



SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Architettura degli Elaboratori

19 – Memoria Virtuale

Prof. Andrea Sterbini – sterbini@di.uniroma1.it



Argomenti

▶ **Argomenti della lezione**

- Memoria Virtuale
- Esercizi

▶ **Problema:**

- **In un sistema multiprocesso la gestione della memoria era complicata**
- **Inoltre la memoria fisica può non essere sufficiente**

Argomenti

▶ **Argomenti della lezione**

- Memoria Virtuale
- Esercizi

▶ **Problema:**

- **In un sistema multiprocesso la gestione della memoria era complicata**
- **Inoltre la memoria fisica può non essere sufficiente**

▶ **Requisiti:**

Argomenti

▶ **Argomenti della lezione**

- Memoria Virtuale
- Esercizi

▶ **Problema:**

- **In un sistema multiprocesso la gestione della memoria era complicata**
- **Inoltre la memoria fisica può non essere sufficiente**

▶ **Requisiti:**

- **Rendere lo spazio di memoria disponibile ai processi indipendente dallo spazio effettivamente disponibile (memoria fisica)**

Argomenti

▶ **Argomenti della lezione**

- Memoria Virtuale
- Esercizi

▶ **Problema:**

- **In un sistema multiprocesso la gestione della memoria era complicata**
- **Inoltre la memoria fisica può non essere sufficiente**

▶ **Requisiti:**

- **Rendere lo spazio di memoria disponibile ai processi indipendente dallo spazio effettivamente disponibile (memoria fisica)**

▶ **Vantaggi:**

Argomenti

▶ **Argomenti della lezione**

- Memoria Virtuale
- Esercizi

▶ **Problema:**

- **In un sistema multiprocesso la gestione della memoria era complicata**
- **Inoltre la memoria fisica può non essere sufficiente**

▶ **Requisiti:**

- **Rendere lo spazio di memoria disponibile ai processi indipendente dallo spazio effettivamente disponibile (memoria fisica)**

▶ **Vantaggi:**

- **lo spazio di indirizzamento di un processo si semplifica**

Argomenti

▶ Argomenti della lezione

- Memoria Virtuale
- Esercizi

▶ Problema:

- In un sistema multiprocesso la gestione della memoria era complicata
- Inoltre la memoria fisica può non essere sufficiente

▶ Requisiti:

- Rendere lo spazio di memoria disponibile ai processi indipendente dallo spazio effettivamente disponibile (memoria fisica)

▶ Vantaggi:

- lo spazio di indirizzamento di un processo si semplifica
- gli spazi di memoria dei processi vengono separati e protetti da accessi esterni

Argomenti

▶ Argomenti della lezione

- Memoria Virtuale
- Esercizi

▶ Problema:

- In un sistema multiprocesso la gestione della memoria era complicata
- Inoltre la memoria fisica può non essere sufficiente

▶ Requisiti:

- Rendere lo spazio di memoria disponibile ai processi indipendente dallo spazio effettivamente disponibile (memoria fisica)

▶ Vantaggi:

- lo spazio di indirizzamento di un processo si semplifica
- gli spazi di memoria dei processi vengono separati e protetti da accessi esterni

▶ IDEA:

Argomenti

▶ **Argomenti della lezione**

- Memoria Virtuale
- Esercizi

▶ **Problema:**

- **In un sistema multiprocesso la gestione della memoria era complicata**
- **Inoltre la memoria fisica può non essere sufficiente**

▶ **Requisiti:**

- **Rendere lo spazio di memoria disponibile ai processi indipendente dallo spazio effettivamente disponibile (memoria fisica)**

▶ **Vantaggi:**

- **lo spazio di indirizzamento di un processo si semplifica**
- **gli spazi di memoria dei processi vengono separati e protetti da accessi esterni**

▶ **IDEA:**

- l gli indirizzi (istruzioni e dati) prodotti dalla CPU sono «virtuali» e vengono trasformati in indirizzi «fisici» per accedere alla vera posizione dei dati

Argomenti

▶ **Argomenti della lezione**

- Memoria Virtuale
- Esercizi

▶ **Problema:**

- **In un sistema multiprocesso la gestione della memoria era complicata**
- **Inoltre la memoria fisica può non essere sufficiente**

▶ **Requisiti:**

- **Rendere lo spazio di memoria disponibile ai processi indipendente dallo spazio effettivamente disponibile (memoria fisica)**

▶ **Vantaggi:**

- **lo spazio di indirizzamento di un processo si semplifica**
- **gli spazi di memoria dei processi vengono separati e protetti da accessi esterni**

▶ **IDEA:**

- 1 gli indirizzi (istruzioni e dati) prodotti dalla CPU sono «virtuali» e vengono trasformati in indirizzi «fisici» per accedere alla vera posizione dei dati
- 2 La memoria è divisa in «pagine» che sono presenti in memoria fisica solo se necessario (altrimenti possono essere memorizzate in una memoria più lenta)

