



CAPITOLO 17: Gestione dei File

The Architecture of Computer Hardware, Systems Software & Networking: An Information Technology Approach

5th Edition, Irv Englander

John Wiley and Sons ©2013

PowerPoint slides authored by Angela Clark, University of South Alabama

PowerPoint slides for the 4th edition were authored by Wilson Wong, Bentley University

PowerPoint slides modified by Gianluca Amato e Francesca Scozzari, Univ. Chieti-Pescara



Introduzione ai file

- Collezione di dati correlati
- Nella maggior parte dei SO moderni, un file è semplicemente una sequenza di byte
 - Programmi binari, programmi sorgente, immagini, directory, etc..
 - È la “**visione logica**” del file
- Per “**visione fisica**” si intende invece il modo in cui il file è fisicamente memorizzato su disco
 - Blocchi
 - minima unità di memorizzaz. su disco
 - Cluster
 - Minima unità di allocazione del file system



Introduzione ai file

- I file oltre al contenuto hanno vari attributi:
 - Nome
 - Estensione
 - Diritti di accesso
 - Associazione di una applicazione
 - Data di creazione/utilizzo

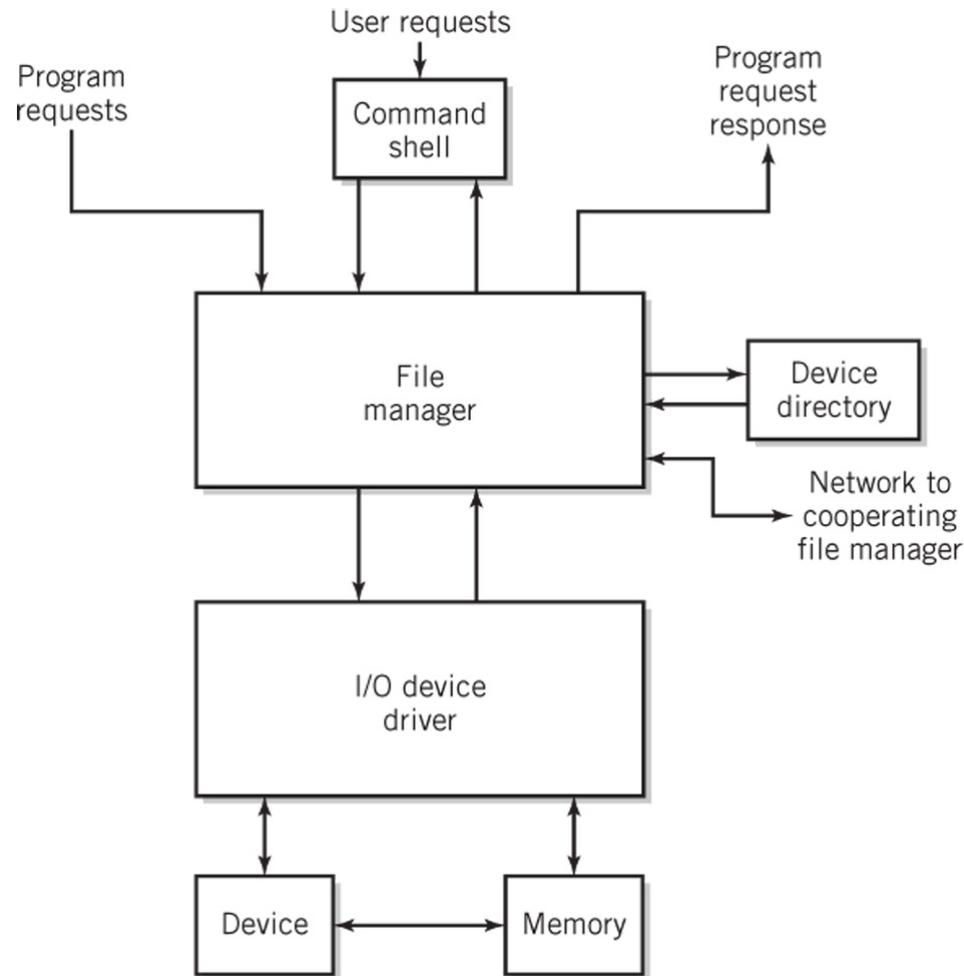


File Management System

- Fornisce agli utenti la visione logica dei file (sequenza di byte), e nasconde i dettagli sulla loro memorizzazione a livello fisico.
- Fornisce un insieme consistente di comandi che vengono tradotti in una forma appropriata al dispositivo fisico
- Organizza i file in directory
- Si occupa della sicurezza e protezione dei file



File Manager Request Handling





Operazioni sui file (1)

- Operazioni sul file intero
 - Copia, Spostamento su altro dispositivo
 - Visualizzazione, Stampa
 - Caricamento ed esecuzione di programmi
 - Caricamento file in memoria
 - Salvataggio file dalla memoria
 - Aggiungere dati da memoria ad un file



Operazioni sui file (2)

- Operazioni all'interno di un file
 - Apertura
 - Leggere alcuni byte da un file
 - Scrivere alcuni byte da un file
 - Spostare il puntatore del file
 - Chiudere un file



Operazioni su directory

- Operazioni su directory
 - Crea un nuovo file vuoto
 - Spostare un file da una cartella ad un'altra
 - Cambiare nome a un file
 - Cancellare un file



Metodi di accesso a file

- Accesso sequenziale
 - Il file è letto dall'inizio alla fine
 - Comune per tutti i file
- Accesso casuale
 - Consente di accedere ad una certa posizione del file
 - esempio: byte n. 2345, record n. 23
- Accesso indicizzato
 - Accesso ai record sulla base del valore di un campo chiave
 - Supportato solo dai file system che supportano viste logiche sofisticate (IBM z/OS)



