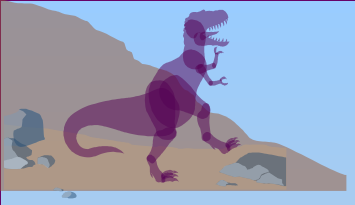
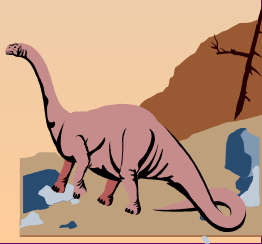


Capitolo 9: Memoria Virtuale

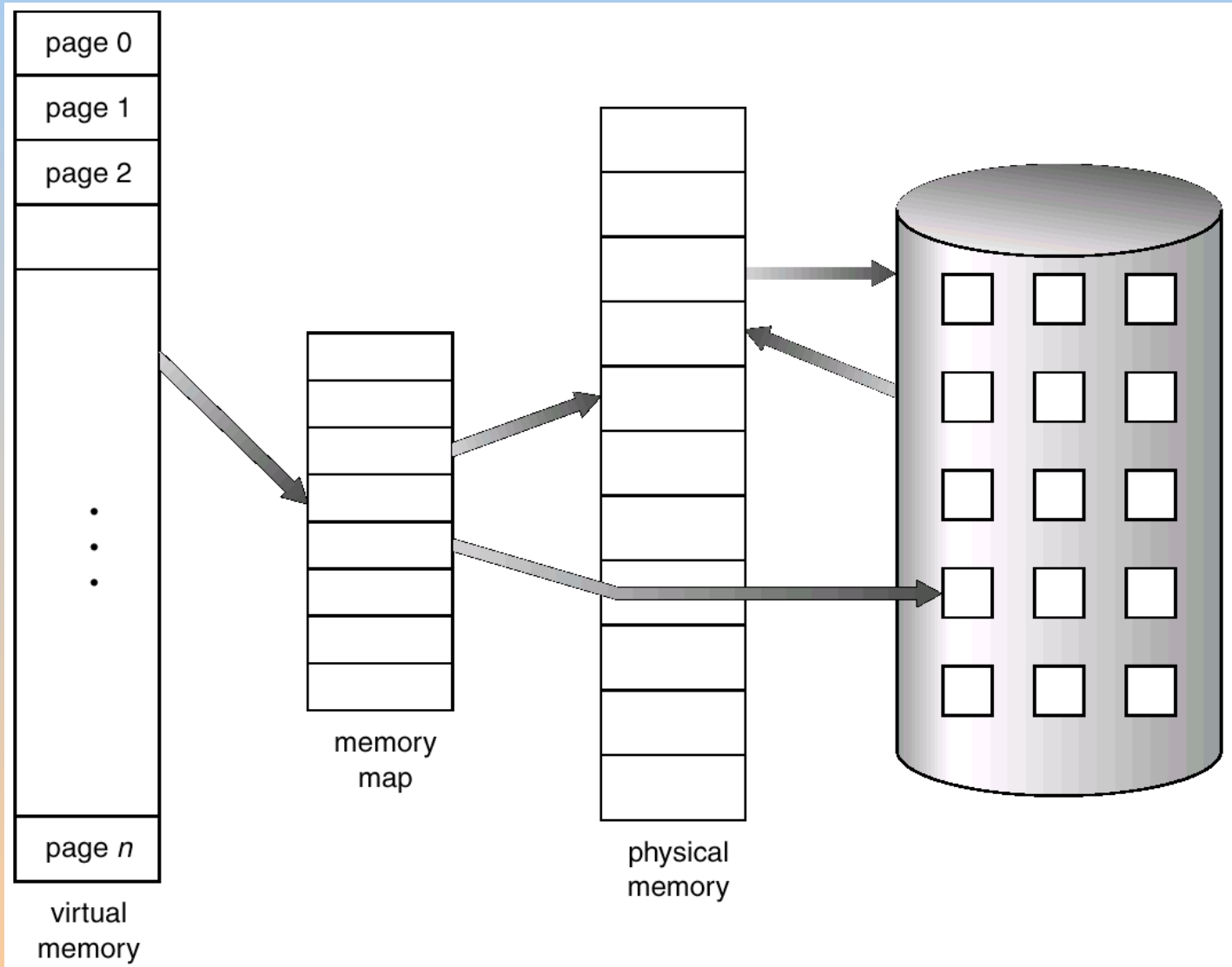
- Introduzione
- Paginazione su richiesta
- Copiatura su scrittura
- Sostituzione delle pagine
- Allocazione dei frame
- Attività di paginazione degenere (trashing)
- File mappati in memoria
- Allocazione di memoria del kernel
- Altre considerazioni
- Esempi tra i sistemi operativi



Introduzione

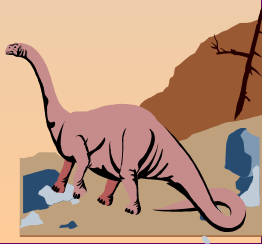
- La memoria virtuale e' una tecnica che permette di eseguire processi che possono anche non risiedere completamente in memoria.
- Permette la separazione della memoria logica dell'utente dalla memoria fisica:
  - Solo parte del programma viene caricato in memoria per essere eseguito.
  - Lo spazio di indirizzi logici può quindi essere più grande dello spazio degli indirizzi fisici.
  - Permette di condividere lo stesso spazio indirizzi tra più processi.
  - Permette tecniche più efficienti di creazione di processi.
- La memoria virtuale puo' essere implementata tramite:
  - **paginazione su richiesta** (anziche' effettuare swapping sulla memoria di tutto il processo, si utilizza uno swapper *pigro*, che non riversa mai una pagina in memoria a meno che non sia necessaria - *pager*)
  - **segmentazione su richiesta**

Schema di memoria virtuale più grande della memoria fisica.