Mapping degli indirizzi virtuali

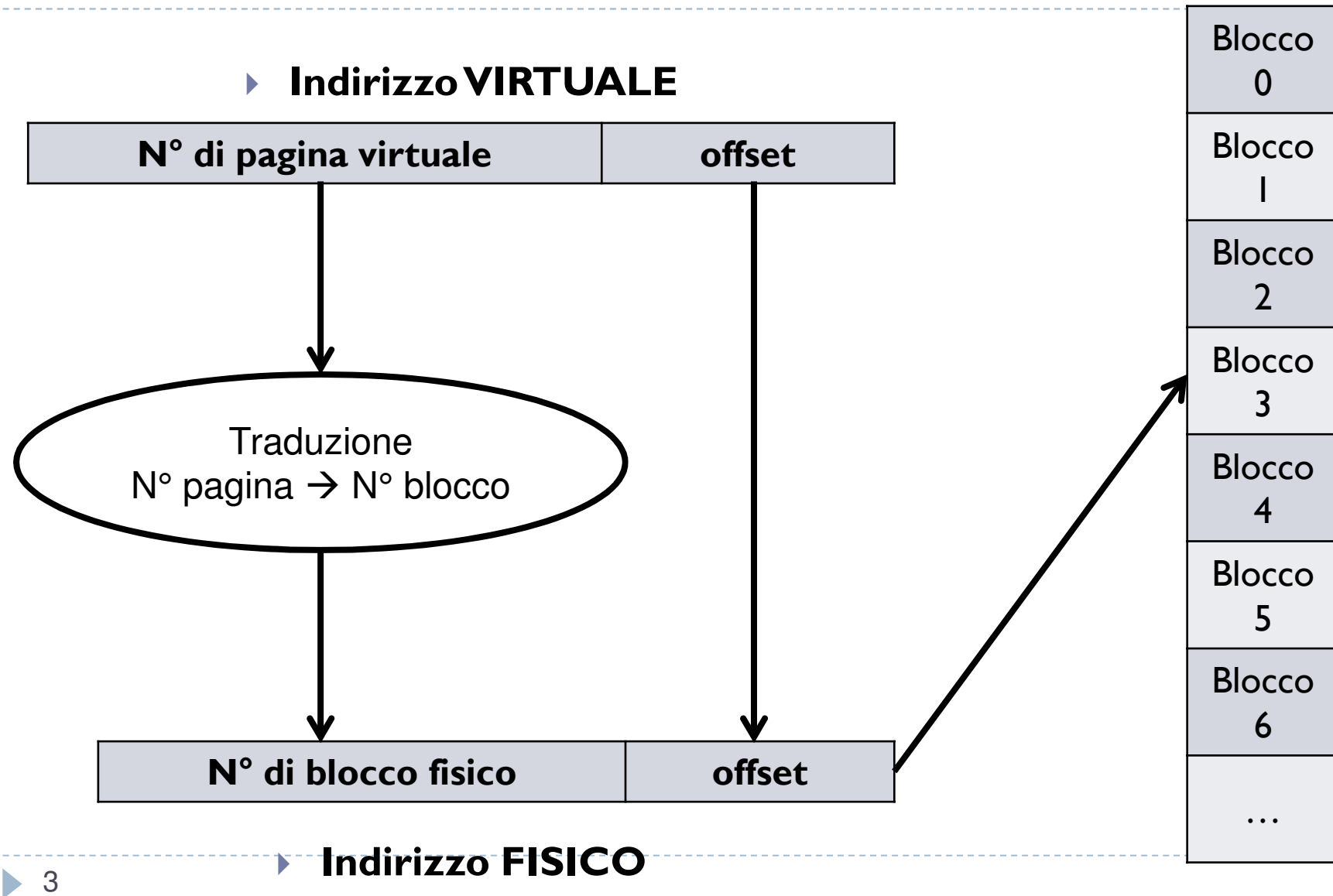
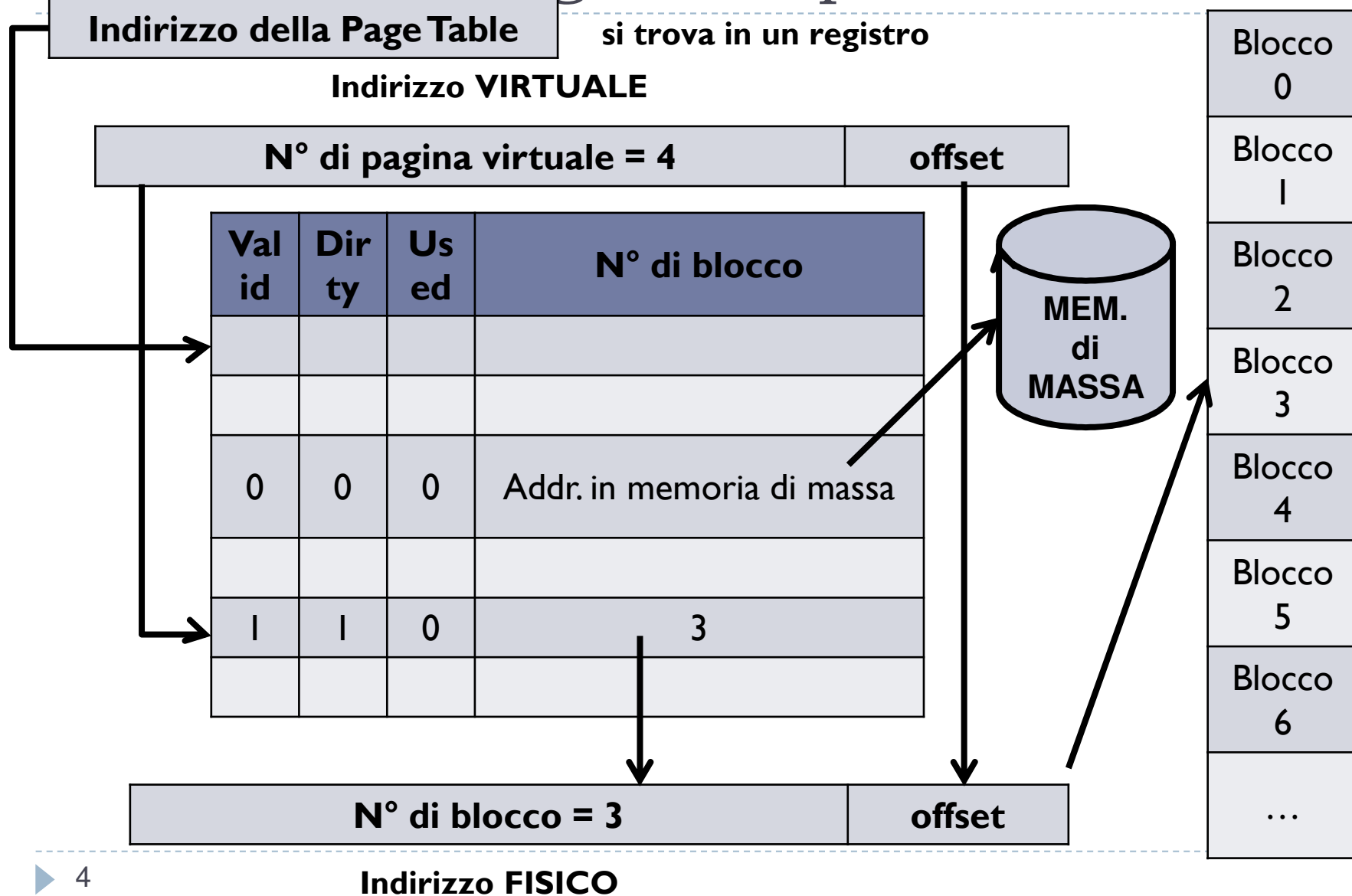