Memorizzazione fisica

- Contigua
- Non-contigue
 - Collegata o concatenata
 - Indicizzata
- Esempi
 - DOS/Windows FAT
 - UNIX i-nodes
 - Windows NTFS
- Gestione spazio libero



Memorizzazione contigua

- Assegna blocchi consecutivi al file
- Accesso sequenziale e casuale semplice
- Svantaggi
 - Serve spazio contiguo per salvare i file
 - Problemi quando un file cresce
 - Frammentazione
- Strategie di allocazione per ridurre la frammentazione
 - First-fit, best-fit

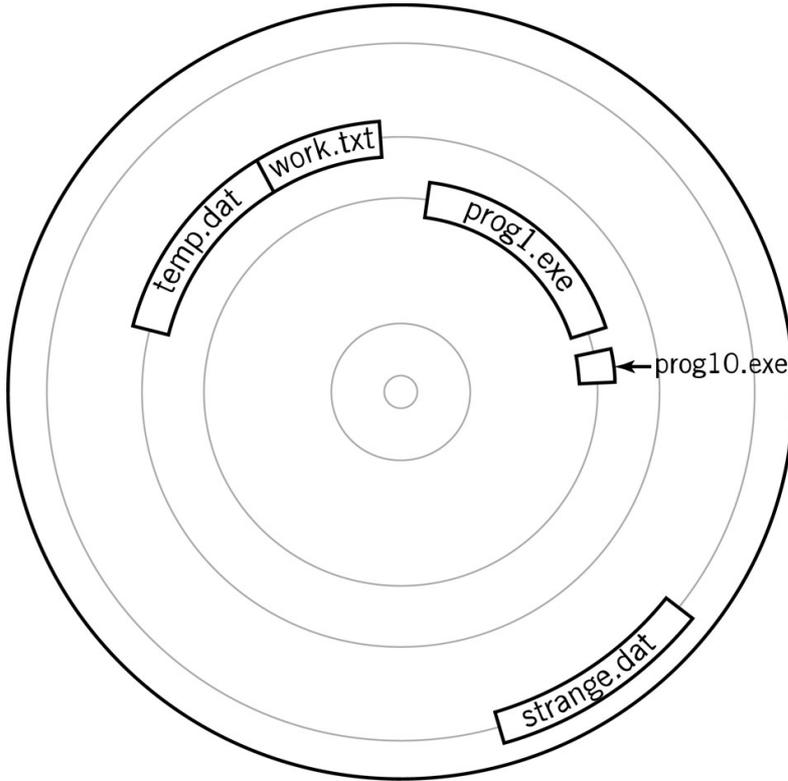


Allocazione concatenata

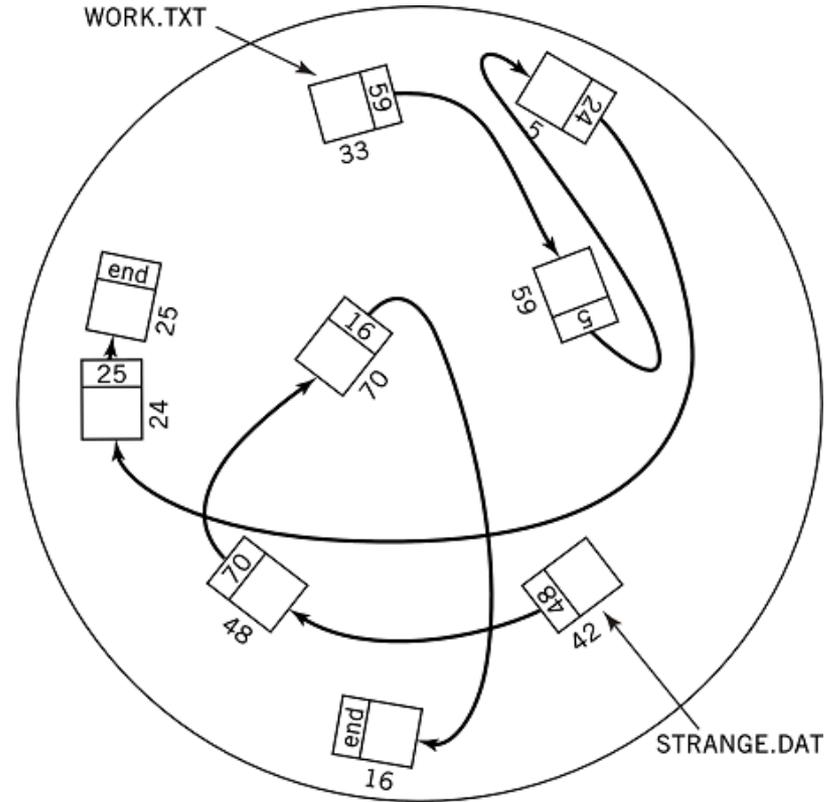
- Non-contigua
- Ogni blocco contiene un puntatore al blocco fisico successivo
- Svantaggi
 - Non c'è frammentazione (nel senso inteso prima)
 - Aggiungere dati ad un file è facile
- Svantaggi
 - **Non adatta ad accessi casuali**
 - Lettura di file frammentati (sparpagliati) richiede molti spostamenti della testina
 - Overhead per memorizzare i puntatori
 - Difficile recupero di settori danneggiati