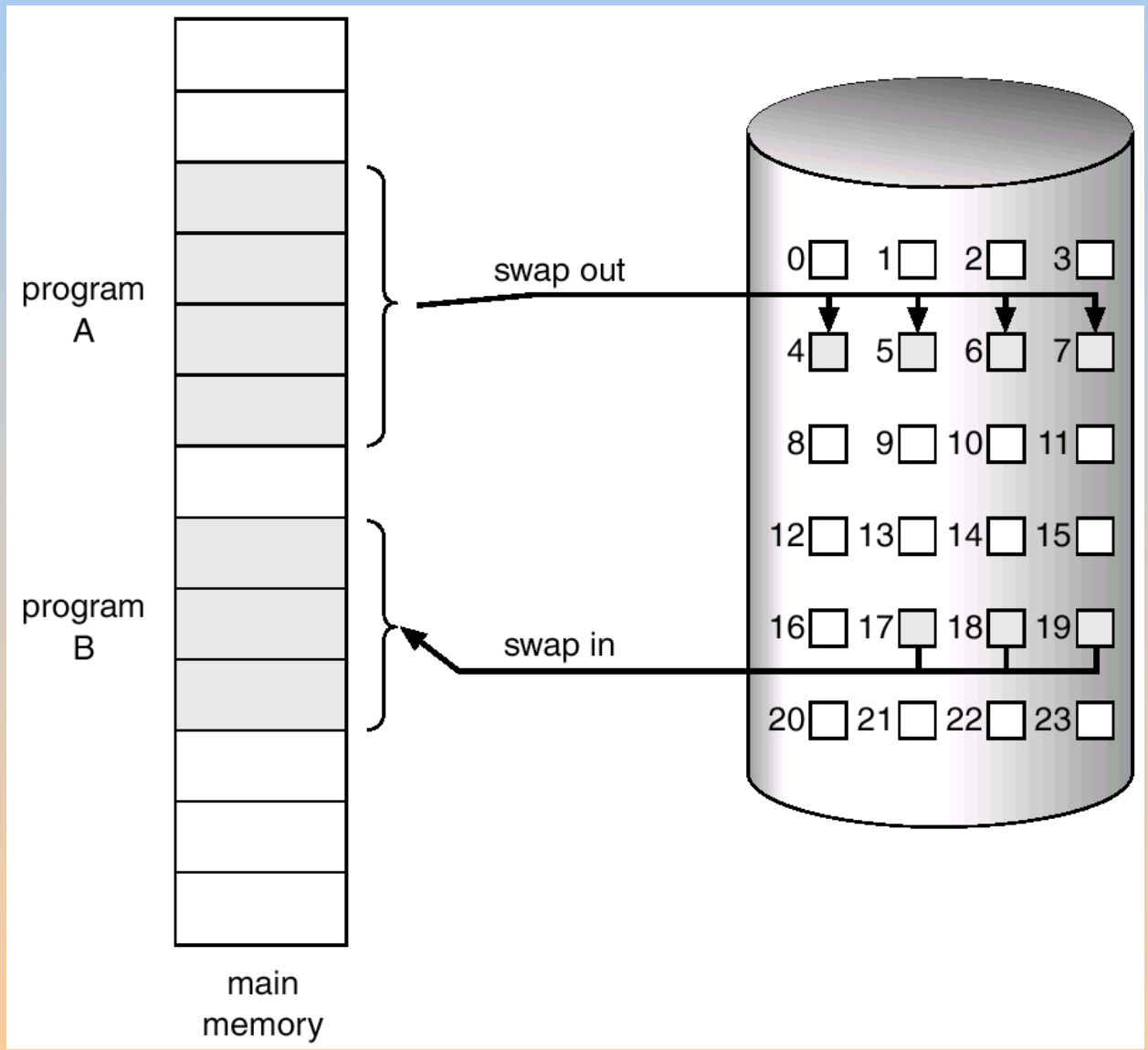


Paginazione su richiesta

- La maggior parte dei sistemi operativi utilizza il metodo della paginazione su richiesta.
- Anziché caricare in memoria l'intero processo, come nella paginazione, si carica una pagina in memoria solo quando essa e' necessaria.
  - Per caricare un programma utente in memoria è necessario un minore tempo di I/O (aumentando la velocita' di esecuzione).
  - Nessun vincolo sulla quantita' di memoria fisica necessaria ad un processo.
  - Aumento del grado di multiprogrammazione grazie al minore utilizzo della memoria fisica.
- Una pagina e' necessaria.⇒ c'è stato un riferimento ad essa.
- Se il processo tenta di accedere ad una pagina che non era stata caricata nella memoria, il sistema genera una *eccezione di pagina mancante* (page fault trap).
  - Il riferimento non era valido ⇒ si termina il processo (abort)
  - Il riferimento era valido ⇒ si porta la pagina in memoria