Tabella delle Pagine del processo



Accesso ad un dato

- ▶ Quando la pagina è già nella memoria fisica:
 - **Valid = 1**, la tabella indica dove si trova la pagina nella memoria fisica
 - Sono necessari 2 accessi per ottenere il dato (1 alla Page Table e 1 alla memoria fisica)
 - **NOTA:** i 2 accessi sono il caso migliore! Vanno resi più veloci!

Accesso ad un dato

- ▶ Quando la pagina è già nella memoria fisica:
 - **Valid = 1**, la tabella indica dove si trova la pagina nella memoria fisica
 - Sono necessari 2 accessi per ottenere il dato (1 alla Page Table e 1 alla memoria fisica)
 - **NOTA:** i 2 accessi sono il caso migliore! Vanno resi più veloci!
- ▶ Quando la pagina NON è in memoria fisica:
 - **Valid = 0**, la tabella indica dove si trova il blocco nella memoria di massa
 - Viene lanciato una eccezione di «Page Fault» per chiedere al sistema operativo che la pagina sia recuperata e inserita in memoria fisica
 - Sono necessari milioni di cicli di clock per trovare e copiare il blocco dal disco (lento)

Accesso ad un dato

- ▶ Quando la pagina è già nella memoria fisica:
 - **Valid = 1**, la tabella indica dove si trova la pagina nella memoria fisica
 - Sono necessari 2 accessi per ottenere il dato (1 alla Page Table e 1 alla memoria fisica)
 - **NOTA:** i 2 accessi sono il caso migliore! Vanno resi più veloci!
- ▶ Quando la pagina NON è in memoria fisica:
 - **Valid = 0**, la tabella indica dove si trova il blocco nella memoria di massa
 - Viene lanciato una eccezione di «Page Fault» per chiedere al sistema operativo che la pagina sia recuperata e inserita in memoria fisica
 - Sono necessari milioni di cicli di clock per trovare e copiare il blocco dal disco (lento)
- ▶ Gli altri bit della Page Table:
 - **Dirty:** indica che la pagina è stata modificata in memoria, per cui sarà necessario salvarla in memoria di massa prima di eliminarla dalla memoria fisica
 - **Used:** indica che la pagina è stata utilizzata di recente, serve ad approssimare la politica di rimpiazzo LRU