File Allocation



Contiguous Storage Allocation



Linked Allocation

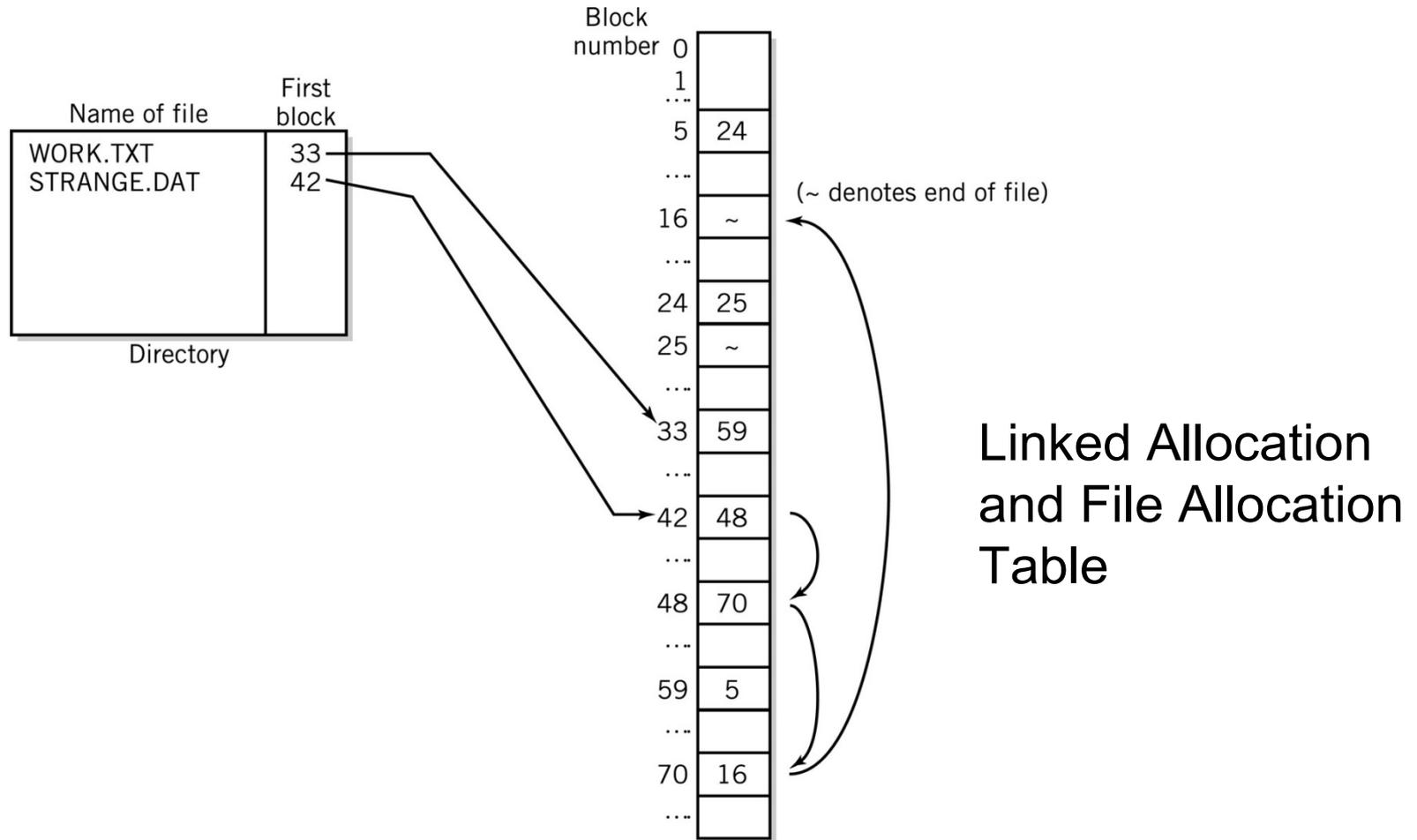


Microsoft FAT

- File Allocation Table (FAT)
 - Allocazione concatenata con puntatori memorizzati in una tabella a parte (FAT)
 - La directory contiene il numero del primo blocco di ogni file sul disco
 - I blocchi successivi vengono cercati nella tabella
- Rispetto alla soluzione collegata tradizionale
 - Richiede molto spazio aggiuntivo in memoria principale per la tabella
 - Consente accesso casuale



File Allocation Table





Allocazione indicizzata

- Non-contigua
- Tutti i puntatori ad i blocchi di file sono memorizzati in un blocco indice
- Un blocco indice per file
- Vantaggi
 - Consente accesso casuale
 - Devo tenere in memoria solo i blocchi indice dei file aperti
- Svantaggi
 - Lettura di file frammentati poco efficiente
 - Overhead per memorizzare i blocchi indice



