

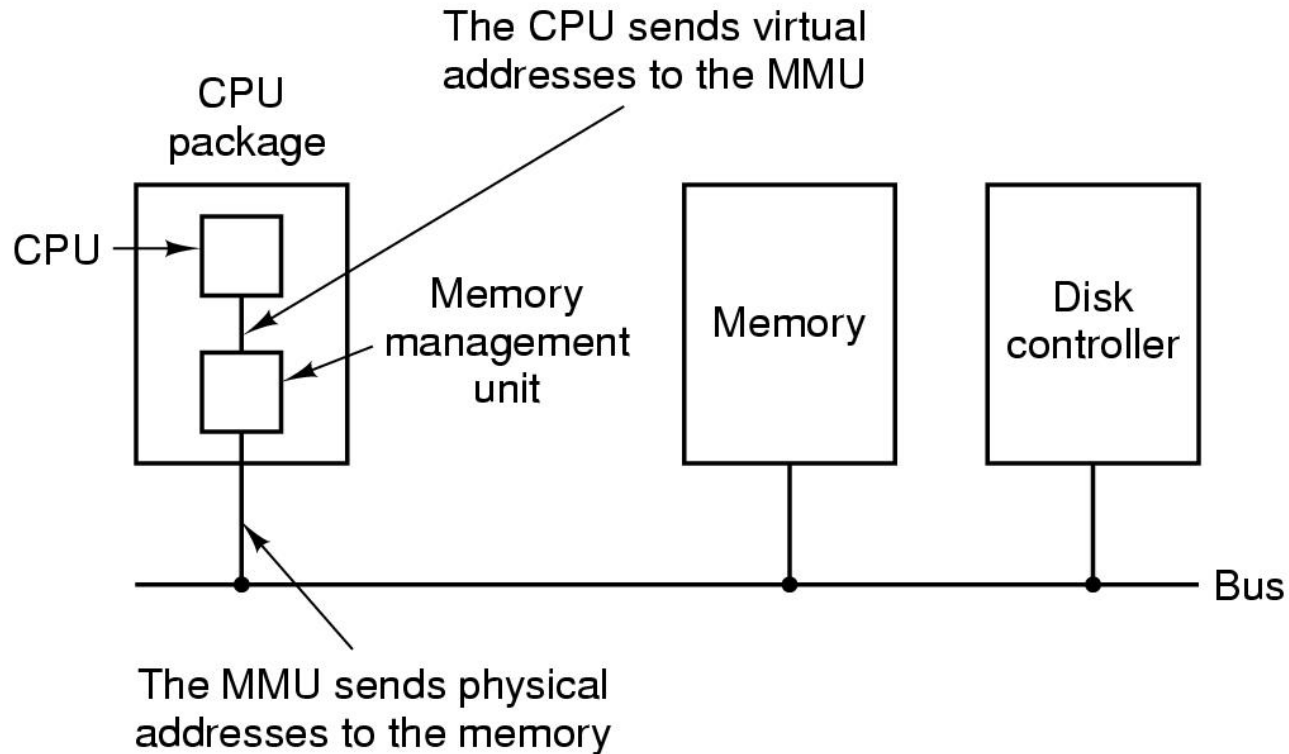
# Capitolo 4

## Gestione della Memoria

- 4.1 Introduzione alla gestione della memoria
- 4.2 Swapping
- 4.3 Memoria virtuale
- 4.4 Implementazione
- 4.5 Algoritmi di sostituzione
- 4.6 Criteri di progetto per la paginazione
- 4.7 Case study: Unix
- 4.8 Case study: Windows 2000