Politiche di rimpiazzo e scrittura

- ▶ Come per la cache dobbiamo sostituire pagine già usate e mantenere la coerenza dei dati

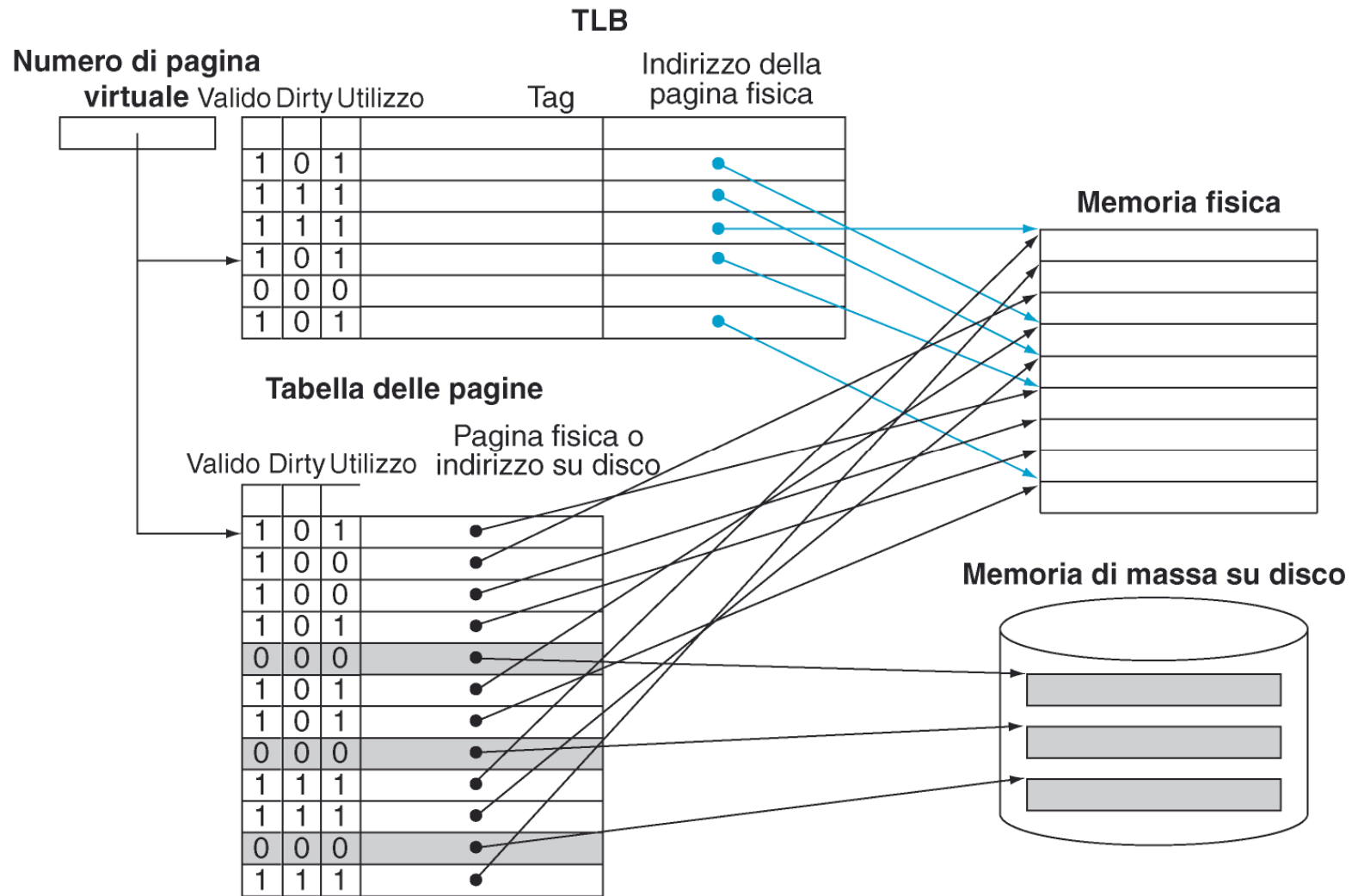
Politiche di rimpiazzo e scrittura

- ▶ Come per la cache dobbiamo sostituire pagine già usate e mantenere la coerenza dei dati
- ▶ **Politiche di rimpiazzo:**
 - **LRU** (Least Recently Used): sostituire la pagina «**utilizzata meno di recente**»
 - **LFU** (Least Frequently Used): sostituire la pagina «**utilizzata meno spesso**»
 - **Random**: sostituire una pagina a caso
- ▶ Nel caso della VM si ha molto più tempo a disposizione e conviene usare politiche molto più sofisticate (anche piccoli miglioramenti danno guadagni di tempo molto grandi)

