

Percorso 5:

- Gestione dischi



Bartolomeo Montrucchio

`bartolomeo.montrucchio@polito.it`

Giovanni Squillero

`giovanni.squillero@polito.it`

Filesystem

- Il file system è organizzato in blocchi logici contigui
 - dimensione fissa di 1024, 2048 o 4096 byte
 - indipendente dalla dimensione del blocco fisico (generalmente 512 byte)
- Un blocco speciale, detto super block, in posizione fissa all'inizio del FS ne descrive le caratteristiche

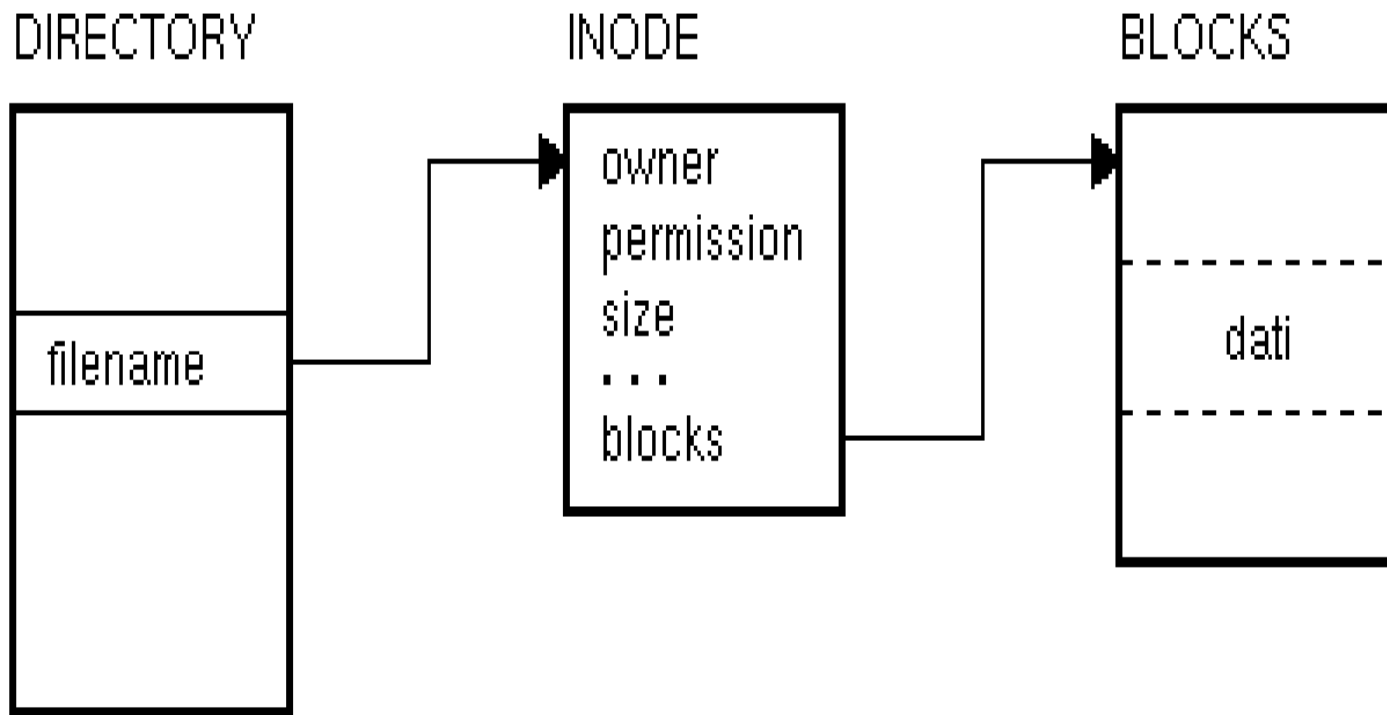
Super block

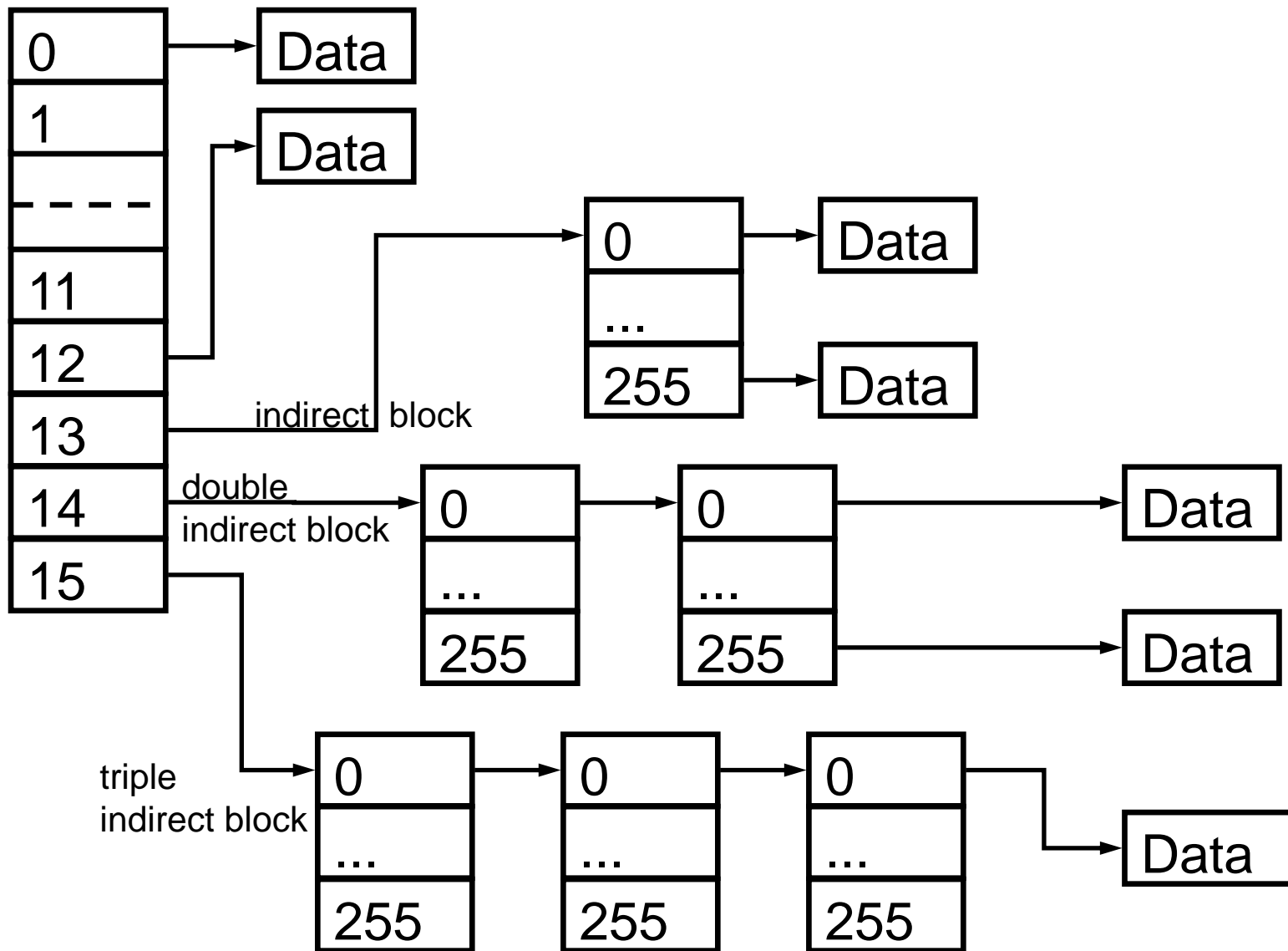
- Caratteristiche del FS
 - dimensioni, struttura, ...
- Parametri modificabili del FS
 - max mount count, ...
- Stato del FS
 - state, free blocks, mount count, ...
- *Non* contiene informazioni sulla struttura fisica e sul tipo di device utilizzato (perché ne costituisce una astrazione)