# Memoria Virtuale

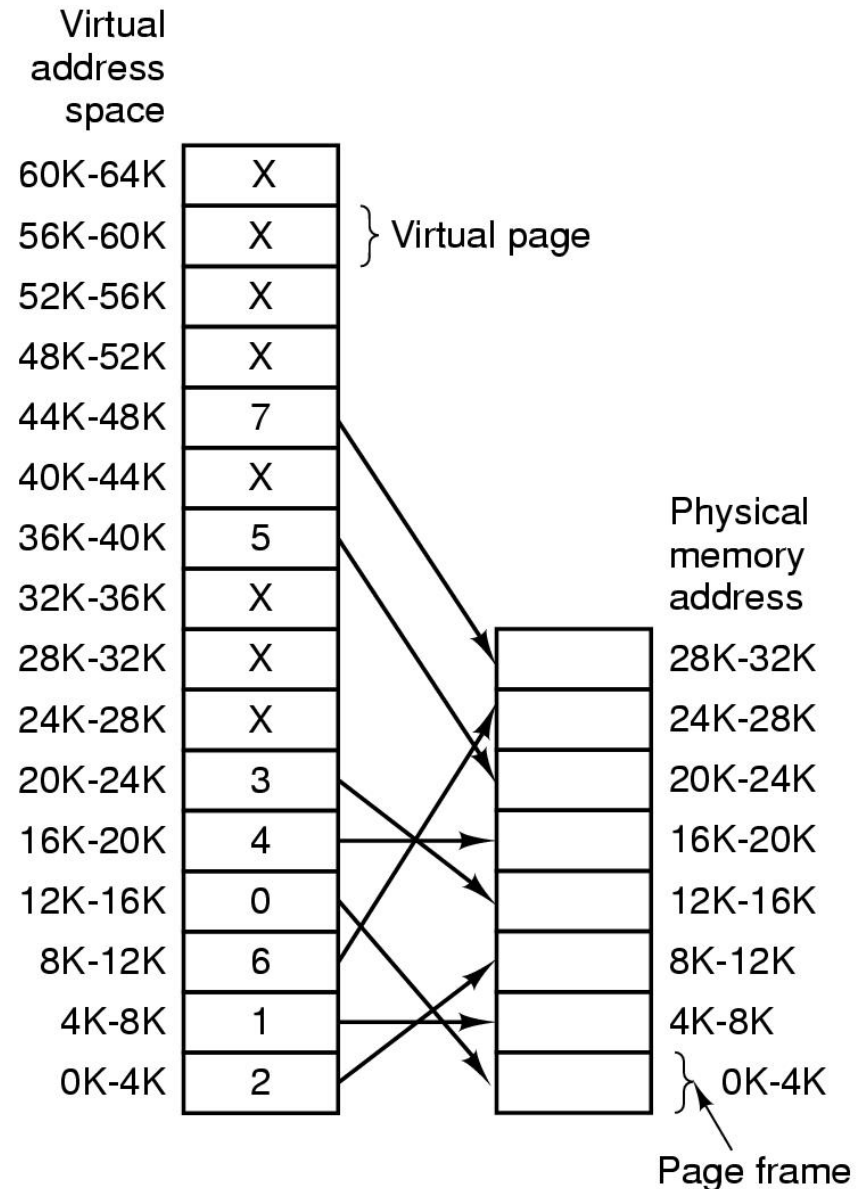
## Paginazione (1)