Politiche di rimpiazzo e scrittura

- ▶ Come per la cache dobbiamo sostituire pagine già usate e mantenere la coerenza dei dati
- ▶ **Politiche di rimpiazzo:**
 - **LRU** (Least Recently Used): sostituire la pagina «**utilizzata meno di recente**»
 - **LFU** (Least Frequently Used): sostituire la pagina «**utilizzata meno spesso**»
 - **Random**: sostituire una pagina a caso
- ▶ Nel caso della VM si ha molto più tempo a disposizione e conviene usare politiche molto più sofisticate (anche piccoli miglioramenti danno guadagni di tempo molto grandi)
- ▶ **Politiche di scrittura:**
 - Ha senso solo la **Write Back** (i tempi per scrivere un blocco sono troppo lunghi)
 - **NOTA:** un blocco viene scritto su disco SOLO se Dirty = 1

Accesso più veloce col TLB (Table Lookaside Buffer)

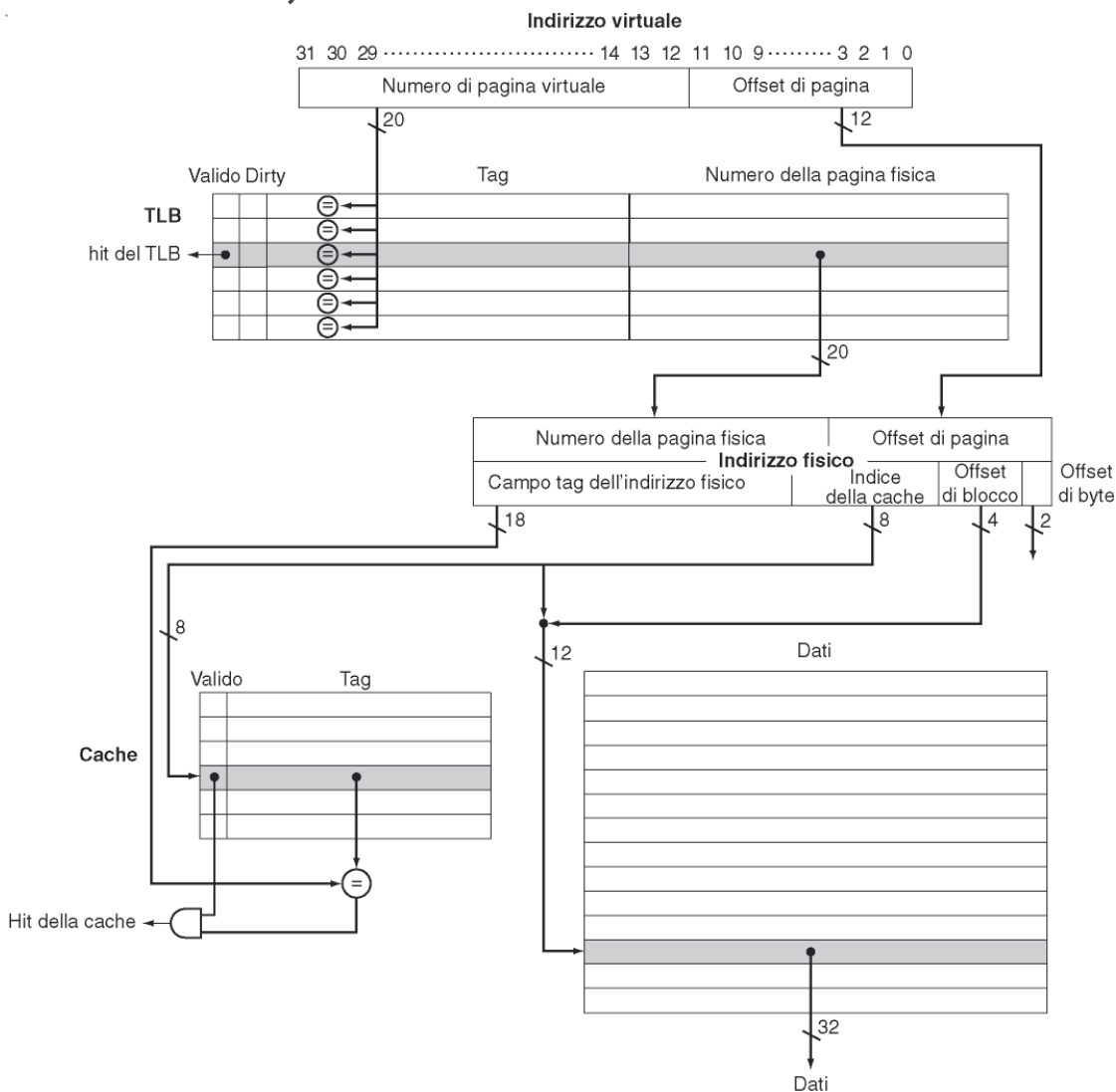


TLB + Cache del dato (processore FastMATH)