Allocazione indicizzata

42
48
70
16
end

STRANGE.DAT

33
59
5
24
25
end

WORK.TXT

Blocchi indice per i file STRANGE.DAT e WORK.TXT

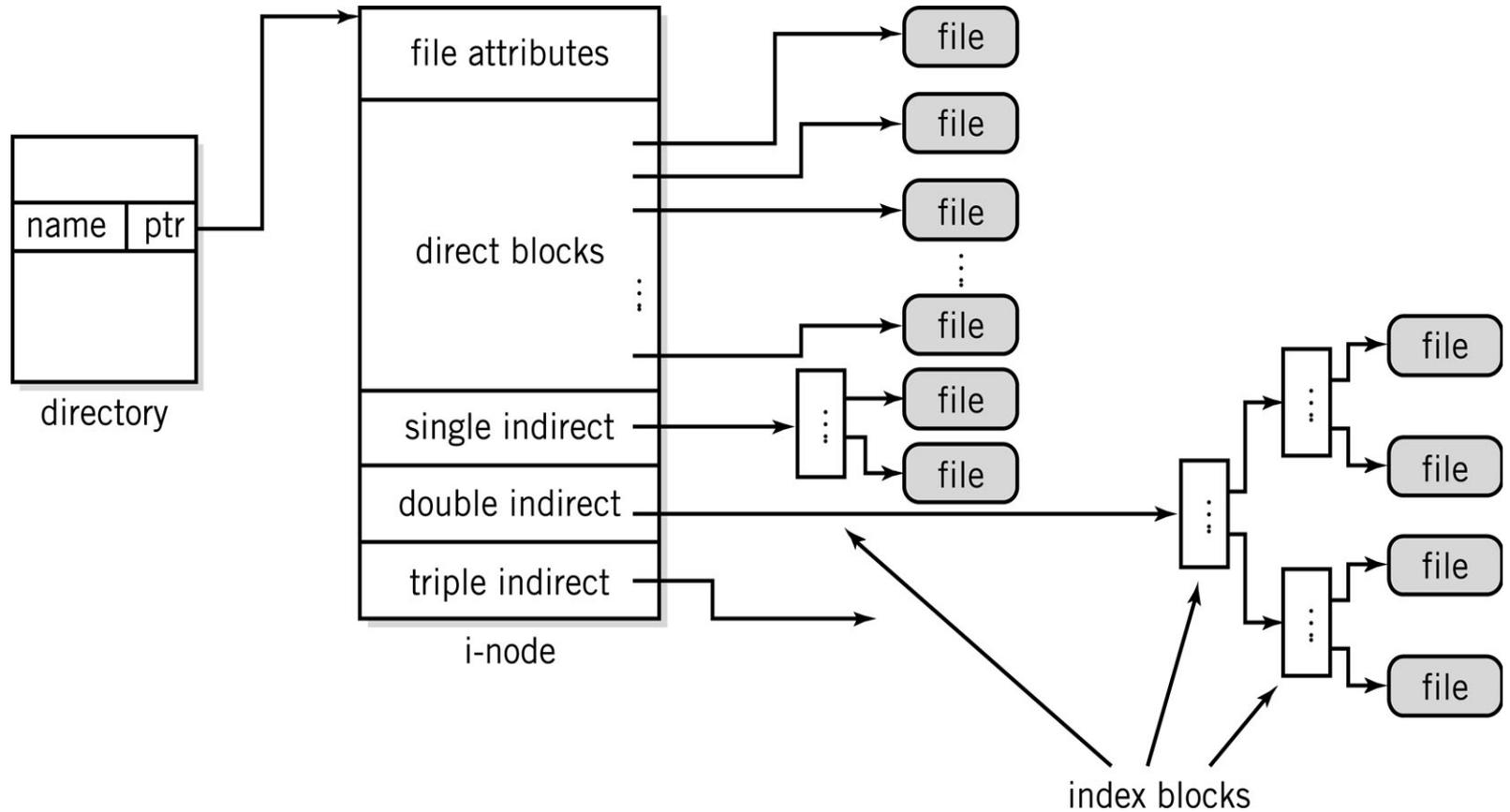


Unix i-nodes

- Allocazione indicizzata su Unix
- Un blocco indice (i-node) contiene
 - Attributi dei file
 - 10 puntatori diretti a blocchi
 - 1 puntatore a blocco indiretto
 - 1 puntatore a blocco indiretto doppio
 - 1 puntatore a blocco indiretto triplo
- Vantaggi
 - Veloce per file piccoli
 - Può gestire facilmente file molto grossi



Unix i-nodes





Gestione spazio libero

- Metodo della bitmap
 - Un bit per ogni blocco per indicare se è libero o occupato

Block numbers

0–15	1001010110101001
16–31	0111101101100000
32–47	1110000111011100
48–63	0101000111010110
...	...

- Metodo della lista
 - Puntatore al primo blocco libero
 - Ogni blocco libero ha il puntatore al successivo
 - Blocchi allocati dal primo blocco libero
 - Blocchi cancellati messi alla fine

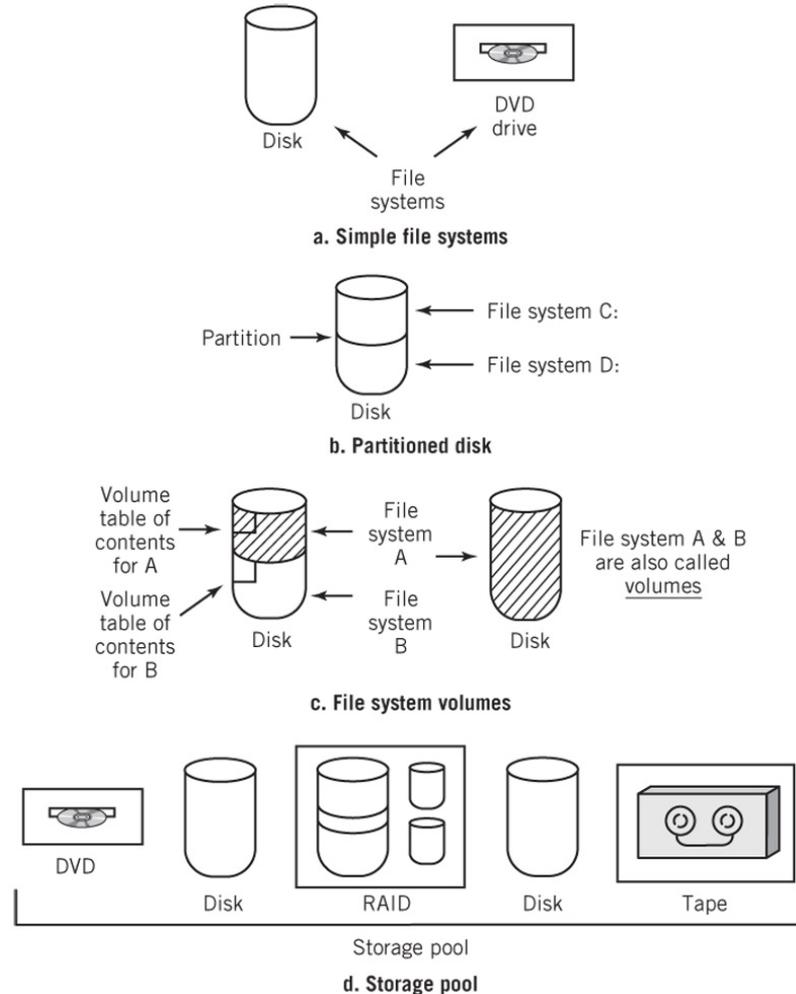


Metodi di allocazione per altri dispositivi di memoria di massa

- Nastri magnetici
 - Non si può creare spazio all'interno del nastro
 - I file che crescono devono essere riscritti
 - I file sono contigui finché possibile
- Dischi ottici
 - Simili ai dischi magnetici
 - ISO-9660 filesystem sola lettura usato per CD-ROM e DVD
 - UDF (Universal Data Format)
 - DVD, BluRay
 - adatto per anche dispositivi ottici riscrivibili