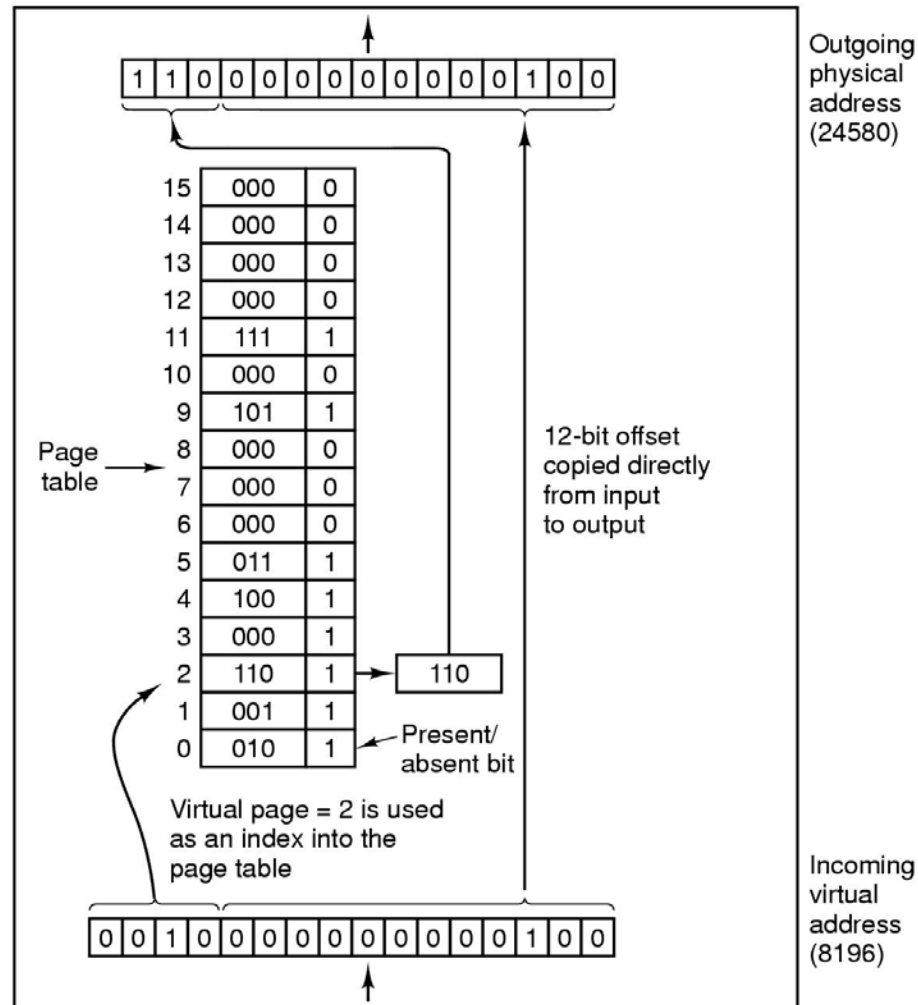
Posizione e funzione della MMU

# Paginazione (2)

La tabella delle  
pagine mostra la  
relazione tra  
indirizzi logici e  
indirizzi fisici