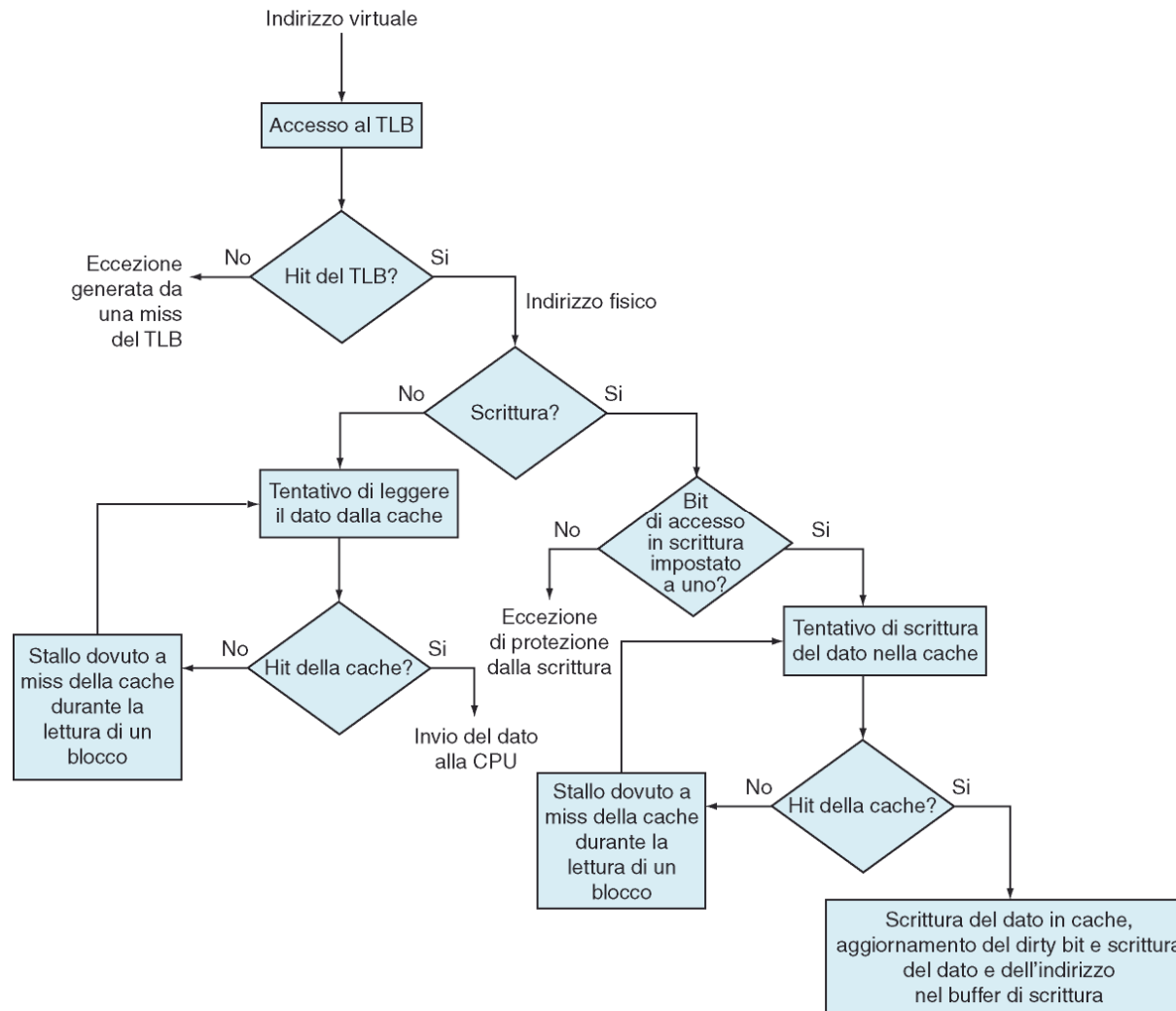
- ▶ Per rendere ancora più veloce l'accesso nel caso migliore si opera inserendo una cache sui due accessi
- **TLB:** cache dell'accesso alla Page Table
- **Cache:** dell'accesso al dato (in memoria fisica)

▶ In genere la cache sul dato può essere indicizzata:

- dall'indirizzo fisico
 - Va svuotata al caricamento di una nuova pagina
- oppure dall'indirizzo virtuale
 - Va svuotata al context switch



