

Inverted Page Table

La page table di ogni processo deve avere **una entry per ogni pagina logica**, a prescindere dal fatto che questa sia **realmente allocata in memoria o meno**



Il **numero totale di entry** può essere infinitamente maggiore del numero di frame disponibili

Inverted Page Table : tiene traccia **solo delle pagine fisiche realmente utilizzate**

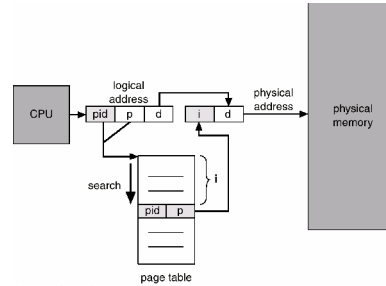
Operating System Concepts

8.48

Silberschatz and Galvin©1999

Perche' "inverted" ?

- Una entry per ogni **pagina fisica realmente utilizzata** in memoria
- Dal **numero di processo e di pagina logica** si può risalire alla pagina fisica



Meno spazio di memoria

Maggiore tempo necessario per ritrovare una pagina: ricerca sequenziale su tutti i frame della coppia (processo, pagina logica)

Si può usare una hash table per limitare la ricerca

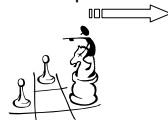
Operating System Concepts

8.49

Silberschatz and Galvin©1999

Condivisione di pagine

- **IDEA** : Mappare una stessa pagina fisica su più pagine logiche, in modo da condividere applicazioni e/o dati
- Non facilmente praticabile con una **inverted page table**
- Se si tratta di **codice** :
 - non deve essere in grado di modificare se' stesso
 - deve apparire nella stessa locazione dello spazio logico di tutti i processi
- **Casi di uso** : editor, compilatori, etc.



Operating System Concepts

8.50

Silberschatz and Galvin©1999

E perche' la pagina condivisa deve apparire nella stessa locazione dello spazio logico di tutti i processi ???

Gli **autoriferimenti alla pagina condivisa** implicano che essa abbia lo stesso numero logico nei diversi programmi perche' **gli indirizzi che essa contiene** sono sempre di tipo **<pagina logica, offset>**

Invece **indirizzi/riferimenti che sono espressi in termini di offset** dal program counter attuale o da un registro che contiene il corrente numero di pagina sono **indiretti** e quindi vanno bene

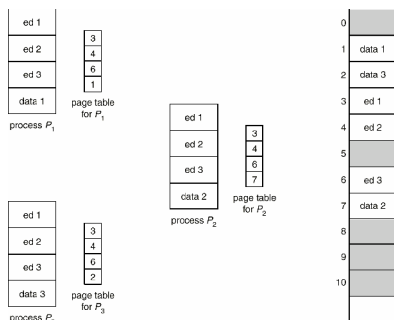


Operating System Concepts

8.51

Silberschatz and Galvin©1999

... ed un esempio di condivisione



Operating System Concepts

8.52

Silberschatz and Galvin©1999

2. Segmentazione : idea di base

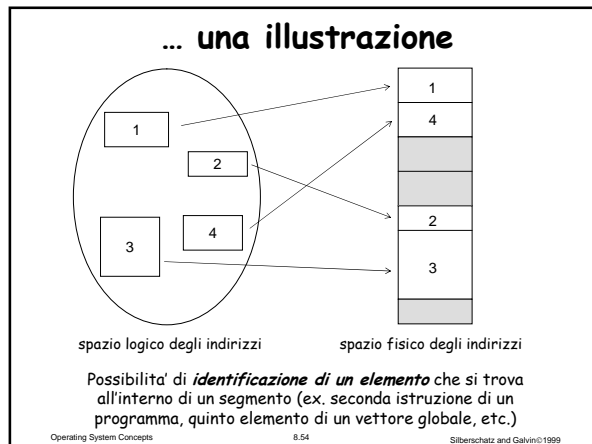
Suddivisione di un processo in **parti logicamente differenti** (segmenti), di **dimensioni diverse**, per esempio:

1. il programma principale
2. le procedure
3. le funzioni
4. le variabili globali