Partizioni, filesystems, volumi





Struttura delle directory

- Fornisce un mezzo organizzativo per trovare facilmente i file
- Contiene spesso gli attributi dei file (come il nome) e
 - Primo blocco del file per allocazione contigua o collegata
 - I-node per allocazione indicizzata

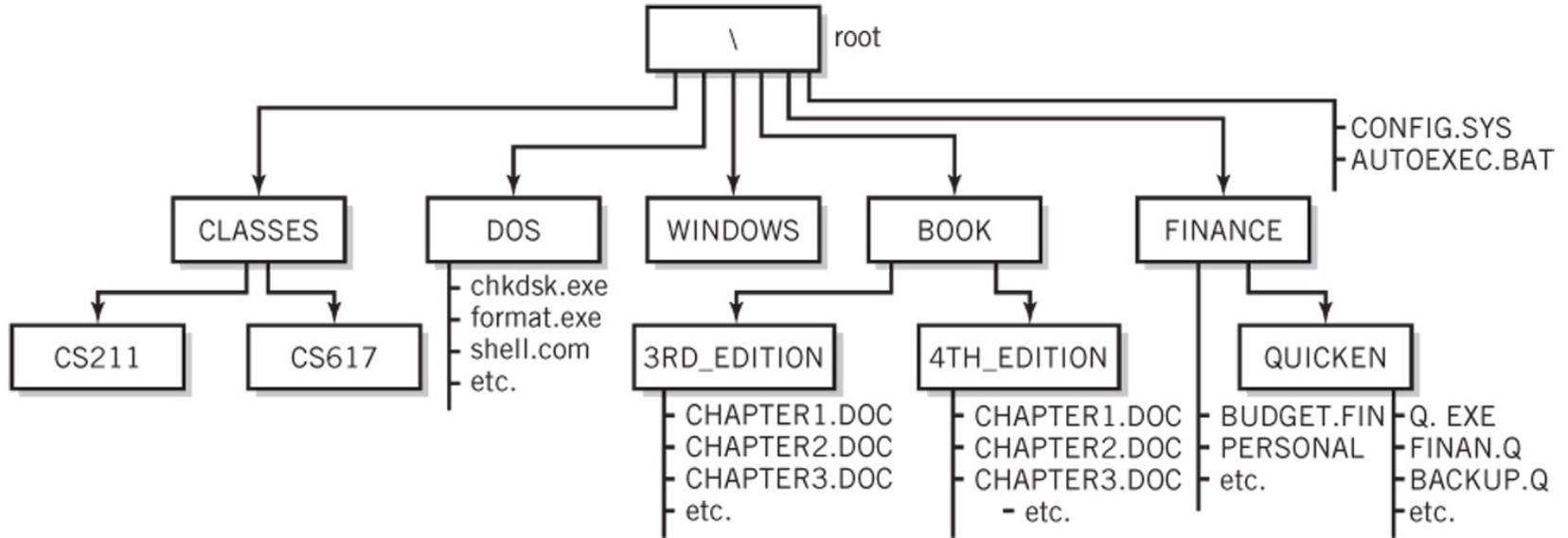


Directory ad albero

- Struttura gerarchica con una singola directory radice da cui derivano tutte le altre
- Tutte le directory e i file hanno dei nomi
- Percorso
 - Assoluto – percorso completo partendo dalla radice
 - Relativo – percorso partendo dalla directory corrente
- Separatore
 - Usato per separare sottodirectori e nomi dei file
 - / UNIX
 - \ DOS, Windows



Directory ad albero



Esempi di percorso:

