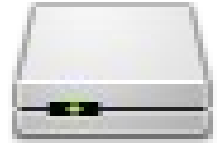
Elenco file system montati



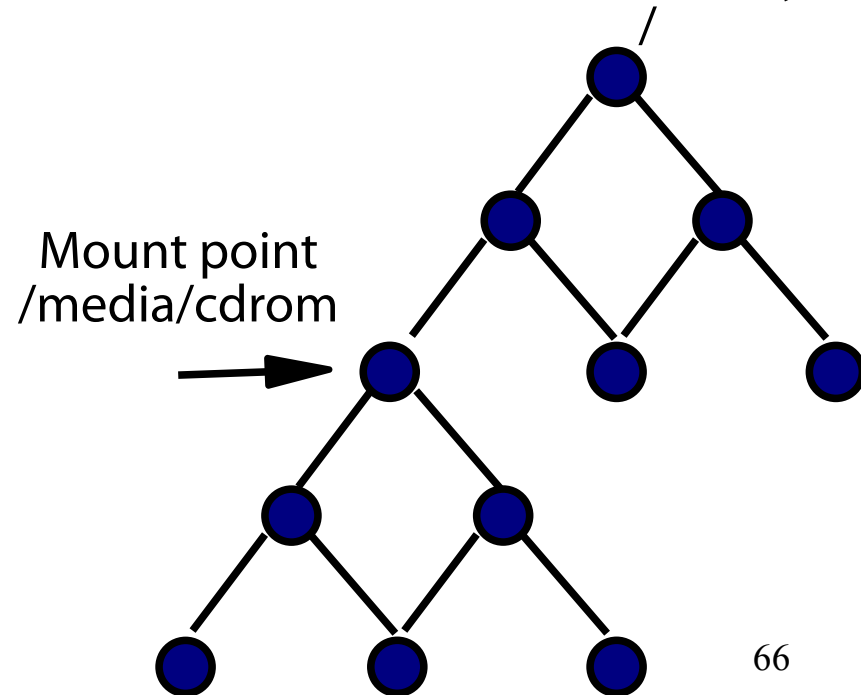
File system su CD-ROM



Root file system su disco rigido



Root directory



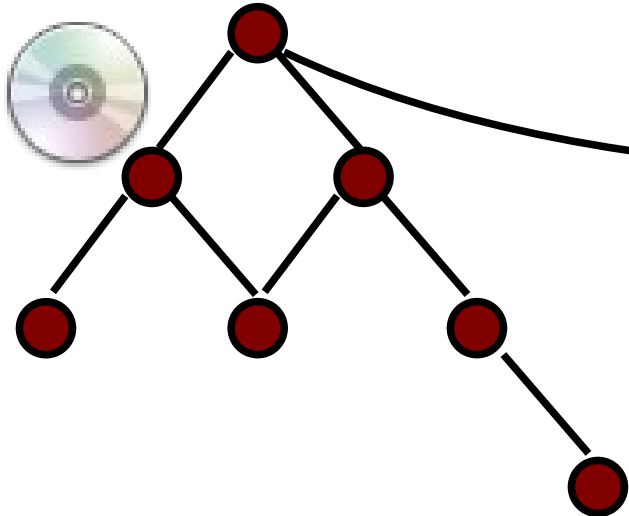
Aggancio file system: dopo

(Due alberi uniti)

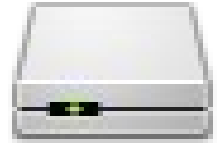
Tabella di mount

/	Disco rigido
/mnt	CDROM

File system
su CD-ROM

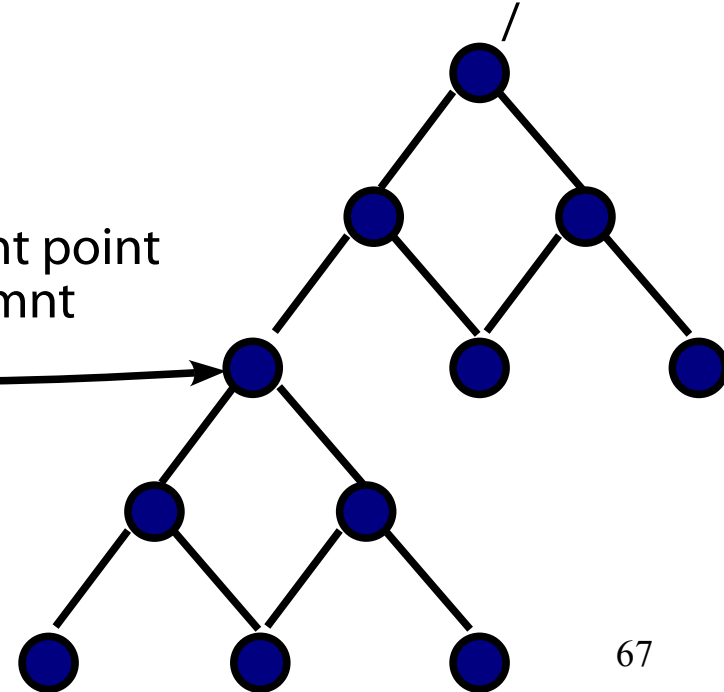


Root file system
su disco rigido



Root directory

Mount point
/mnt



Opzioni di mount

(Lettura e scrittura, esecuzione, sincrona, asincrona)

read-only: il file system è montato in sola lettura.

sync: le scritture sono sincrone (eseguite una dietro l'altra, in maniera bloccante).

async: le scritture sono asincrone (eseguite in blocco, più avanti, nel momento più favorevole).

exec: si permette l'esecuzione dei programmi.