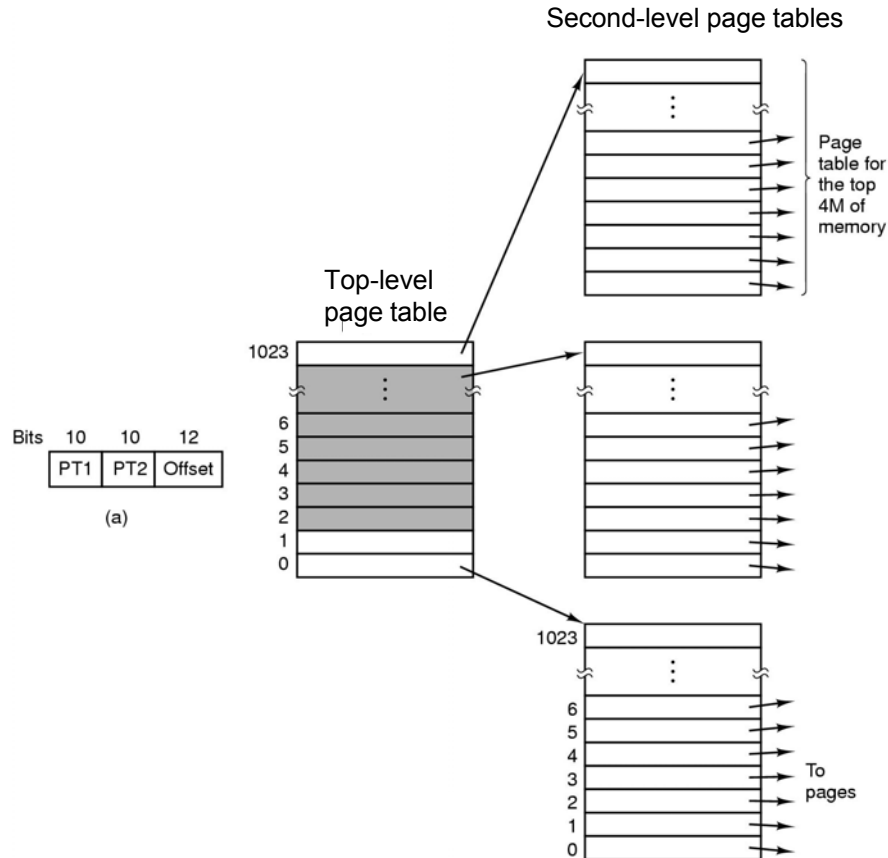


# Tabella delle Pagine (1)



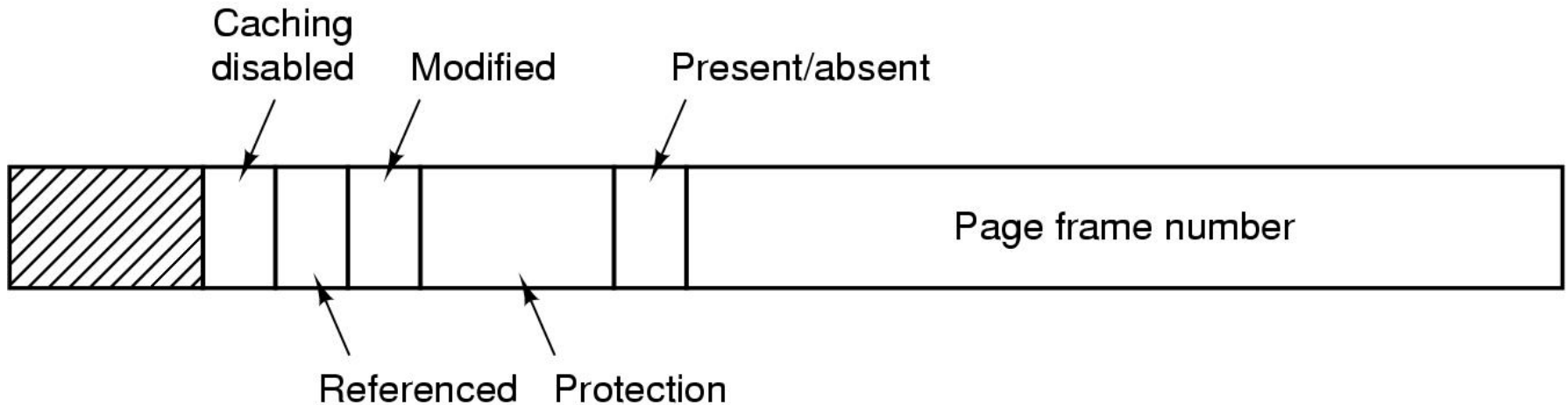
Operazioni di una MMU con 16 pagine di 4 KB

# Tabella delle Pagine (2)



- a) Indirizzi a 32 bit con tabella delle pagine a due campi
- b) Tabella delle pagine a due livelli

# Tabella delle Pagine (3)



Record in una tabella delle pagine

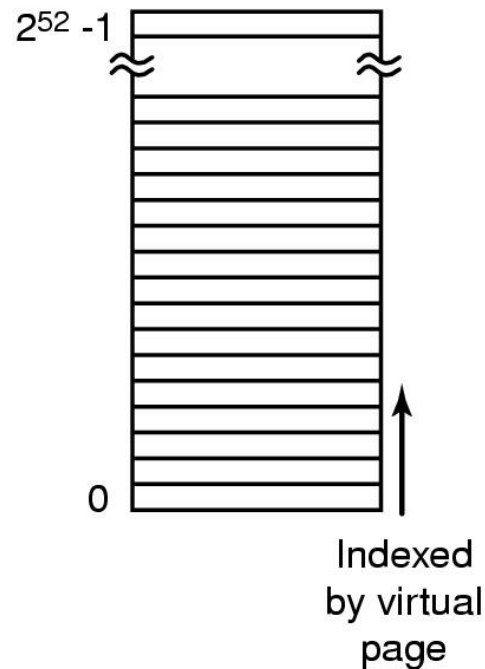
# Memoria Associativa

Valid	Virtual page	Modified	Protection	Page frame
1	140	1	RW	31
1	20	0	R X	38
1	130	1	RW	29
1	129	1	RW	62
1	19	0	R X	50
1	21	0	R X	45
1	860	1	RW	14
1	861	1	RW	75