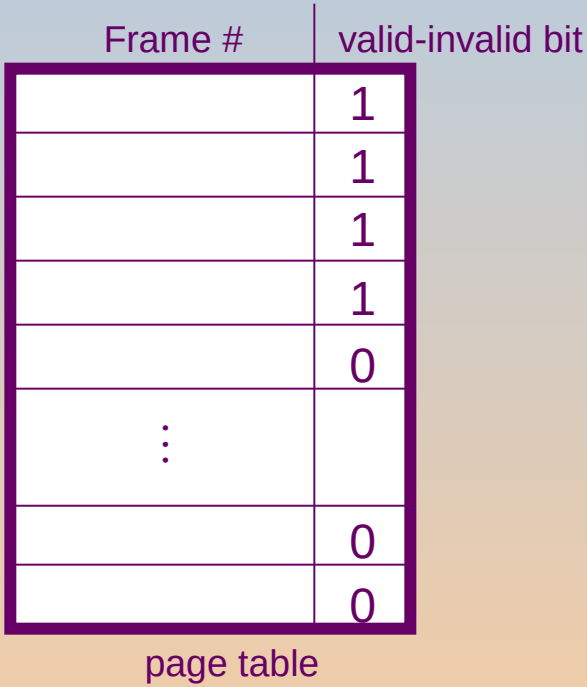


Trasferimento di una memoria paginata nello spazio contiguo di un disco



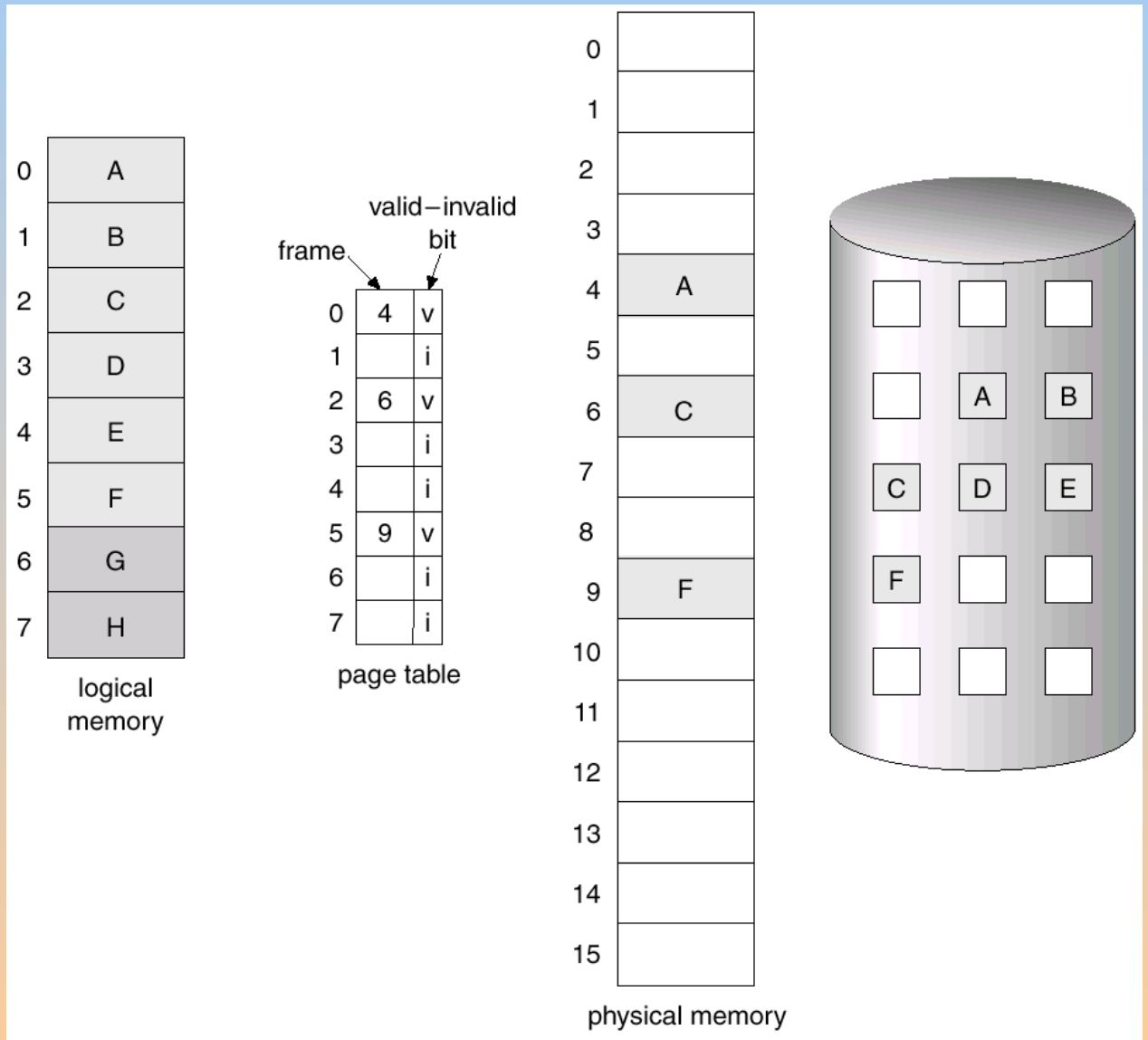
Bit di validità

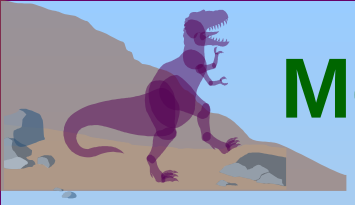
- A ciascuna pagina è associato un bit di validità (1 ⇒ in-memoria, 0 ⇒ non in memoria), inizializzato in partenza a 0.
- Esempio di tabella delle pagine con bit di validità:.



- Se il processo tenta di accedere ad una pagina che non era stata caricata in memoria (bit di validità 0) ⇒ page fault.

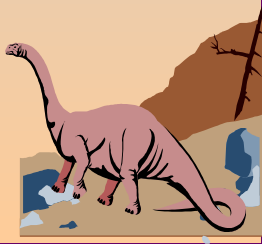
Tabella delle pagine quando alcune pagine non si trovano nella memoria centrale.





