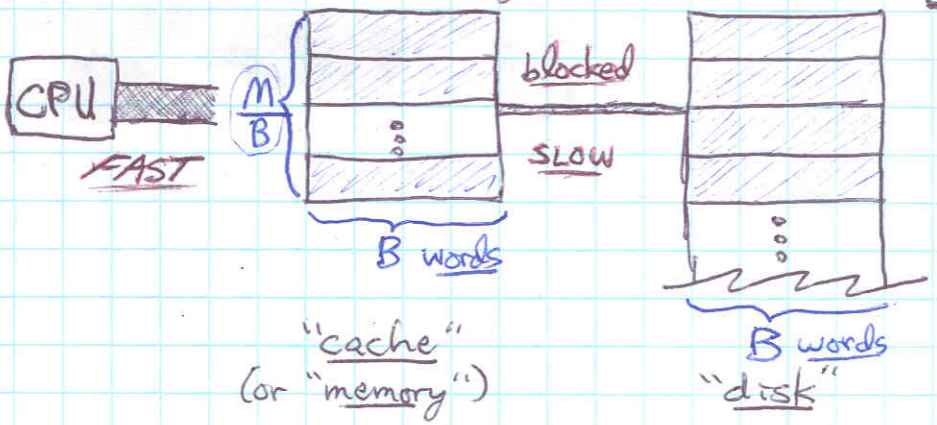
External-memory / I/O / Disk Access Model

[Aggarwal & Vitter – CACM 1988]

6.897
L 23.1
May 3, 2005

- two-level memory hierarchy



- charge mainly for memory transfers between cache & disk
- any algorithm with running time $T(N)$ uses $\leq T(N)$ memory transfers
- want to do better (without increasing running time)

Search trees:

- B-trees with branching factor $\Theta(B)$
  support Insert, Delete, Search in $O(\lg_{B+1} N)$ memory transfers
  ($\&$ $O(\lg N)$ time in comparison model)
- optimal Search bound in comparison model:
  - need to determine where query fits among N items
    $\Rightarrow$ need $\geq \lg(N+1)$ bits
  - each block read determines where query fits among B items
    $\Rightarrow$ learn $\leq \lg(B+1)$ bits per memory transfer
    $\Rightarrow \geq \lg_{B+1}(N+1)$ memory transfers

Sorting: $O\left(\frac{N}{B} \lg_{M/B} \frac{N}{B}\right)$ memory transfers ← much better than B-trees!
$\Omega(\ )$ in comparison model

Permutation: $\Theta\left(\min\left\{N, \frac{N}{B} \lg_{M/B} \frac{N}{B}\right\}\right)$
Ω in "indivisible" model – can't split word among different blocks

↓ execute given permutation

Buffer tree: $O\left(\frac{1}{B} \lg_{M/B} \frac{N}{B}\right)$ am. for delayed queries / batched updates (except Delete-Min)