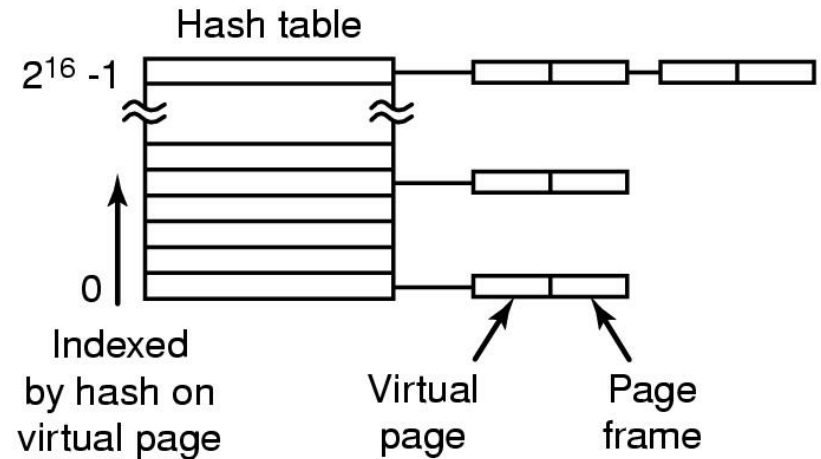
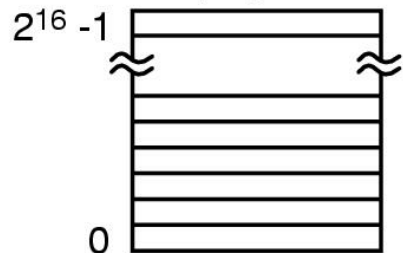
Esempio di memoria associativa

# Tabella delle Pagine Inversa

Traditional page  
table with an entry  
for each of the  $2^{52}$   
pages



256-MB physical  
memory has  $2^{16}$   
4-KB page frames



Confronto tra tabella delle pagine tradizionale e tabella delle  
pagine inversa



# Implementazione della Paginazione

Il Sistema operativo invoca I meccanismi di paginazione in quattro circostanze:

## 1. Creazione di un Processo

- Determina la dimensione del programma
- Crea la tabella delle pagine

## 2. Esecuzione di un Processo

- Reset della MMU per il nuovo processo
- Aggiornamento della cache

## 3. Page fault

- Determina l'indirizzo logico che ha causato il page fault
- Sposta una pagina su disco e carica la pagina richiesta

## 4. Terminazione di un Processo

- Dealloca la tabella delle pagine e le pagine del processo

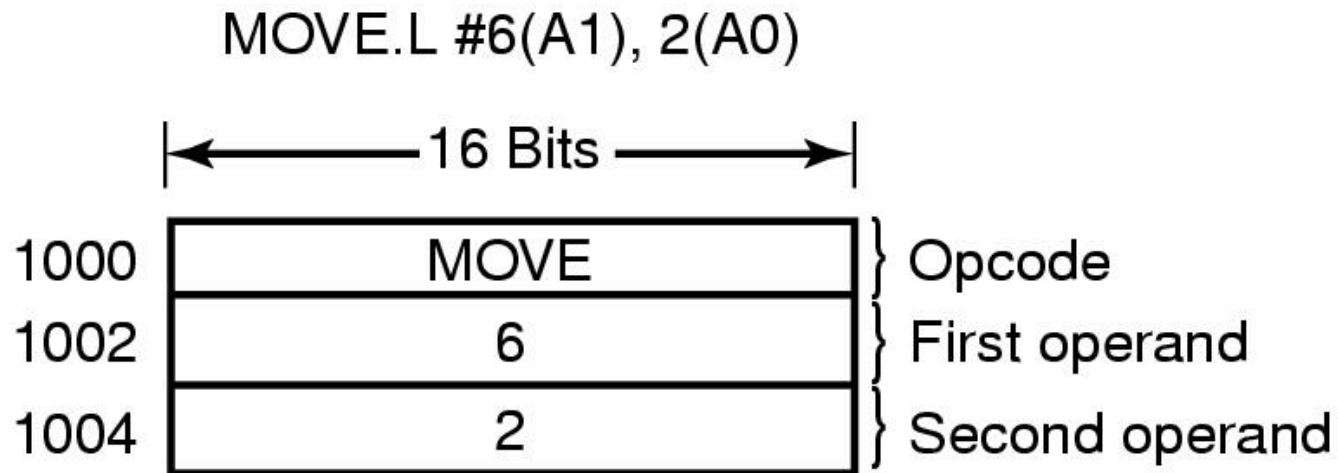
# Gestione del Page Fault (1)

1. Una eccezione provoca l'invocazione del nucleo
2. Salvataggio registri generali
3. Il sistema determina la pagina logica richiesta
4. Il sistema verifica la validità dell'indirizzo, e ricerca una pagina libera o, in alternativa, una pagina vittima
5. Se la pagina selezionata è stata modificata (dirty), la scrive su disco