Meccanismi di ausilio alla paginazione su richiesta

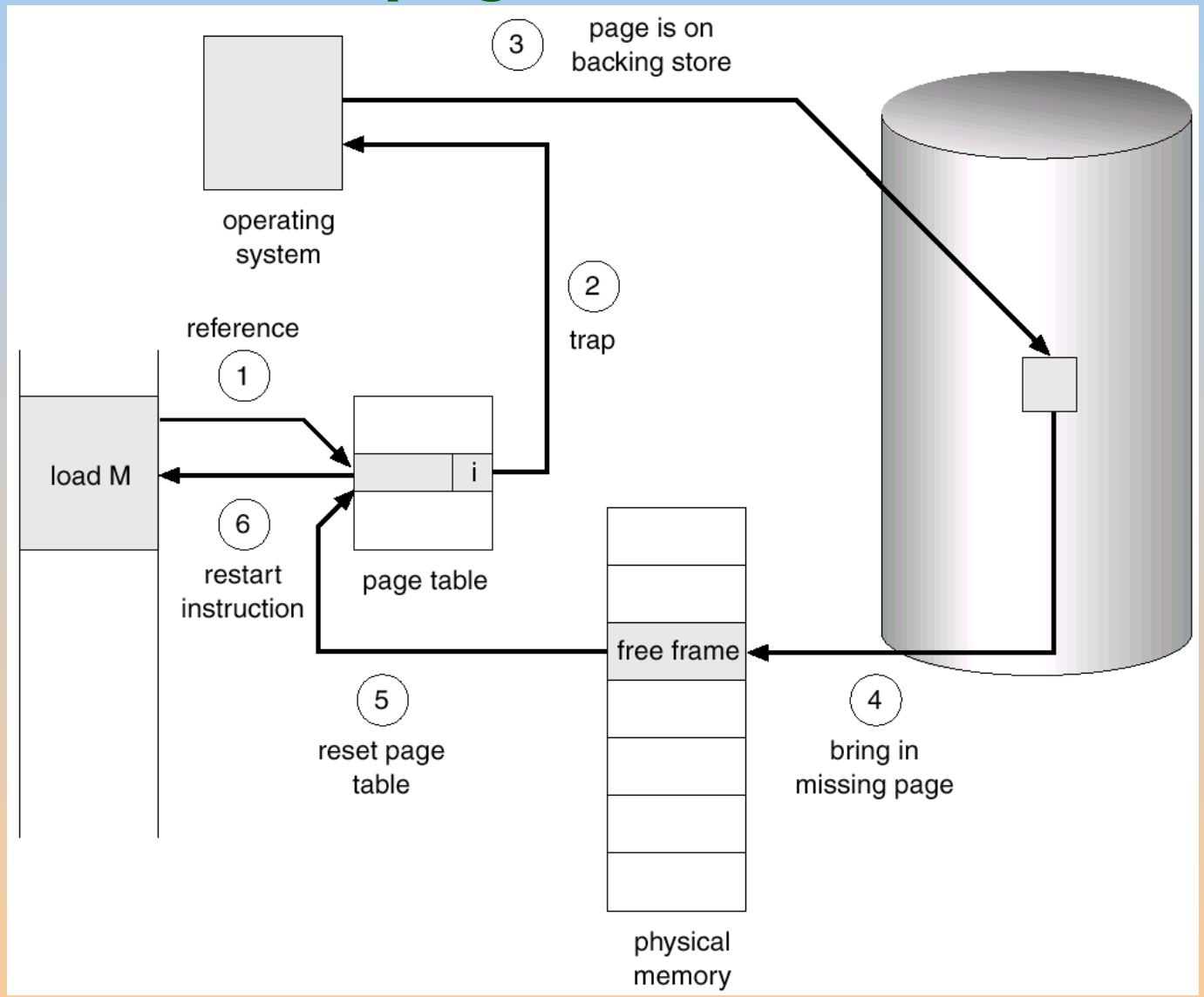
- *Tabella delle pagine*: questa tabella ha la capacita' di contrassegnare un elemento come non valido attraverso un bit di validità oppure un valore speciale dei bit di protezione.
- *Memoria ausiliaria* : in generale il sistema operativo utilizza un file system ad alta velocita' per la partizione di swap.
- Ad esempio il sistema operativo Linux utilizza una partizione distinta da quella del file system convenzionale in modo da velocizzare la gestione dei dati.
- La paginazione su richiesta puo' avere un effetto significativo sulle prestazioni generali di un sistema di calcolo.