Assoluto: C:\FINANCE\QUICKEN\Q.EXE

Relativo alla directory FINANCE: QUICKEN\Q.EXE

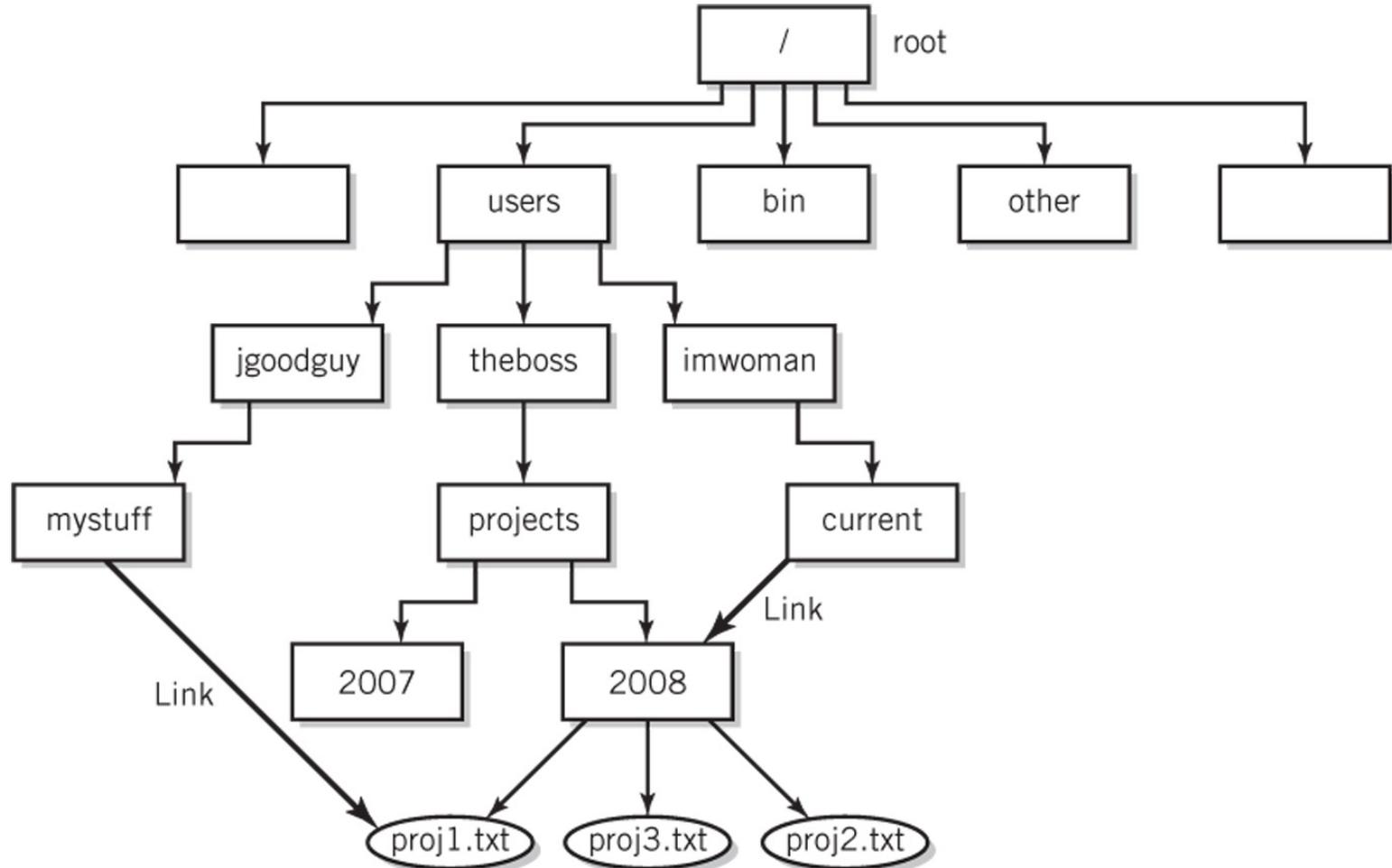


Directory a grafo

- Struttura ad albero con collegamenti tra rami differenti
- Vantaggi
 - Facilita l'accesso degli utenti a dati condivisi
- Svantaggi
 - Cicli e collegamenti a dati non più esistenti
- Esempi:
 - Unix: hard link e link simbolici