# Gestione del Page Fault(2)

6. Il sistema schedula il caricamento della pagina dal disco
7. Aggiorna la tabella delle pagine
8. Viene ripristinata l'istruzione che ha causato il page fault
9. Il processo che ha causato il page fault viene schedulato
10. Ripristino dei registri
  - Il processo riprende l'elaborazione

# Ripristino dell'Istruzione

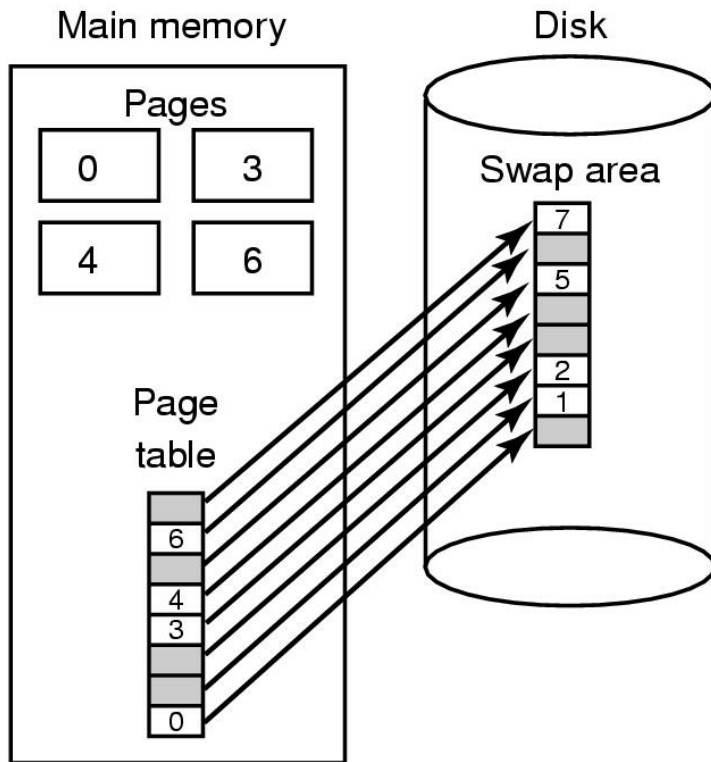


Un'istruzione che ha causato un page fault

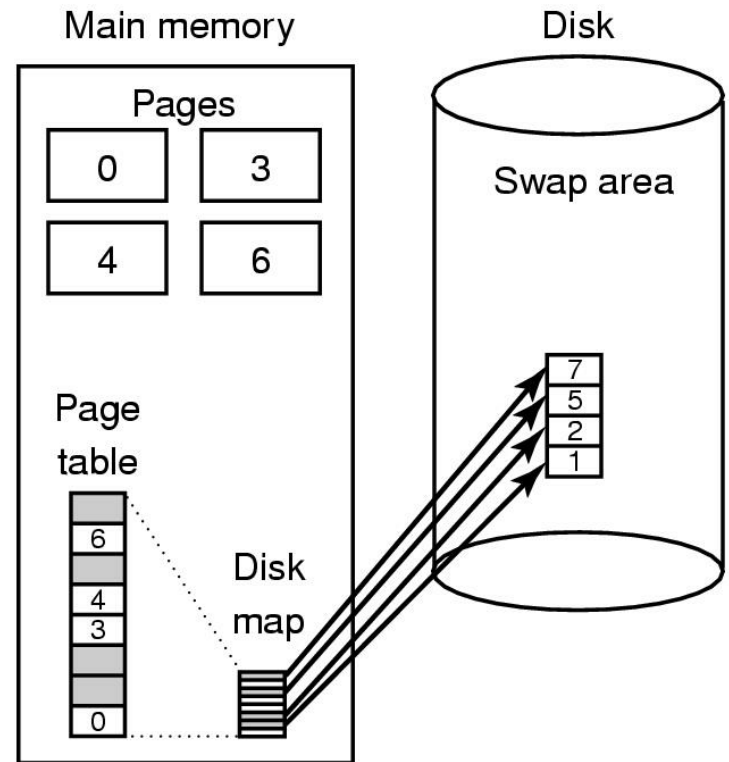
# Vincolare le pagine in Memoria

- Memoria virtuale e I/O interagiscono occasionalmente
- Un processo richiede una lettura da un dispositivo su un buffer
  - Mentre attende per l'I/O passa in esecuzione un altro processo
  - Il processo in esecuzione causa un page fault
  - La pagina contenente il buffer del primo processo può essere selezionata per essere scaricata dalla memoria
- Necessità di vincolare alcune pagine
  - Le pagine vincolate non possono essere scaricate dalla memoria