Gestione del page fault

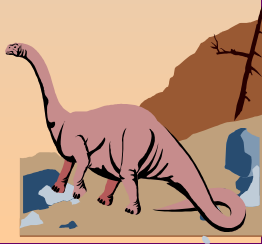
- Quando viene referenziata una pagina non in memoria, ci sarà un segnale di errore $\Rightarrow$ page fault
- La procedura di gestione del page fault è suddivisa in sei fasi:
  1. Si controlla la tabella interna al processo per stabilire se il riferimento alla pagina e' valido o non valido.
  2. Se il riferimento non e' valido si termina il processo, altrimenti se e' valido ma la pagina non e' presente in memoria, se ne effettua l'inserimento.
  3. Viene individuato un frame libero, per esempio prelevandolo dall'elenco dei blocchi liberi.
  4. Si programma un'operazione sui dischi per trasferire la pagina desiderata nel blocco di memoria appena assegnato (swap in).
  5. A swap in terminato, si modifica la tabella interna conservata con il processo e la tabella delle pagine per indicare che la pagina e' disponibile nella memoria centrale.
  6. Si riavvia l' istruzione interrotta dal segnale di eccezione di indirizzo illegale. Il processo può accedere alla pagina.
- E' addirittura possibile avviare l'esecuzione di un processo senza caricare inizialmente alcuna pagina in memoria.

Fasi di gestione di un'eccezione di pagina mancante



Che succede se non c’è un frame libero?

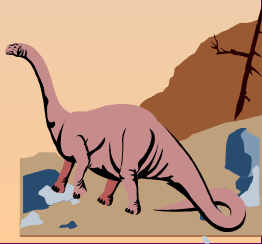
- Rimpiazzamento di pagina – si cerca una pagina non utilizzata tra quelle in memoria, e la si memorizza nella memoria ausiliaria (swap out).
  - algoritmo
  - prestazioni – l’ algoritmo dovrà minimizzare il numero di page fault futuri.
- Quindi alcune pagine dovranno essere portate in memoria più volte.



Prestazioni della paginazione su richiesta

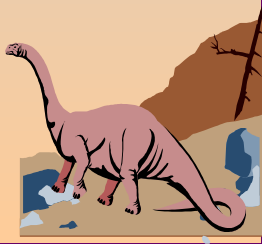
- La paginazione su richiesta può avere un effetto rilevante sulle prestazioni di un calcolatore.
- Tempo di accesso alla memoria $\Rightarrow$ *ma* (normalmente varia da 10 a 200 nanosecondi finché non si verifica un page fault)
- Probabilità di una assenza di pagina (page fault rate): $0 \leq p \leq 1.0$
  - se $p = 0$ nessun page fault
  - se $p = 1$, ogni riferimento genera un page fault
- Tempo di accesso effettivo (Effective Access Time - EAT)

  EAT = ((1 – *p*) x *ma*) + (*p* x tempo di gestione dell'assenza di pagina)
- Per calcolare il tempo d'accesso effettivo occorre conoscere il tempo necessario alla gestione di un'assenza di pagina.



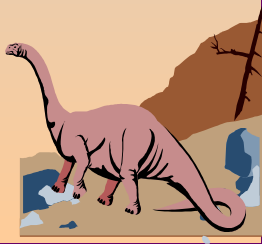
Gestione di una assenza di pagina

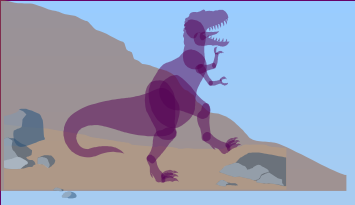
1. Segnale di eccezione al sistema operativo.
2. Salvataggio dei registri utente e dello stato del processo.
3. Determinazione della natura di eccezione di pagina mancante del segnale di eccezione.
4. Controllo della correttezza del riferimento alla pagina e determinazione della locazione della pagina nel disco
5. Lettura dal disco e trasferimento in un blocco di memoria libero:
   a) attesa nella coda del dispositivo fino a che la richiesta di lettura non è servita
   b) attesa del tempo di posizionamento e latenza del dispositivo
   c) inizio del trasferimento della pagina in un blocco di memoria libero
   d) durante l'attesa, assegnazione della CPU ad un altro utente
   e) segnale di interruzione emesso dal controllore del disco (I/O completato)
   f) salvataggio dei registri e dello stato dell'altro processo utente



Gestione di una assenza di pagina (II)

6. Determinazione della provenienza dal disco dell'interruzione.
7. Aggiornamento della tabella delle pagine e di altre tabelle per segnalare che la pagina richiesta è in memoria.Attesa che la CPU sia nuovamente assegnata a questo processo.
8. Recupero dei registri utente, dello stato del processo e della nuova tabella delle pagine, quindi ripresa dell'istruzione interrotta.