Inode

- Ogni file o directory è associato ad un inode
 - descrive le caratteristiche del file
 - identifica i blocchi di cui è composto il file
- Numerati a partire da 1
- Alcuni sono riservati al sistema
 - bad block inode, root inode, boot loader inode

Inode





link

- Hard link (fisico)
 - directory entry che punta ad un inode puntato da un'altra directory entry
 - no cross-file system
- Soft link (simbolico)
 - directory entry che punta ad un inode che punta ad un blocco contenente il path name del file corrispondente
- Fast soft link
 - il path name (<60 byte) è memorizzato nell'inode (nei 15 byte dell'**i_block**)

Virtual File System

- Linux supporta diversi FS:
 - ext2, minix, msdos, vfat, iso9660, hpfs, ...
- VFS astrae il particolare FS fisico occupandosi di mappare le system call orientate ai file sulle funzioni supportate dal FS
- Al mount VFS individua (attraverso una tabella di configurazione del kernel) la specifica funzione da utilizzare che:
 - legge il super blocco e ne inizializza le variabili
 - restituisce a VFS un descrittore di file con cui verranno mappate le successive system call

File descriptor

- Individua univocamente un file aperto
- Ciascun processo può utilizzare fino a `OPEN_MAX` (`<limits.h>`) file descriptor
- File descriptor di default:
 - 0 → standard input
 - 1 → standard output
 - 2 → standard error

Macchine virtuali

- Sviluppate inizialmente da IBM (inizio anni '70)
- Sono usate per cross-emulare od emulare altre architetture HW (anche per videogiochi o calcolatrici (emu48))
- VMware, Virtual Box
- Xen, KVM (paravirtualizzazione)
- Host è la macchina (una sola) che ospita le varie macchine virtuali (possono essere numerose, ma RAM e disco devono essere adeguati, oltre ad almeno una CPU per ogni guest)
- Guest è la/le macchine ospitate
- Qui utilizzeremo due o tre guest in contemporanea
- Virtual Box non è l'implementazione più potente, ma è molto adatta a scopi didattici ed è gratuita

Virtual Box

- VirtualBox 4.3.14 Oracle VM VirtualBox Extension Pack
 - Contiene ad esempio il supporto per USB 2.0
 - Le versioni possono avere bug, verificare sempre
- Guest additions
 - Vengono installate nel sistema operativo guest montando un cd virtuale (sotto Linux viene coinvolto il kernel) e danno miglioramenti soprattutto su:
 - Mouse pointer integration
 - Shared folders
 - Better video support
 - Seamless windows
 - Shared clipboard

Virtual Box - Formato .OVA

- Import/Export Appliance
- Il formato .OVA (meglio 1.0) permette di esportare il guest anche verso alcuni altri sistemi (es. Da Virtual Box a VMWare)
- All'interno il file system è in formato .OVF
- Utilizzare anche l'opzione Manifest per aggiungere dati sul guest

Disk Utility palimpsest

- Dopo la 12.04 è gnome-disks
- Permette di gestire i dischi tramite interfaccia grafica
- formattazione, edit, check, mount

Disk Utility

fdisk

- fdisk
- cfdisk, semigrafico
- Ha una interfaccia a caratteri
- **ATTENZIONE:** le modifiche al file system si hanno solo dopo l'uso dell'opzione w
- Può essere richiesto un reboot
- **ATTENZIONE:** è molto facile fare gravi danni!

Disk Utility mkfs

- Viene utilizzato per la formattazione dei dischi o delle partizioni
- Si può scegliere il tipo di file system
- Nelle nuove versioni l'ext4 è il file system di scelta
- L'utilizzo del journaling rende le operazioni molto più affidabili

mount, /etc/fstab

- Mount permette di montare il file system nell'unico albero come già visto in precedenza nel corso
- fstab permette di specificare quali filesystem e dove montarli
- mount -a permette di rimontare tutto ciò che è presente in fstab
 - Molto utile dopo modifiche, per vedere che tutto funzioni senza dover fare reboot

