

Paginazione multilivello

In un sistema a 64 bit, la memoria virtuale è molto più grande della memoria fisica. Ciò significa che rappresentare la tabella delle pagine come un array ha un overhead significativo, anche considerando che ogni processo deve avere la sua tabella.

Segmentazione con pagine

L'indirizzo logico è $\langle \#segmento + \langle \#pagina + offset \rangle \rangle$ (2 registri). La tabella dei segmenti contiene l'indirizzo della tabella delle pagine di ciascun segmento e la relativa lunghezza.

La tabella dei segmenti a volte è conservata in hardware dedicato, le tabelle delle pagine solitamente sono in memoria fisica. In genere si limita la lunghezza massima dei segmenti per facilitare l'allocazione delle tabelle. Così facendo si può usare un singolo registro per l'intero indirizzo logico, per esempio 10 bit per il segmento, 10 per la pagina, 12 per l'offset. Con questa configurazione la dimensione di una tabella delle pagine è esattamente una pagina da 4 KB (1K entry, una parola da 4 byte per entry).

I segmenti hanno sempre dimensione multipla della dimensione delle pagine, quindi c'è frammentazione interna, che è limitata se si usano pagine di dimensioni ridotte.

Oltre a diminuire i requisiti di spazio, risolve il problema di assegnare protezioni/condivisione a pagine che contengono indirizzi di tipo diverso (e.g. codice e dati) perché si trovano al confine tra due regioni – qui non succede, ogni segmento ha le sue pagine.

Più livelli di pagine

I primi $n - 1$ livelli contengono indirizzi delle tabelle del livello successivo, l'ultimo i numeri di pagina fisica. L'hardware conserva l'indirizzo della tabella di primo livello per il processo corrente. Indirizzi logici con 3 livelli: $\langle i_0, i_1, i_2, \text{offset} \rangle$.

Spesso le tabelle sono progettate per avere la stessa dimensione di una pagina fisica. Non è necessario allocare le tabelle (eccetto la prima) finché non si usano: su 32 bit, 2 livelli, pagine da 4 KB:

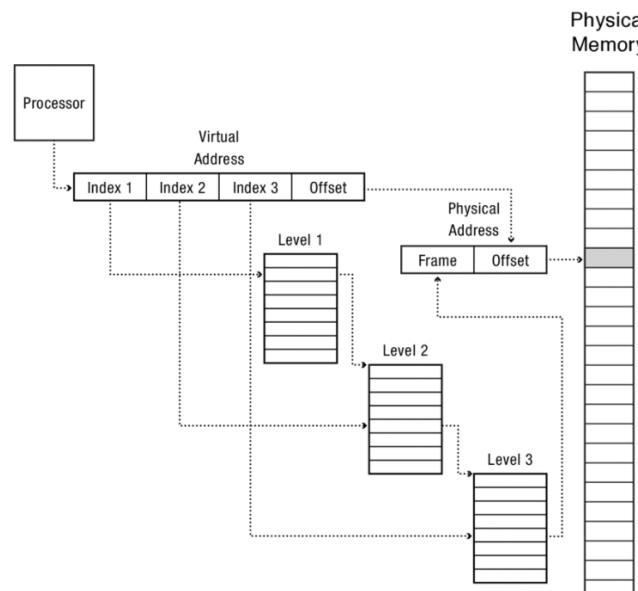
indirizzi logici 10 bit per il primo livello, 10 bit per il secondo, 12 di offset
 \Rightarrow tabelle da 4 KB;

caso ottimo tutto lo spazio di indirizzamento utilizzato è nella stessa tabella di secondo livello: una tabella per livello \Rightarrow 8 KB;

caso pessimo servono tutte le tabelle di secondo livello: $4 \text{ KB} + 2^{10} \cdot 4 \text{ KB} = 4 \text{ MB} + 4 \text{ KB}$.

Con un solo livello servono sempre 4 MB.

Le tabelle che sono state allocate ma non vengono usate da un po' possono essere spostate su disco per risparmiare memoria.



Segmentazione con paginazione multilivello

Usata per esempio da x86: la tabella dei segmenti (*global descriptor table* in x86, indirizzo della GDT corrente memorizzato nel GDTR) contiene l'indirizzo della tabella di primo livello per ciascun segmento.