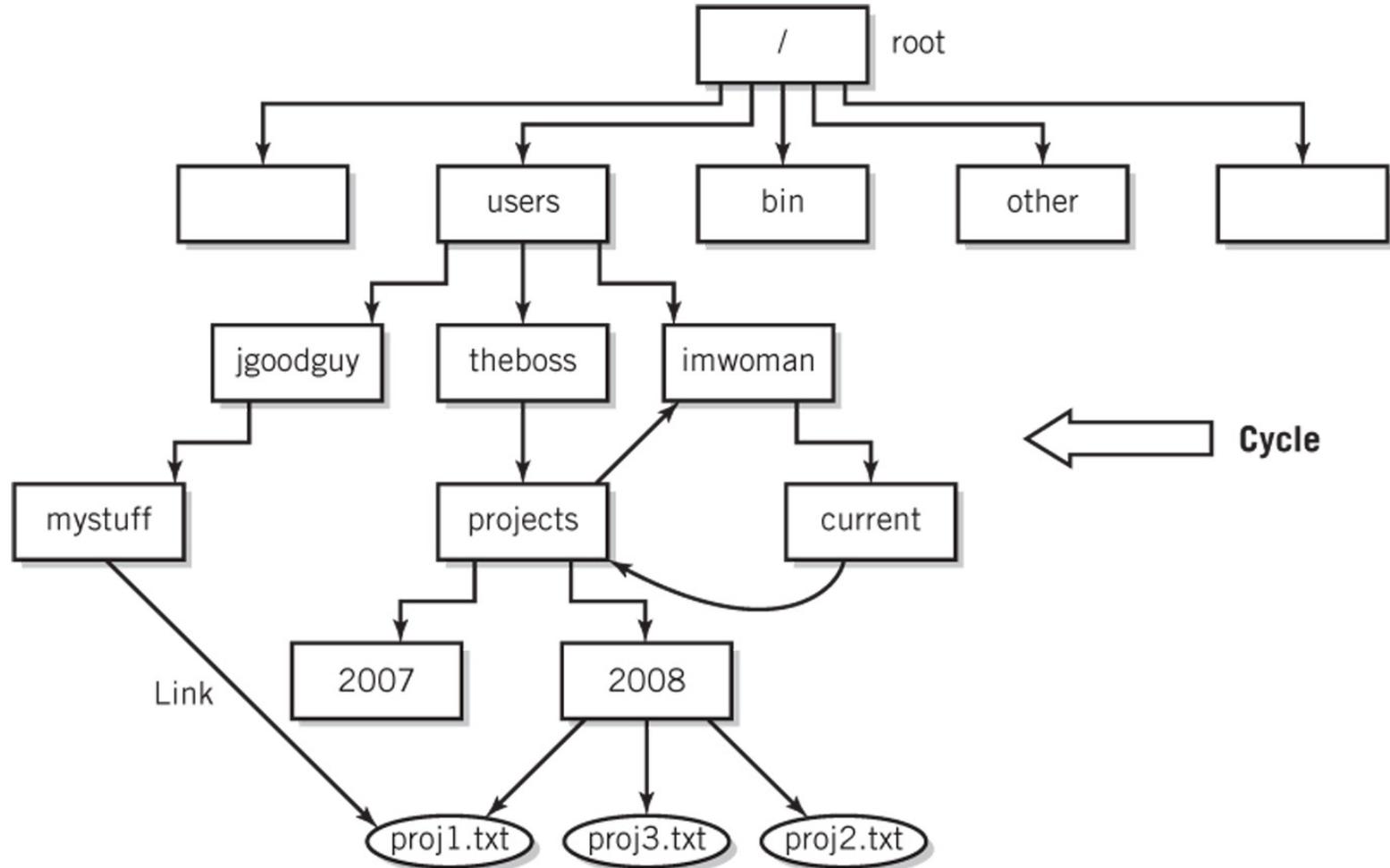


Un directory a grafo aciclico



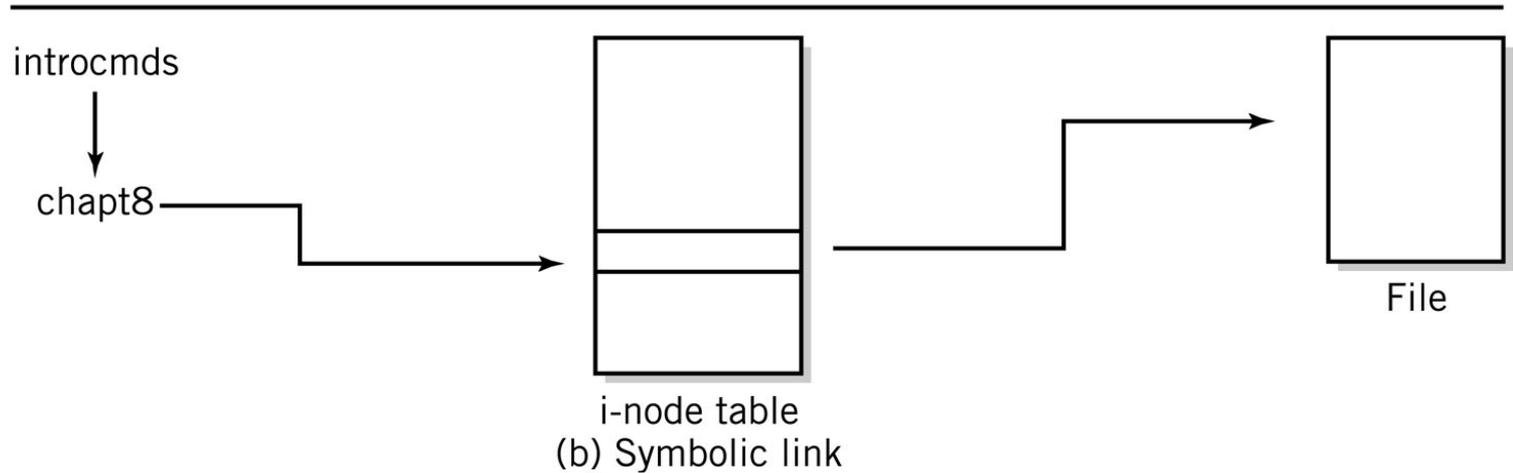
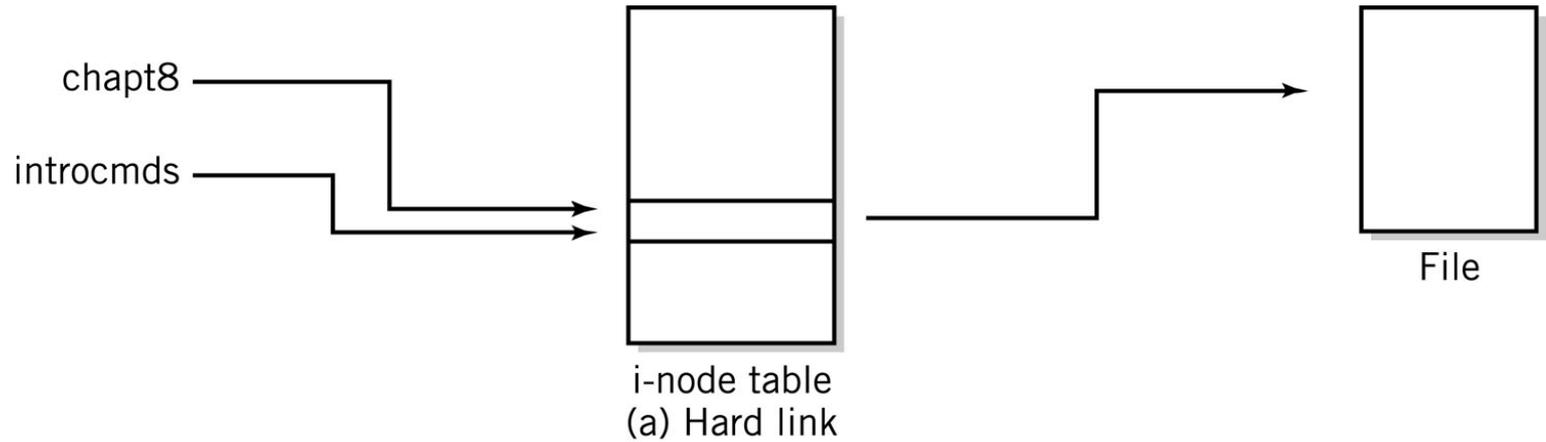