Esercizio

- Come si può utilizzare una penna USB prima sull'host e poi sul guest?
- Provare a formattare/montare una penna USB in formato FAT32 e poi in formato ext4
- Verificare l'utilizzabilità dei permessi sui file sia su FAT32 sia su ext4 sulla penna USB
 - Provare a creare file e sottodirectory settandone i permessi Unix per poi verificarli

Disk Utility

fsck

- Permette di effettuare una verifica di correttezza del filesystem da un punto di vista logico
- Tramite i dati ridondati inseriti nel filesystem può riparare file system danneggiati
- Un tipico danno può essere dovuto a mancanza di corrente
- L'approccio tipico del file system è write back e non write through
- Non si può utilizzare su file system montati
- Se il file system è unico si può usare un cd per fare il boot di recovery
- Ricordare sempre, in caso di guasto anomalo, di verificare la memoria di sistema tramite il memtest, sempre da cd Live

Disk Usage Analyzer baobab

- Permette di effettuare una scansione del disco per effettuare una statistica dell'uso
- `du -ks` permette di avere risultati simili da terminale, pur con forti limitazioni

Quota

- Si tratta della possibilità di limitare l'uso del disco da parte degli utenti
- Richiede il pacchetto quota
- La configurazione richiede di modificare `/etc/fstab`, oltre all'uso di `edquota`

tune2fs

- Permette di settare i vari parametri del filesystem nel caso di ext2, ext3 ed ext4
- Con `tune2fs -l` si possono vedere i parametri correnti
- `tune2fs -l /dev/sda1 | more`

dumpe2fs

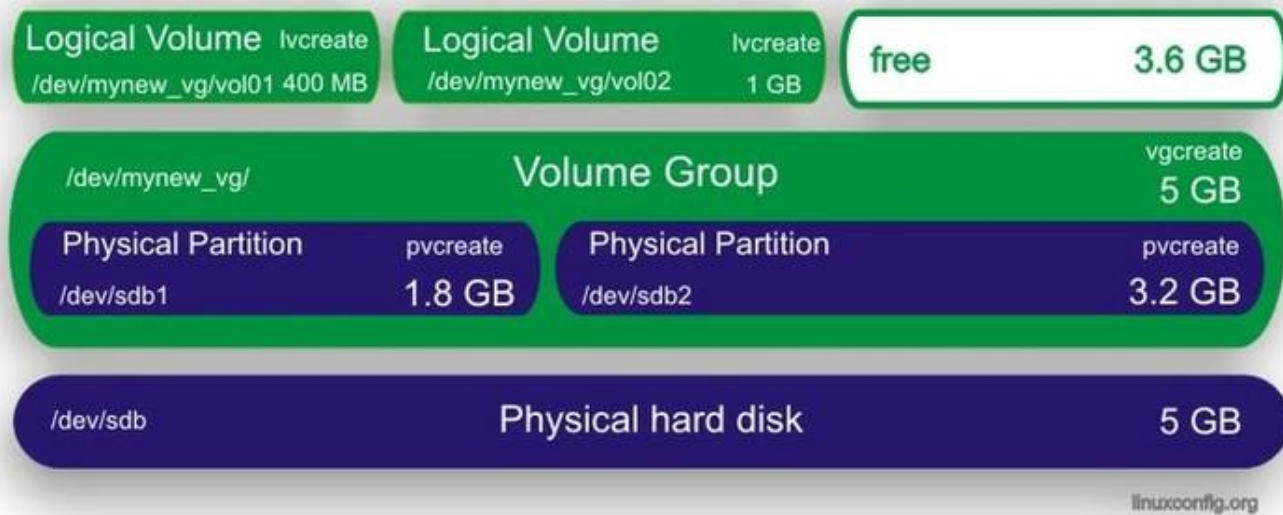
- Stampa le caratteristiche del super blocco e di altre informazioni per il filesystem presente nel dispositivo
- `dumpe2fs /dev/sda1 | more`

Esercizi

- Utilizzando la macchina virtuale di tipo server:
 - Individuare l'eventuale spazio libero su disco
 - Creare una o più partizioni
 - Montarle sul file system e verificarne il funzionamento
- Cosa succede estraendo una penna USB senza effettuare l'umount? Provare, per quanto possibile
- Le penne USB (come le compact flash e le schede di tipo SD) che tipo di file system hanno? Perché? Quali sono le implicazioni hardware delle memorie allo stato solido?