# Backing Store



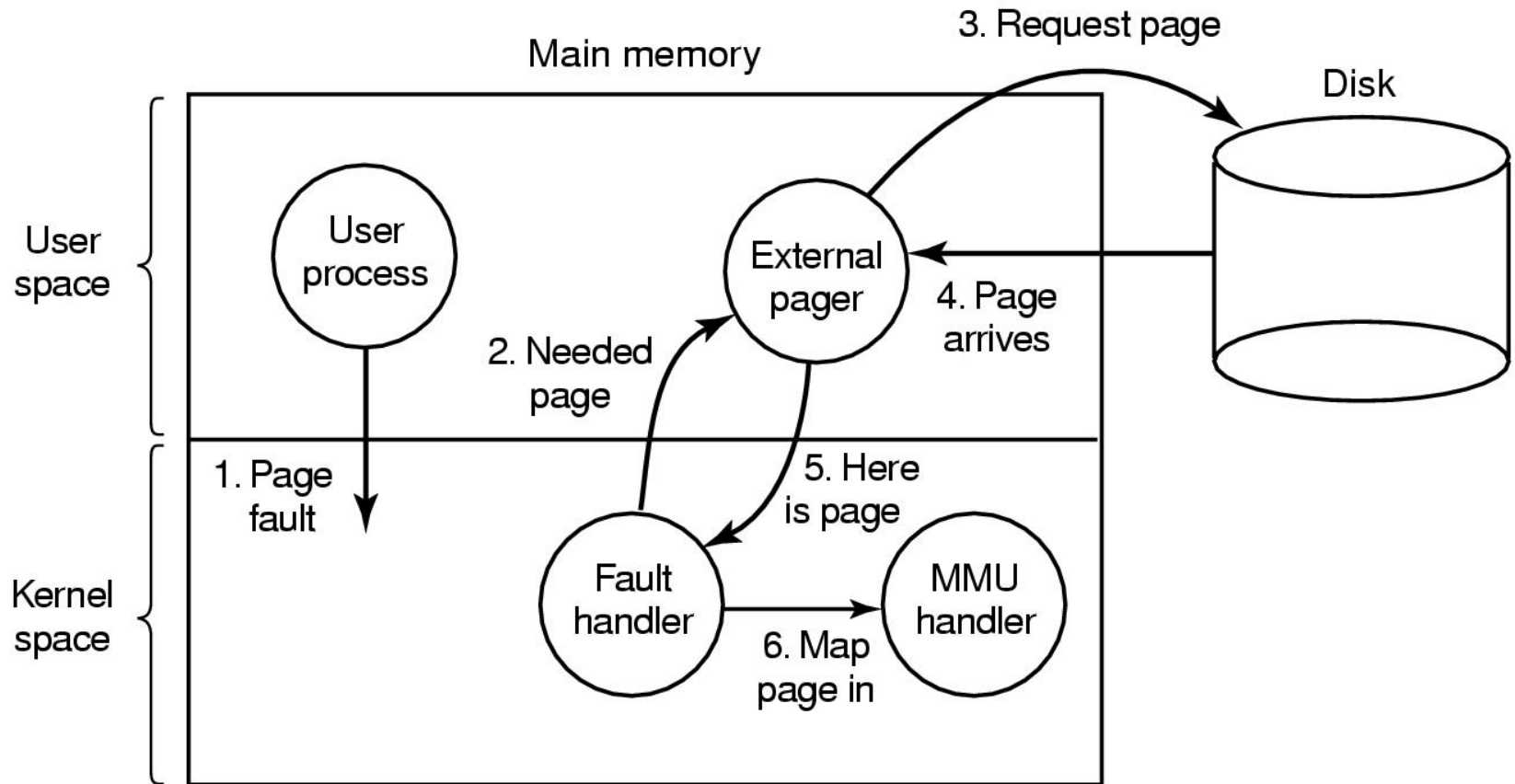
(a)



(b)

- (a) Paginazione con un'area di swap statica
- (b) Paginazione con area di swap dinamica

# Separazione tra Politiche e Meccanismi



Gestione di un Page fault con un pager esterno