Grafo con un ciclo





Hard Links vs. Symbolic Links



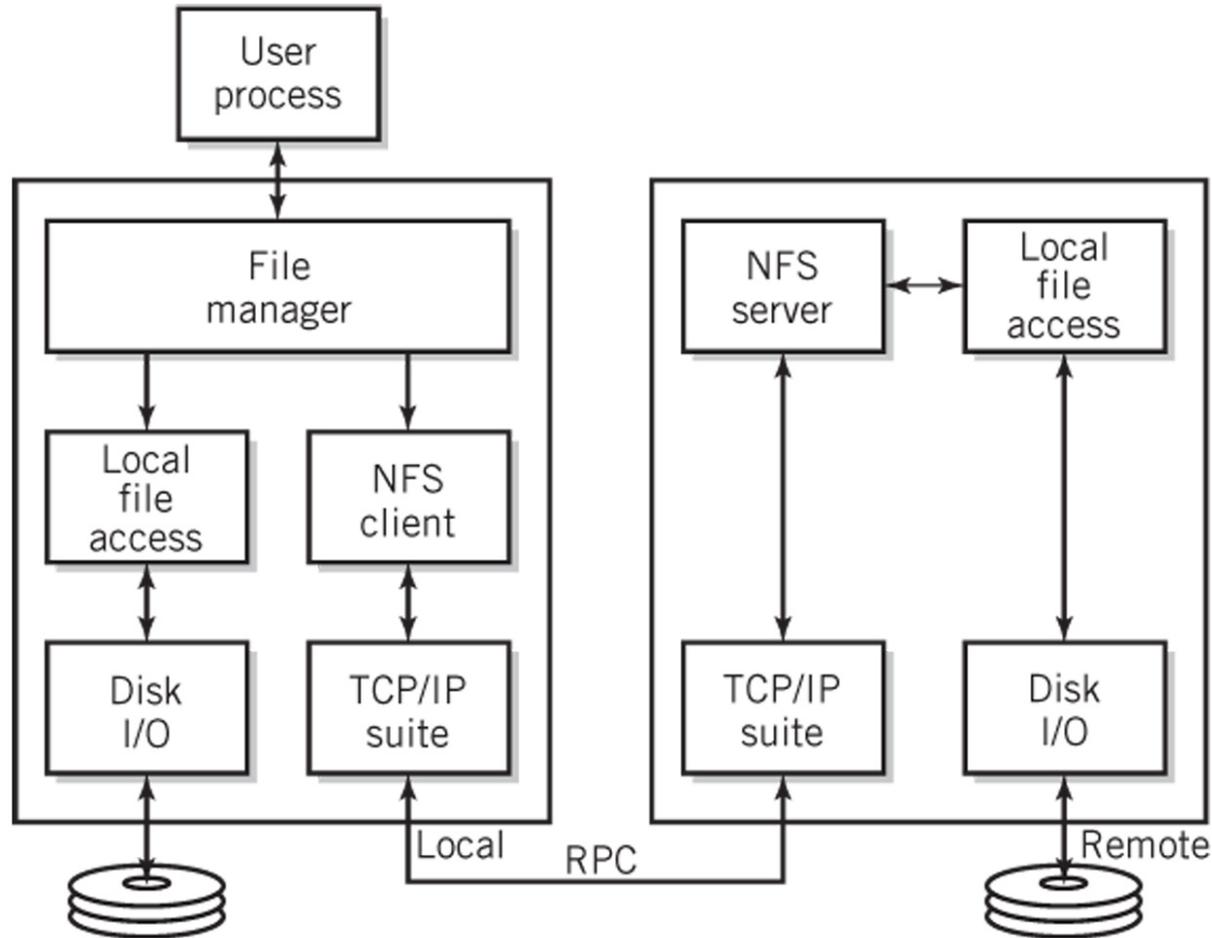


Accesso ai file tramite rete

- FTP
 - File Transfer Protocol
- Network file systems
 - Windows
 - Lettere di dispositivo associate a cartelle remote
 - Common Internet File System (CIFS)
 - UNIX
 - Percorsi associati a cartelle remote
 - Network File System (NFS)



Filesystem di rete: NFS





Protezione dei file

- Possibili diritti:
 - Lettura
 - Scrittura
 - esecuz.
- ACL – access control list
- UNIX/Linux – owner, group, everyone



Directory e sicurezza su Unix/Linux

```
$ls -lF
drwx----- 1 iengland csdept 36005 Feb 15 12:02 bookchapters/
-rw-r--r--  1 iengland csdept  370 Sep 17  1:02 assignment1.txt
--wx--x---  2 iengland csdept 1104 Mar  5 17:35 littleman*
-rwxrwx---  1 iengland csdept 2933 May 22  5:15 airport shell*
drwxr--r--  1 iengland csdept 5343 Dec  3 12:34 class syllabi/
```

ls -lF elenca file in una directory usando un formato lungo
stringa di 10 caratteri per la protezione dei file

1st char d for directory, - for file, s for symbolic link

2nd to 4th char permissions for the owner

5th to 7th char permissions for the group

8th to 10th char permissions for everyone

r - read permission, w - write permission, x - execute permission



Journaling File Systems

- Tiene traccia separatamente di ogni transazione che richiede la modifica dei file (nel journal)
- Due possibilità
 1. Proteggere l'integrità del solo file system
 - Esempio: Windows NTFS file system
 2. Garantire anche l'integrità dei dati
 - Esempi: Linux ext3 and ext4, IBM JFS



Copyright 2013 John Wiley & Sons

All rights reserved. Reproduction or translation of this work beyond that permitted in section 117 of the 1976 United States Copyright Act without express permission of the copyright owner is unlawful. Request for further information should be addressed to the Permissions Department, John Wiley & Sons, Inc. The purchaser may make back-up copies for his/her own use only and not for distribution or resale. The Publisher assumes no responsibility for errors, omissions, or damages caused by the use of these programs or from the use of the information contained herein.