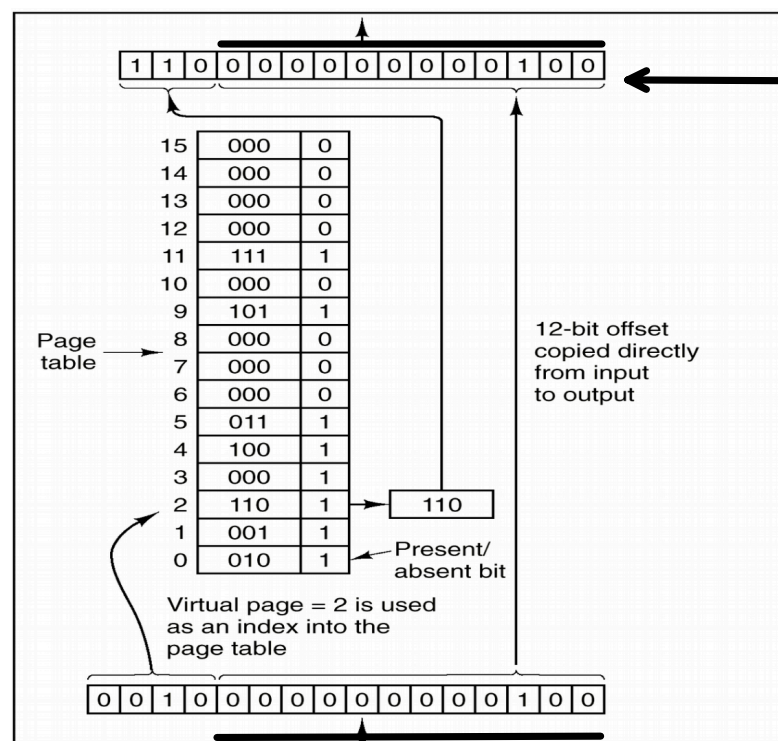


Gestione dei page-fault

Per conoscere quali sono le pagine materializzate dobbiamo sicuramente avere questa informazione all'interno della page table.

Quello che già sappiamo è che una Page Table ci dice che, ogni volta che noi vogliamo accedere ad una specifica pagina, l'idea è che nella tabella delle pagine c'è sicuramente una entry associata al numero della pagina e che ci dice che quella pagina è associata al frame X.

Per esempio, la pagina logica 0010, ossia la pagina numero 2, è associata al frame 110.



E quindi il frame in RAM è posizionato in quella posizione a cui poi applichiamo l'offset per ottenere l'indirizzo della RAM a cui dobbiamo accedere.

Però attenzione, nello schema di memoria virtuale non è sempre vero che l'informazione che troviamo nella page table sia valida: perché magari la pagina logica 0010 a cui stiamo cercando di accedere attualmente non è posizionata in RAM, e sappiamo che invece una tabella delle pagine sta lì per indicarci le posizioni delle pagine logiche all'interno della memoria fisica.

Per poter discriminare se, una entry di una tabella delle pagine è correntemente valida, dobbiamo avere un bit di informazione che ci dice se quella entry è valida.

Se il bit è pari a 1 possiamo realmente utilizzare quella informazione per andare a comporre l'indirizzo fisico sopra. Se il bit è pari a 0 e proviamo ad accedere alla pagina, il sottosistema di controllo del mio processore non riesce ad eseguire la traduzione da indirizzo logico ad indirizzo fisico, perché verificando questo bit va ad indicare che questa traduzione non è eseguibile: questo genera il page fault, ossia l'eccezione che cambia lo stato del processore e porta il controllo a un modulo del kernel del sistema operativo.

OFFSET SU QUELLA PAGINA LOGICA

Quindi la struttura di una page table è fondamentale per andare a poter generare i Page Fault che permettono al software del sistema operativo di prendere il controllo e di gestire il fatto che alcune pagine logiche possono non essere correntemente presenti all'interno della memoria, e ovviamente magari portarcele o cominciare a rappresentarcele, in particolare se le pagine logiche sono validamente accessibili, ossia sono mappate all'interno dell'address space. Il page fault è suddiviso in Major-Fault e Minor-Fault.

Un Major-Fault accade quando noi accediamo ad una pagina logica che era precedentemente ospitata in RAM ma che è stata poi portata fuori dalla RAM, quindi è stata swapped out: la gestione di questo fault prevede che noi dobbiamo interrogare un dispositivo, il dispositivo dove ospitiamo le informazioni di swap, per andare a ricaricare quella pagina all'interno della RAM. È un'operazione più complessa rispetto a scegliere un frame libero per cominciare a rappresentare quella pagina, e questo è il caso del Minor-Fault: quindi il fault che noi otteniamo nel momento in cui accediamo ad una pagina che non è mai stata materializzata, quindi non dobbiamo ricaricare ora le informazioni che riguardano il contenuto di quella pagina all'interno della RAM.

Avviene tramite il bit di presenza all'interno della tabella delle pagine



Se il bit di presenza vale 0 si ha un **page fault** e la pagina va caricata da hard-drive (swap-in) in memoria se mappata e materializzata in precedenza – altrimenti viene materializzata se mappata in precedenza

Major-fault

Minor-fault