Prestazioni della paginazione su richiesta: un esempio

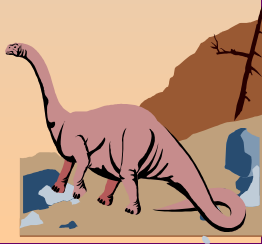
- Considerando un tempo medio di servizio dell'eccezione di pagina mancante di 25 millisecondi e un tempo di accesso alla memoria di 100 nanosecondi, il tempo effettivo d'accesso in nanosecondi è

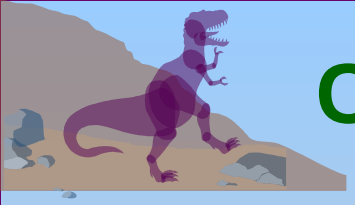
  $EAT = (1 - p) \times 100 + p \times (25 \text{ millisecondi}) =$

  $= (1 - p) \times 100 + p \times 25.000.000 =$

  $= 100 + 24.999.900 \times p$

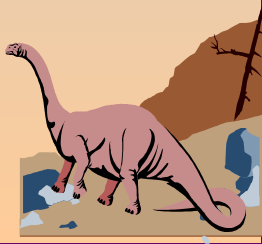
- Il tempo d'accesso effettivo è direttamente proporzionale alla frequenza delle assenze di pagina.
- Ad es. se un accesso su 1000 accusa un'assenza di pagina, il tempo d'accesso effettivo è di 25 microsecondi.
- Se si desidera un rallentamento inferiore al 10%: $p < 0{,}0000004$.
- Cioè per mantenere a un livello ragionevole il rallentamento dovuto alla paginazione, si può permettere un'assenza di pagina ogni 2.500.000 accessi alla memoria.





Creazione ed esecuzione di processi

- La memoria virtuale offre altri vantaggi per quel che riguarda la creazione ed esecuzione dei processi.
- Copiatura su scrittura (Copy-on-Write):
  - Questa tecnica permette una creazione dei processi molto rapida e minimizza il numero di pagine che si devono assegnare al nuovo processo
- Associazione dei file alla memoria (Memory-Mapped Files):
  - Questa tecnica semplifica e velocizza gli accessi al file system trattando l'I/O su file tramite la paginazione in memoria centrale, invece che tramite le chiamate a sistema di **read()** e **write()**.

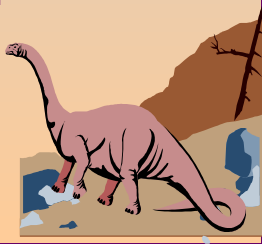


Copiatura su scrittura (copy-on-write)

