

Indirizzi logici e fisici

Questo ci porta la necessità di introdurre due nuovi concetti. Uno è il cosiddetto indirizzo logico e l'altro è il cosiddetto indirizzo fisico.



Se noi rianalizziamo i tipi di binding che abbiamo scritto in precedenza, quindi sia a tempo di compilazione che a tempo di caricamento, possiamo dire che quando l'applicazione realmente esegue, gli indirizzi logici e gli indirizzi fisici coincidono.

Quando compiliamo un'applicazione con "binding a tempo di compilazione", noi andiamo già ad esprimere riferimento a qualsiasi cosa all'interno dell'address space come un riferimento esattamente alla posizione di memoria dove l'oggetto viene ad essere ospitato.

Quando abbiamo il binding a tempo di compilazione, il riferimento al salto per l'istruzione B, viene fatto esattamente con la posizione di memoria all'interno della RAM di questa istruzione B.

Questa stessa cosa avviene quando abbiamo il "binding a tempo di caricamento" e carichiamo un programma rilocabile utilizzando la rilocazione di questi riferimenti.

Per il binding a tempo d'esecuzione possono NON coincidere. Quindi dato un address space, quando il mio processore sta cercando di lavorare all'interno di questo address space e vuole accedere ad un oggetto B (Istruzione o Dato), per identificare questo oggetto B come posizione all'interno dell'address space si esprime un offset INTERNO all'address space, questo è l'indirizzo LOGICO. Questo poi verrà a sua volta trasformato nello specifico indirizzo RAM dove l'oggetto B va realmente a risiedere, e quest'ultimo è l'indirizzo fisico, e per capire dove questo indirizzo logico è mappato in termini fisici, questa operazione viene eseguita da tramite un apposito dispositivo hardware (presente all'interno dei processori) che si chiama **MMU (Memory Management Unit)**.