LVM(1)

- Il Logical Volume Manager [5] serve a gestire i dischi con maggiore flessibilità
 - Si può per esempio estendere un disco già esistente in caso di spazio insufficiente



LVM(2)

- Creare le partizioni
 - fdisk per creare una o più partizioni fisiche
 - /dev/sdb1 e /dev/sdb2 in questo esempio
- Creare i volumi fisici
 - # pvcreate /dev/sdb1
 - # pvcreate /dev/sdb2
 - # pvdisplay
- Creare i gruppi virtuali (qui uno composto da due volumi fisici)
 - # vgcreate mynew_vg /dev/sdb1 /dev/sdb2
 - si possono anche estendere con # vgextend mynew_vg /dev/sdb2 (partendo da uno solo)

LVM(3)

- Creare i volumi logici
 - # lvcreate -L 400 -n vol01 mynew_vg
 - # lvcreate -L 1000 -n vol02 mynew_vg
 - #lvdisplay
 - #vgdisplay
- Creare i file system sui volumi logici
 - # mkfs.ext3 -m 0 /dev/mynew_vg/vol01
 -

LVM(4)

- Preparare /etc/fstab con i dati richiesti

```
# /etc/fstab: static file system information.
#
# <file system> <mount point> <type> <options> <dump> <pass>
proc /proc proc defaults 0 0
/dev/sda1 / ext3 defaults,errors=remount-ro 0 1
/dev/sda9 /home ext3 defaults 0 2
/dev/sda8 /tmp ext3 defaults 0 2
/dev/sda5 /usr ext3 defaults 0 2
/dev/sda6 /var ext3 defaults 0 2
/dev/sda7 none swap sw 0 0
/dev/hdc /media/cdrom0 iso9660 ro,user,noauto 0 0
/dev/fd0 /media/floppy0 auto rw,user,noauto 0 0
/dev/mynew_vg/vol01 /home/foobar ext3 defaults 0 2
~
~
~
~
~
~
~
```

LVM(5)

```
linuxconfig.org# mkdir /home/foobar
linuxconfig.org# mount -a
linuxconfig.org# cd /home/foobar/
linuxconfig.org# df -h .
Filesystem                Size      Used Avail Use% Mounted on
/dev/mapper/mynew_vg-vol01 388M    8,1M   360M   3% /home/foobar
linuxconfig.org# █
```

- Preparare la directory e montarla
- Estendere un volume logico
 - # lvextend -L +800 /dev/mynew_vg/vol01
 - # resize2fs /dev/mynew_vg/vol01

LVM(6)

- Rimuovere un volume logico
 - # `lvremove /dev/mynew_vg/vol02`
- `lvdisplay` serve poi a vedere che sia tutto in ordine

Esercizio

- Provare ad installare una macchina virtuale Linux desktop lasciando spazio libero sul disco
- Provare ad utilizzare LVM per creare e modificare un file system, seguendo quanto appena visto [5]
- Potrebbe essere necessario installare il package lvm2 con apt-get

Bibliografia

- <https://wiki.ubuntu.com/>
- <http://wiki.ubuntu-it.org>
- <http://help.ubuntu-it.org/>
- <http://free-electrons.com/docs/>
- [5] <http://linuxconfig.org/linux-lvm-logical-volume-manager>
- <http://mohannetworking.wordpress.com/2012/09/29/how-to-configure-disk-quotas-on-ubuntu/>

These slides are licensed under a **Creative Commons**

**Attribution
Non Commercial
Share Alike
4.0 International**

To view a copy of this license, visit

<http://creativecommons.org/licenses/by-nc-sa/4.0/>

Versione in Italiano:

<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.it>