Fasi dell'accesso



Casistica

TLB	Tabella delle pagine	Cache	Possibile? Se sì, in quale situazione?
Hit	Hit	Miss	Possibile, sebbene la tabella delle pagine non venga mai realmente controllata se si verifica una hit del TLB.
Miss	Hit	Hit	Miss del TLB, ma l'elemento si trova nella tabella delle pagine; al secondo tentativo, il dato viene trovato nella cache.
Miss	Hit	Miss	Miss del TLB, ma l'elemento si trova nella tabella delle pagine; al secondo tentativo, si verifica però una miss della cache.
Miss	Miss	Miss	Miss del TLB, seguita da un page fault; al secondo tentativo, l'accesso al dato deve provocare una miss della cache.
Hit	Miss	Miss	Impossibile: il TLB non può fornire la traduzione di una pagina che non è presente in memoria.
Hit	Miss	Hit	Impossibile: il TLB non può fornire la traduzione di una pagina che non è presente in memoria.
Miss	Miss	Hit	Impossibile: i dati non possono trovarsi nella cache se la pagina non è presente in memoria.

Casistica

TLB	Tabella delle pagine	Cache	Possibile? Se sì, in quale situazione?
Hit	Hit	Miss	Possibile, sebbene la tabella delle pagine non venga mai realmente controllata se si verifica una hit del TLB.
Miss	Hit	Hit	Miss del TLB, ma l'elemento si trova nella tabella delle pagine; al secondo tentativo, il dato viene trovato nella cache.
Miss	Hit	Miss	Miss del TLB, ma l'elemento si trova nella tabella delle pagine; al secondo tentativo, si verifica però una miss della cache.
Miss	Miss	Miss	Miss del TLB, seguita da un page fault; al secondo tentativo, l'accesso al dato deve provocare una miss della cache.
Hit	Miss	Miss	Impossibile: il TLB non può fornire la traduzione di una pagina che non è presente in memoria.
Hit	Miss	Hit	Impossibile: il TLB non può fornire la traduzione di una pagina che non è presente in memoria.
Miss	Miss	Hit	Impossibile: i dati non possono trovarsi nella cache se la pagina non è presente in memoria.

Supporto del Sistema Operativo

- ▶ Dev'esserci un **unico gestore delle pagine** fisiche: il **Sistema Operativo**
- Risponde alle eccezioni di Page Fault caricando in memoria fisica le pagine virtuali
- Assegna le nuove pagine fisiche ai processi che le richiedono
- Recupera le pagine dai processi che hanno terminato l'esecuzione (o che le rilasciano)

Supporto del Sistema Operativo

- ▶ Dev'esserci un **unico gestore delle pagine** fisiche: il **Sistema Operativo**
 - Risponde alle eccezioni di Page Fault caricando in memoria fisica le pagine virtuali
 - Assegna le nuove pagine fisiche ai processi che le richiedono
 - Recupera le pagine dai processi che hanno terminato l'esecuzione (o che le rilasciano)

- ▶ **E' l'unico che può modificare le Page Table dei processi**
 - Impedisce che il processo possa accedere alle pagine di altri processi
 - Svuota il TLB quando fa un process switch (oppure il TLB contiene l'id del processo)

Supporto del Sistema Operativo

- ▶ Dev'esserci un **unico gestore delle pagine** fisiche: il **Sistema Operativo**
 - Risponde alle eccezioni di Page Fault caricando in memoria fisica le pagine virtuali
 - Assegna le nuove pagine fisiche ai processi che le richiedono
 - Recupera le pagine dai processi che hanno terminato l'esecuzione (o che le rilasciano)

- ▶ **E' l'unico che può modificare le Page Table dei processi**
 - Impedisce che il processo possa accedere alle pagine di altri processi
 - Svuota il TLB quando fa un process switch (oppure il TLB contiene l'id del processo)

- ▶ La CPU deve impedire che il programma possa fare operazioni «pericolose» e gira in 2 modi:
 - **«user mode»** per i programmi utente
 - Non sono disponibili le istruzioni per modificare Page Table e Process Table, Interruzioni ...
 - **«kernel mode»** per il solo Sistema Operativo (tutte le istruzioni sono disponibili)

Quanto è grande la Page Table?

▶ **Esempio:** Pagine di 4 Kbyte, indirizzi di 32 bit

- 12 bit di offset
- 20 bit per il N° di pagina virtuale → **1 Mega pagine!** (cioè circa 4Mbytes x processo!)

▶ **Come ridurre la Page Table a seconda del comportamento dello spazio di memoria? Ci sono diverse possibilità:**

Quanto è grande la Page Table?

▶ **Esempio:** Pagine di 4 Kbyte, indirizzi di 32 bit

- 12 bit di offset
- 20 bit per il N° di pagina virtuale → **1 Mega pagine!** (cioè circa 4Mbytes x processo!)

▶ **Come ridurre la Page Table a seconda del comportamento dello spazio di memoria? Ci sono diverse possibilità:**

- **Se cresce linearmente**

- Puntatore all'inizio + lunghezza

Quanto è grande la Page Table?