Operating System Concepts

8.53

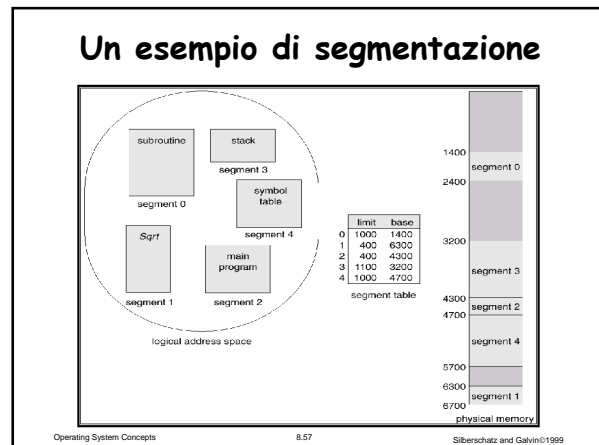
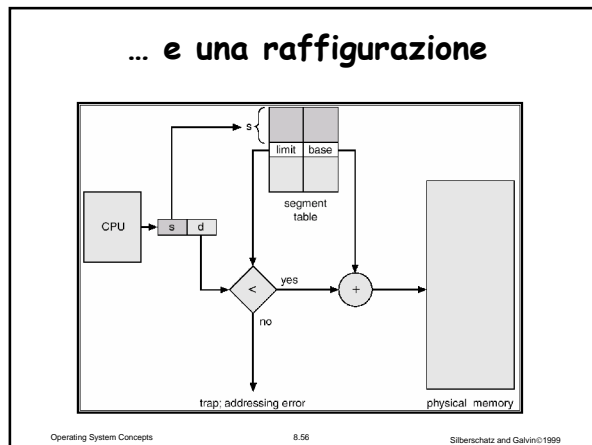
Silberschatz and Galvin©1999



... implementazione...

- L'indirizzo logico consiste nella coppia:
 $\langle \text{segment-number}, \text{offset} \rangle$
- **Segment table** - ogni entry corrispondente ad un segment number contiene:
 - **base**: l'indirizzo fisico di inizio del segmento
 - **limit**: la lunghezza del segmento
- Analogamente al caso della paginazione:
 - **Segment-table base register (STBR)**
 - **Segment-table length register (STLR)**
- ... e tutte le considerazioni a proposito del **doppio accesso** a memoria fatte per la paginazione possono essere riproposte

Operating System Concepts 8.55 Silberschatz and Galvin©1999



Fondamentale differenza tra paginazione e segmentazione

Poiche' i segmenti possono essere di **differente dimensione**, ritrovare spazio in memoria per essi e' un problema di **allocazione dinamica della memoria**

↓

Si ritorna alle **First-fit, Best-fit, Worst-fit**, e alla **compattazione**

Operating System Concepts 8.58 Silberschatz and Galvin©1999



Ed infine una soluzione ibrida: Segmentazione paginata...

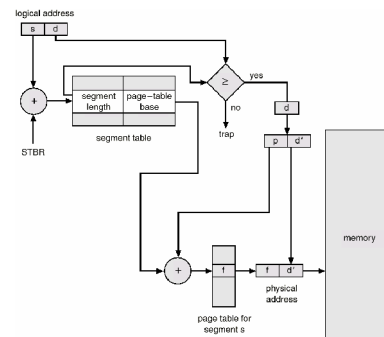
- Elimina il problema di frammentazione esterna per *segmenti troppo grandi*: essi vengono *suddivisi in pagine* e quindi allocate in frame equivalenti non necessariamente contigui
- OS MULTICS - *una page table distinta per ogni segmento*
- La soluzione differisce dalla segmentazione pura in quanto *una entry della segment table* non *contiene* l'indirizzo base di un segmento, ma *l'indirizzo base della page table di quel segmento*

Operating System Concepts

8.60

Silberschatz and Galvin©1999

... ed una sua illustrazione



Si potrebbe perfino pensare di *paginare a sua volta la segment table*, in quanto troppo grande!



Operating System Concepts

8.61

Silberschatz and Galvin©1999