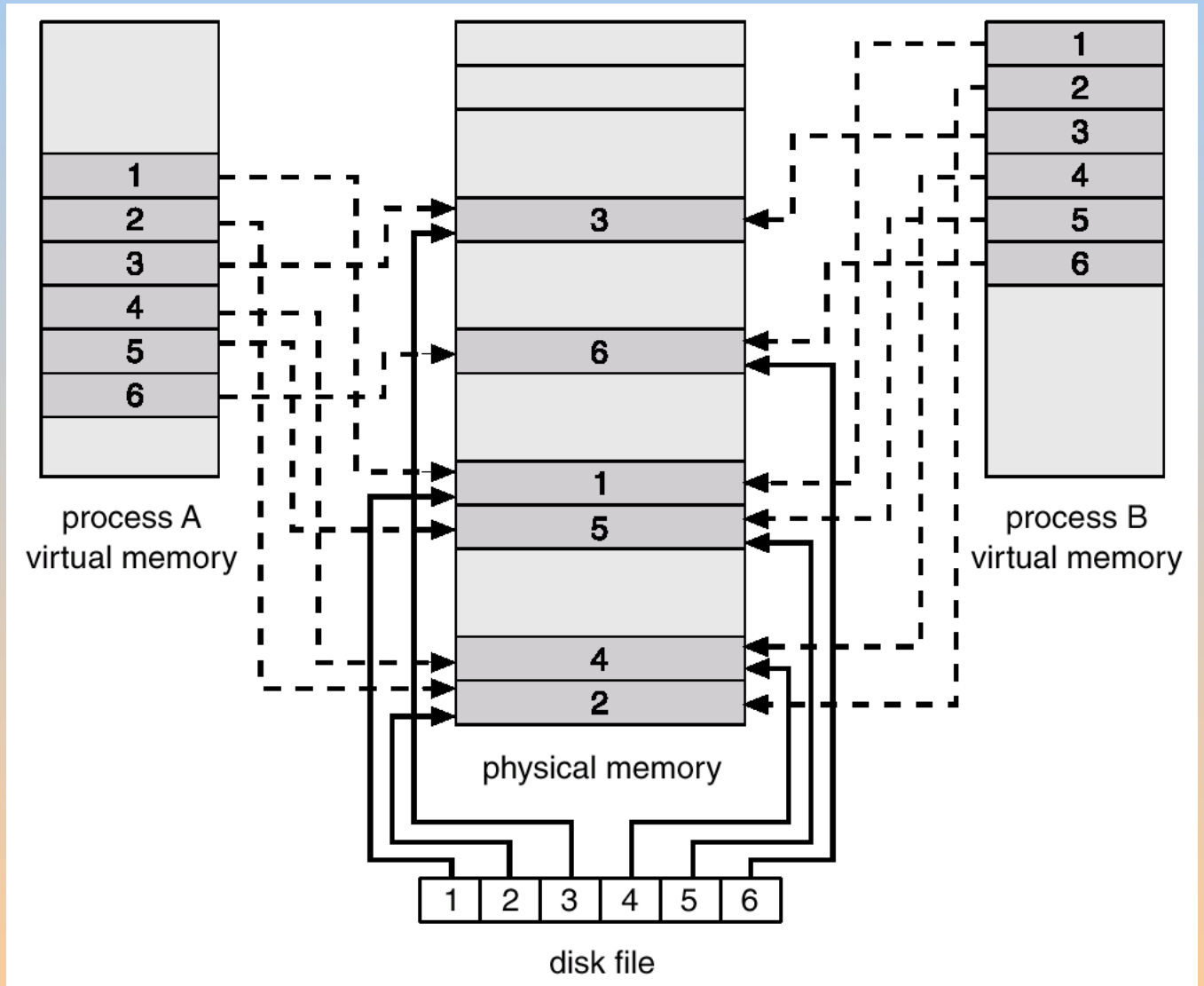
- *fork* crea un processo figlio come duplicato del genitore.
- Nella versione originale *fork* creava per il figlio una copia dello spazio di indirizzi del genitore, duplicando le pagine appartenenti al processo genitore.
- Considerando che spesso ad una *fork* segue una *exec*, questa copiatura risulta inutile.
- La tecnica di *copiatura su scrittura* (*Copy-on-Write - COW*) permette sia al processo genitore che al processo figlio di condividere inzialmente le stesse pagine di memoria.
- Se uno dei due processi modifica una pagina, solo allora la pagina è duplicata.
- COW permette una creazione di processi più efficiente e rapida.
- Quando una pagina deve essere duplicata, si attinge ad un pool di pagine libere utilizzate anche per gestire l'ingrandimento di strutture come pile o *heap* di un processo, e che vengono prima riempite di zeri e poi assegnate (*azzeramento su richiesta*)

Associazione dei file alla memoria

- Se consideriamo un accesso sequenziale in lettura o scrittura ad un file su disco, ogni operazione di lettura o di scrittura necessiterà di una chiamata al sistema e di un accesso al disco.
- L' **associazione alla memoria** permette di trattare l' I/O di file su disco come un generico accesso a memoria facendo corrispondere un blocco del disco ad una pagina (o più pagine) di memoria.
- L'accesso a file avviene inizialmente tramite l'ordinario meccanismo di paginazione su richiesta, cui segue, però, il caricamento in una pagina fisica di una porzione di file della dimensione di una pagina, leggendola dal file system.
- Le operazioni successive di lettura o scrittura si gestiscono come normali accessi alla memoria.
- Questa tecnica pemette inoltre l'associazione dello stesso file alla memoria virtuale di più processi allo scopo di condividere dati.
- Le chiamate a sistema per l'associazione alla memoria possono anche disporre della funzione di copiatura su scrittura, permettendo ai processi di condividere un file per la lettura, mantenendo però una propria copia dei dati da essi modificati.