▶ **Esempio:** Pagine di 4 Kbyte, indirizzi di 32 bit

- 12 bit di offset
- 20 bit per il N° di pagina virtuale → **1 Mega pagine!** (cioè circa 4Mbytes x processo!)

▶ **Come ridurre la Page Table a seconda del comportamento dello spazio di memoria? Ci sono diverse possibilità:**

- **Se cresce linearmente**

- Puntatore all'inizio + lunghezza

- **Se è divisibile in due parti che crescono linearmente** (p. es. stack + heap)

- 2 puntatori ai due inizi e 2 lunghezze

Quanto è grande la Page Table?

▶ **Esempio:** Pagine di 4 Kbyte, indirizzi di 32 bit

- 12 bit di offset
- 20 bit per il N° di pagina virtuale → **1 Mega pagine!** (cioè circa 4Mbytes x processo!)

▶ **Come ridurre la Page Table a seconda del comportamento dello spazio di memoria? Ci sono diverse possibilità:**

- **Se cresce linearmente**

- Puntatore all'inizio + lunghezza

- **Se è divisibile in due parti che crescono linearmente** (p. es. stack + heap)

- 2 puntatori ai due inizi e 2 lunghezze

- Usare una **funzione di hashing** che conduce ad uno spazio pari al num. di blocchi fisici

Quanto è grande la Page Table?

▶ **Esempio:** Pagine di 4 Kbyte, indirizzi di 32 bit

- 12 bit di offset
- 20 bit per il N° di pagina virtuale → **1 Mega pagine!** (cioè circa 4Mbytes x processo!)

▶ **Come ridurre la Page Table a seconda del comportamento dello spazio di memoria? Ci sono diverse possibilità:**

- **Se cresce linearmente**

- Puntatore all'inizio + lunghezza

- **Se è divisibile in due parti che crescono linearmente** (p. es. stack + heap)

- 2 puntatori ai due inizi e 2 lunghezze

- Usare una **funzione di hashing** che conduce ad uno spazio pari al num. di blocchi fisici

- Strutturare la **tabella gerarchicamente** (albero di sotto-tabelle)

- **Pro:** OK per spazi di memoria sparsi
- **Contro:** il numero di accessi alla memoria aumenta per ogni livello dell'albero

Quanto è grande la Page Table?

► **Esempio:** Pagine di 4 Kbyte, indirizzi di 32 bit

- 12 bit di offset
- 20 bit per il N° di pagina virtuale → **1 Mega pagine!** (cioè circa 4Mbytes x processo!)

► **Come ridurre la Page Table a seconda del comportamento dello spazio di memoria? Ci sono diverse possibilità:**

- **Se cresce linearmente**

- Puntatore all'inizio + lunghezza

- **Se è divisibile in due parti che crescono linearmente** (p. es. stack + heap)

- 2 puntatori ai due inizi e 2 lunghezze

- Usare una **funzione di hashing** che conduce ad uno spazio pari al num. di blocchi fisici

- Strutturare la **tabella gerarchicamente** (albero di sotto-tabelle)

- **Pro:** OK per spazi di memoria sparsi
- **Contro:** il numero di accessi alla memoria aumenta per ogni livello dell'albero

- Mettere la **Page Table nella memoria virtuale** (tabella paginata)

- **Pro:** Solo le parti necessarie saranno in memoria
- **Contro:** 3 accessi in memoria (Page Table, n° di blocco, dato)
- **NOTA:** alcune pagine devono restare sempre in RAM

Confronto tra livelli diversi

Caratteristica	Valori tipici per una cache L1	Valori tipici per una cache L2	Valori tipici per una memoria a pagine	Valori tipici per un TLB
Dimensione totale in blocchi	250-2000	15 000-50 000	16 000-250 000	40-1024
Dimensione totale in kilobyte	16-64	500-4000	1 000 000-1 000 000 000	0,25-16
Dimensione del blocco in byte	16-64	64-128	4000-64 000	4-32
Penalità di miss in cicli di clock	10-25	100-1000	10 000 000-100 000 000	10-1000
Frequenza di miss (globale per L2)	2%-5%	0,1%-2%	0,00001-0,0001%	0,01%-2%

► **Esercizio per casa:**

- Supponete di avere un sistema di memoria a più livelli formato da

- **CPU** clock a 2 GHz con 3 CPI
- **Cache L1** Tempo di HIT = 5 ns, percentuale di MISS = 3%
- **Cache L2** Tempo di HIT = 30ns, percentuale di MISS = 1%
- **RAM** Tempo di lettura blocco (MISS su L2) = 200 ns
- **TLB** Tempo di HIT = 1 ns, percentuale di MISS = 0.5%
- **Memoria Virtuale** Tempo di Page Fault = 3 secondi, Percentuale di MISS = 0,0001%

► Qual è il tempo totale di accesso per 1000000 di accessi?

► Quante istruzioni vengono svolte in media per ogni accesso?

► **NOTA:** ignorate i tempi di scrittura delle pagine su disco necessari per il Write Back