

Directory

Cosa contengono?

Le directory contengono le informazioni sui file che sono:

- attributi
- posizione (dove sono i dati)
- proprietario

Una directory è essa stessa un file (speciale) che fornisce un mapping tra nomi dei file e file stessi

Operazioni su una directory

Su una directory si può effettuare operazioni di:

- ricerca
 - creazione file
 - cancellazione file
 - lista del contenuto della directory
 - modifica della directory
-

Elementi delle directory

Informazioni di base

Le informazioni di base che devono essere da un file system sono:

- Nome del file → nome scelto dal creatore (utente o processo) e unico in una directory data
- Tipo del file → eseguibile, testo, binario, etc.
- Organizzazione del file → per sistemi che supportano diverse possibili organizzazioni (outdated)

Informazioni sull'indirizzo

- Volume → indica il dispositivo su cui il file è memorizzato
- Indirizzo di partenza → dove si trovano le informazioni (dipende fortemente dal file system, ad es. da quale settore o traccia del disco)
- Dimensione attuale → in byte, word o blocchi
- Dimensione allocata → dimensione massima del file

Controllo di accesso

- Proprietario → può concedere/negare i permessi ad altri utenti (leggere/scrivere) e può anche cambiare tali impostazioni
- Informazioni sull'accesso → potrebbe contenere username e password per ogni utente autorizzato (chi può leggere scrivere)
- Azioni permesse → per controllare lettura, scrittura, esecuzione, spedizione tramite rete

Informazioni sull'uso

- Data di creazione
- Identità del creatore
- Data dell'ultimo accesso in lettura
- Data dell'ultimo accesso in scrittura
- Identità dell'ultimo lettore
- Identità dell'ultimo scrittore
- Data dell'ultimo backup
- Uso attuale → lock, azione corrente, etc.

Strutture per le directory

Inizialmente (negli anni '50/'60) il metodo usato per memorizzare le informazioni era quello di fare una lista di entry, una per ogni file in un'unica directory. Ma con il tempo ci si è resi conto che non era più sufficiente per la quantità di file presenti. Si passò quindi ad uno schema a due livelli

Schema a due livello per le directory

Si passò quindi ad un sistema con una directory per ogni utente, più una *master* che le contiene (la master contiene anche l'indirizzo e le informazioni per il controllo

dell'accesso).

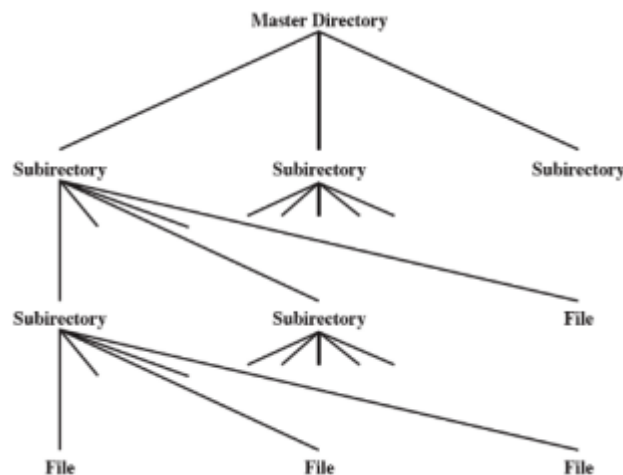
Ogni directory utente è solo una lista dei file di quell'utente (non offre una struttura per insiemi di files)

Schema gerarchico ad albero per le directory

Infine si arrivò ad uno schema gerarchico ad albero. Qui è presente una directory *master* che contiene tutte le directory utente e ogni directory utente può contenere file oppure altre directory utente.

Ci sono anche sottodirectory di sistema sempre dentro la directory master

Viene quindi permesso per la prima volta di creare directory ad un utente

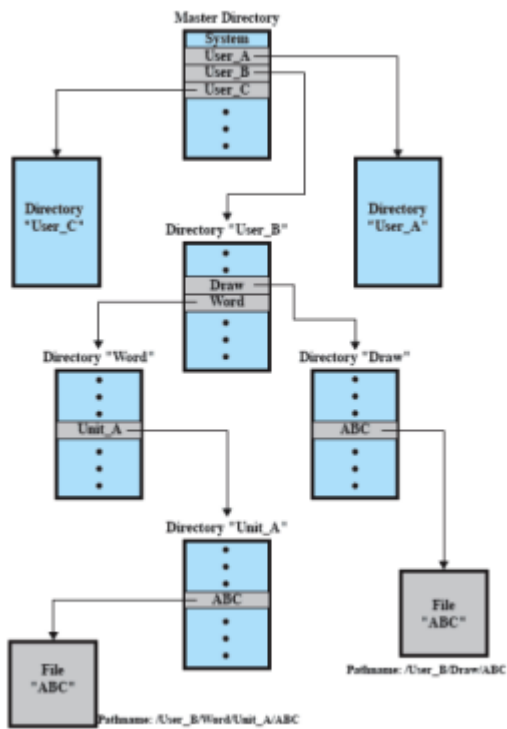


Nomi

Gli utenti devono potersi riferire ad un file usando solo il loro nome, il quale deve essere unico per ogni directory (si possono creare file con nomi uguali ma devono trovarsi in directory diverse), ma un utente può non aver accesso a tutti i file dell'intero sistema

La struttura ad albero permette di trovare un file seguendo un percorso ad albero (*directory path*)

Esempio



Directory di lavoro

Dover dare ogni volta il path completo prima del nome del file può essere lungo e noioso. Solitamente, gli utenti o i processi interattivi, hanno associata una *working directory*, in cui tutti i nomi di file sono dati relativamente a questa directory (rimane possibile dare esplicitamente l'intero percorso se necessario)