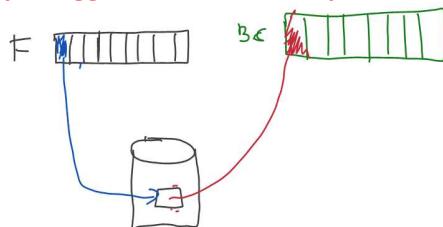


# Virtual file system 4 Caching

martedì 4 novembre 2025 10:05

Andiamo ora a vedere un'alternativa valida all'allocazione dei blocchi : dal file system attraverso una system call i blocchi vengono caricati in ram , bypassando ogni volta l'interazione con i dispositivi. Quindi i blocchi vengono caricati nella **buffer cache** (area di memoria dove i blocchi vengono caricati) ed il sistema operativo legge/scrive in questa area (informazioni nuovamente accessibili ). Poi a tempo debito il sistema scriverà i blocchi nel file system. Quindi con questa tecnica si ha un miglioramento della latenza ed una riduzione di carico sugli hard drive (diminuiscono le interazioni con l'hard drive) . Quindi si arriva ad un meccanismo di **lettura anticipata/scrittura ritardata** : si parla di lettura anticipata quando leggo un record dal dispositivo di massa , devo prendere tutto il blocco e portarlo in buffer cache ; analogamente per la scrittura : i record già presenti in buffer cache una volta modificati vengono scritti nell'hard drive. **Quindi con questa tecnica si ha un passaggio da i/o bound a cpu bound** . In dettaglio :



Vediamo ora come avviene la sostituzione dei blocchi :

## 1. Least recently used

- Il blocco sostituito del buffer cache è l'ultimo usato
- Si usa una struttura a stack
- Si va a vedere a ritroso ultimo blocco usato
- Quindi attenzione a come e quando la modifichiamo

## 2. Least frequently used

- Si mantiene un contatore di riferimenti al blocco indicante il numero di accessi
- Ma attenzione al "carico di lavoro": potrebbe essere che un blocco ci ho lavorato tanto ma tanto tempo fa ed un altro che ci ho lavorato poco ma ultimamente
- Non si capisce la variazione di località

Per ovviare al problema della variazione di località vediamo due soluzioni :

## 1. Buffer cache a due sezioni

- 
- The diagram shows a horizontal line divided into three sections: 'Sezione nuova' (new section), 'Sezione vecchia' (old section), and another unnamed section. In the 'Sezione nuova', there are two yellow squares. In the unnamed section, there is one green square. Arrows point from the text labels to their corresponding squares. A dotted arrow labeled 'passaggio di sezione + incremento del contatore' points from the boundary between the new and old sections to the green square. Another arrow labeled 'riferimento' points to the green square.
- riferimento (nessun effetto)
  - Il contatore viene incrementato se e solo se il blocco viene portato nella sezione nuova (acceduto dalle applicazioni)
  - Possibile spostare blocchi da sezione nuova a sezione vecchia , magari per la saturazione della zona vecchia : portiamo l'elemento che ha il contatore minore
  - Migliora così la località degli accessi

## 2. Buffer cache a tre sezioni

- 
- The diagram shows a horizontal line divided into three sections: 'Sezione nuova', 'Sezione intermedia', and 'Sezione vecchia'. In the 'Sezione nuova', there are two yellow squares. In the 'Sezione intermedia', there is one blue square. In the 'Sezione vecchia', there are two green squares. A legend at the top right indicates that a grey arrow points to 'Sezioni in cui avviene l'incremento dei contatori' (sections where counters are incremented). Dotted arrows labeled 'sostituzione' point from the boundaries between the sections to the blue square in the middle and the green squares in the rightmost section. An arrow labeled 'uscita' points to the rightmost boundary.
- Sezioni in cui avviene l'incremento dei contatori
  - Il passaggio per la sezione intermedia è obbligatorio (da nuova a vecchia) , avendo così il passaggio da intermedia a vecchia per effetto della politica di sostituzione
  - Difficile che un blocco nella sezione nuova verrà escluso dal buffer

Vediamo ora la relazione tra le operazioni di i/o e lo swapping (eseguo i/o dei dati o meglio dell'address space di un'applicazione) , andando a vedere come fare :

**1. File system**

- a. Scarico contenuto dell'address space all'interno di un file (funzioni del file system)
- b. Ma attenzione all'inefficienza (trade off spazio-tempo)
- c. Taglia dell'area non è predefinita

**2. Partizione privata**

- a. Usata in linux
- b. Il software gestisce appositamente l'area di swapping
- c. La disposizione del dispositivo può essere tale da ottimizzare l'accesso in termini di velocità
- d. Ma così si ha frammentazione interna
- e. Taglia predefinita