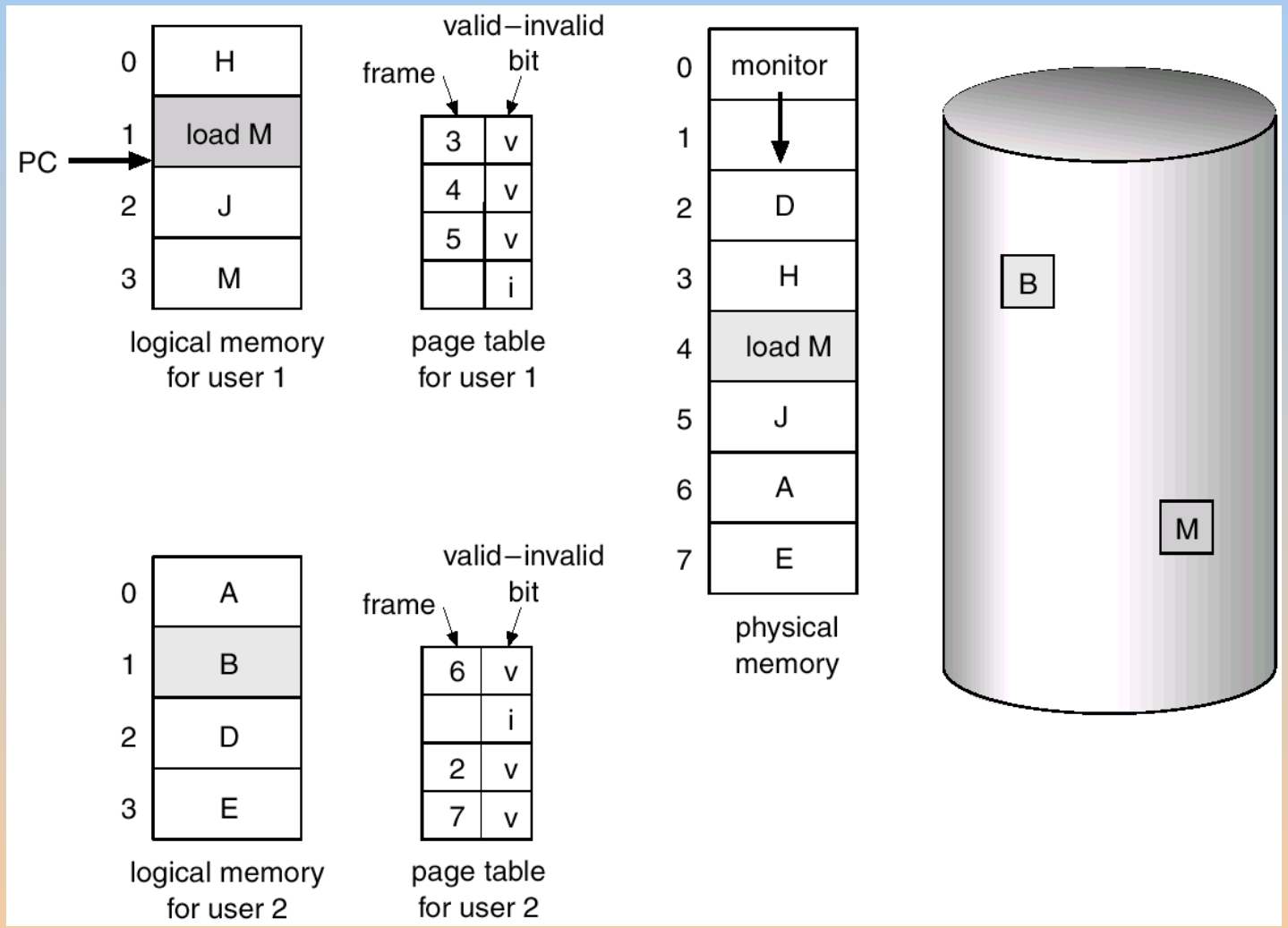
Associazione dei file alla memoria: un esempio

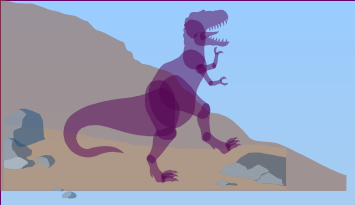


Sostituzione delle pagine

- In presenza di una page fault dovuto ad un riferimento ad una pagina non in memoria, quindi, si dovrà portare la pagina in memoria.
- Se nessun blocco di memoria e' libero, si potrà liberarne uno attualmente inutilizzato, scrivendo il suo contenuto nell'aria di avvicendamento e modificando la tabella delle pagine (e le altre tabelle) per indicare che quella pagina non è più in memoria.
- Il blocco di memoria liberato si potrà usare per caricare la pagina che ha causato l'eccezione.
- La procedura di servizio dell'eccezione di pagina mancante verrà modificata in modo da includere l'eventuale sostituzione della pagina.
- L'architettura fisica del sistema di calcolo può disporre di un *bit di modifica* (*modify - dirty bit*), associato ad ogni pagina che mette a 1 automaticamente ogni volta che la pagina viene modificata.
- Solo le pagine che sono state modificate vengono riscritte su disco.
- La sostituzione delle pagine completa la separazione tra memoria logica e memoria fisica: possiamo avere una memoria virtuale molto ampia con una memoria fisica più piccola.

Necessità di sostituzione di pagine





Schema di base

- La procedura di servizio dell'eccezione di pagina mancante deve modificata in modo da includere l'eventuale sostituzione della pagina:
  1- Si individua la locazione nel disco della pagina richiesta
  2- Si cerca un blocco di memoria libero:
     a) se esiste, lo si usa;
     b) altrimenti si impiega un algoritmo di sostituzione delle pagine per scegliere un blocco di memoria "vittima";
     c) si scrive la pagina "vittima" nel disco; si modificano adeguatamente le tabelle delle pagine e dei blocchi di memoria;
  3- Si scrive la pagina richiesta nel blocco di memoria appena liberato e si modificano le tabelle delle pagine e dei blocchi di memoria;
  4- Si riavvia il processo utente.
- Esistono diversi algoritmi usati per la sostituzione delle pagine.
- Un algoritmo di solito viene valutato effettuandone l'esecuzione su di una particolare stringa di riferimenti e calcolando il numero di page fault che esso causa.

Sostituzione di una pagina