Montaggio del file system in UNIX

(Si usa il comando **mount**)

Il comando **mount** permette di associare un supporto (file speciale, file contenente un file system) ad un mount point.

Sintassi generale:

```
mount -o options -t fs_type special_file mount_point
```

Un esempio di associazione

(Il comando **mount** in azione)

Per associare in sola lettura alla directory **/mnt** un file system di tipo EXT3 (contenuto nella partizione **/dev/sda2**), si esegue il comando seguente da **root**:

```
mount -o ro -t ext3 /dev/sda2 /mnt
```

Se non si specificano opzioni e tipo, **mount** prova a scoprire il tipo del file system e a montarlo con opzioni di default (read-write, esecuzione).

Aggancio del file system creato

(Si usa sempre il comando **mount**)

Si provi a montare con il comando seguente il file system creato precedentemente:

```
mount /dev/sdb1 /mnt
```

La directory **/mnt** è, per convenzione, dedicata al montaggio di un file system esterno.

Nulla osta a montare il file system su un'altra directory.

Esercizio 4 (2 min.)

Leggete la documentazione del comando **mount** e trovate un modo per visualizzare tutti i file system agganciati.

Visione dei file system su supporto

(Si usa il comando **lsblk**)

Il comando **lsblk** visualizza i file system memorizzati su supporti fisici o file, con annessi mount point.

lsblk

Le opzioni di **lsblk** configurano le possibili modalità di visualizzazione.

Sono tante; **man lsblk** per tutti i dettagli.

Esercizio 5 (2 min.)

Leggete la documentazione del comando **lsblk** e individuate una modalità di visualizzazione contenente il tipo di file system.

Mostrate tutti i file system associati ed i relativi tipi.

Verifica di un punto di aggancio

(Si usa il comando **mountpoint**)

Il comando esterno **mountpoint** controlla se una data directory sia un punto di aggancio oppure no.

```
mountpoint /  
mountpoint /home  
mountpoint /proc
```

Sgancio del file system

(File system umount)

Il file system può essere staccato dal suo punto di aggancio tramite l'operazione di **sgancio** (**umount** nel gergo UNIX), sostanzialmente l'inversa di mount.

Sgancio del file system in UNIX

(Si usa il comando **umount**)

Una volta usato il file system, lo si può sganciare dal mount point con il comando seguente:

```
umount [special_file | mount_point]...
```

È possibile specificare il file speciale del dispositivo ospitante il file system oppure il mount point:

```
umount /dev/sda2
```

```
umount /mnt
```

Una doverosa osservazione

(Nessuna applicazione deve usare file o directory del file system)

Per sganciare il file system è necessario che nessuna applicazione stia attualmente usando alcun suo file o directory!

Un classico esempio è una sessione di BASH aperta in una sottodirectory del file system.

Esercizio 6 (1 min.)

Sganciate il file system ospitato sul disco esterno.

CONTROLLO DI CONSISTENZA

Scenario

(Come si prepara un dispositivo all'uso? Come lo si attiva? Come si controlla?)

Scenario: l'utente ha sganciato un file system e vorrebbe controllarne la struttura.

Interrogativi:

Come si fa a controllare la consistenza di un file system?

Controllo di consistenza e riparazione

(Nel caso in cui un file system sia danneggiato)

Ciascun file system ha bisogno di un controllo periodico di consistenza e riparazione.

In GNU/Linux, consistenza e riparazione di un file system sono gestite dal comando **fsck**.

fsck: "file system consistency check".

Comando wrapper che invoca il comando giusto di controllo della consistenza e riparazione.

I comandi di gestione hanno nomi del tipo:

fsck.tipo_fs

dove **tipo_fs** è il nome del file system.

Attivazione del controllo

(All'avvio oppure manualmente)

Quando avviene il controllo?

Al boot, automaticamente, per il controllo della struttura in caso di problemi rilevati durante la procedura di spegnimento precedente.

Manualmente, da linea di comando (a file system sganciato), con possibilità di riparazione.

Controllo di un file system

(Eseguitelo a file system sganciato!)

Per controllare il file system di tipo EXT3 sul nuovo disco rigido si digiti il seguente comando da amministratore:

```
fsck /dev/sdb1
```

ATTENZIONE! Il file system deve essere sganciato, pena la possibile corruzione del medesimo!

Esercizio 7 (2 min.)

Effettuate un controllo di consistenza del file system creato nell'Esercizio 4.

Leggendo le opportune pagine di manuale, individuate un modo per forzare il controllo di esistenza di blocchi cattivi (**bad block**).