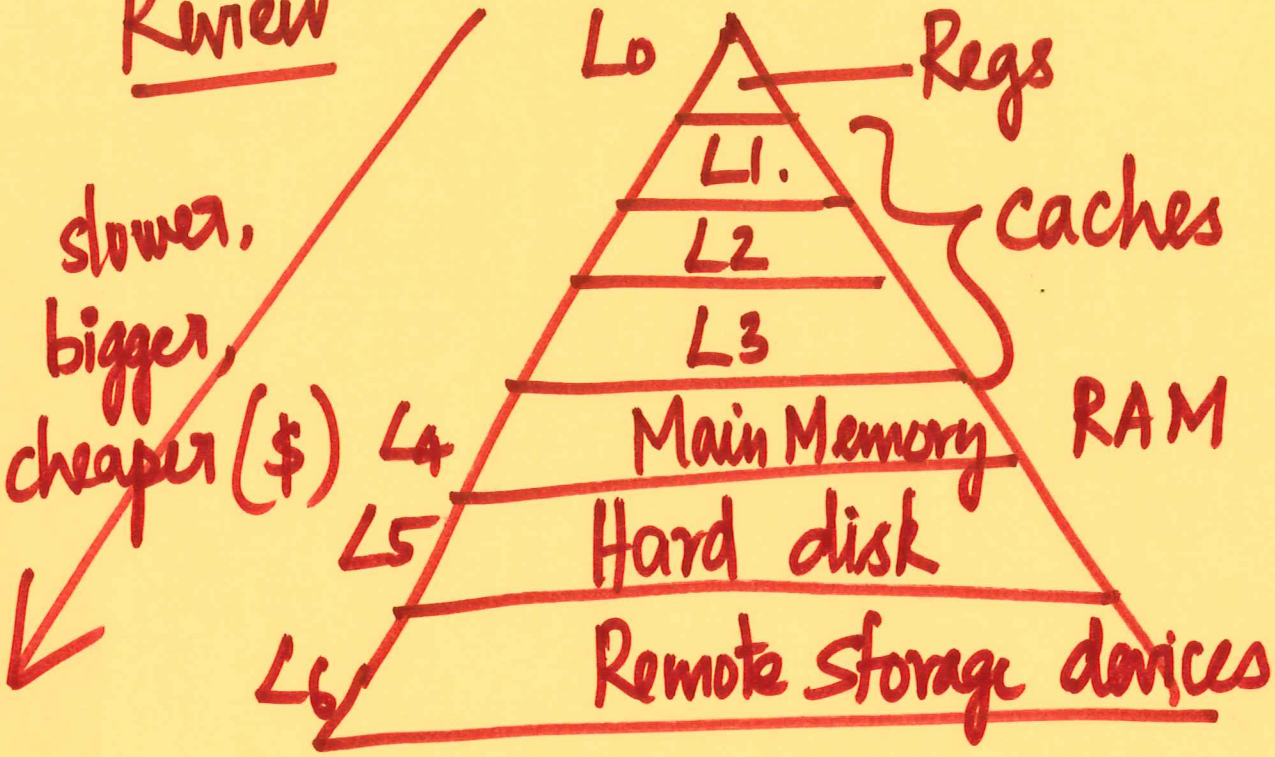
Lecture - 24

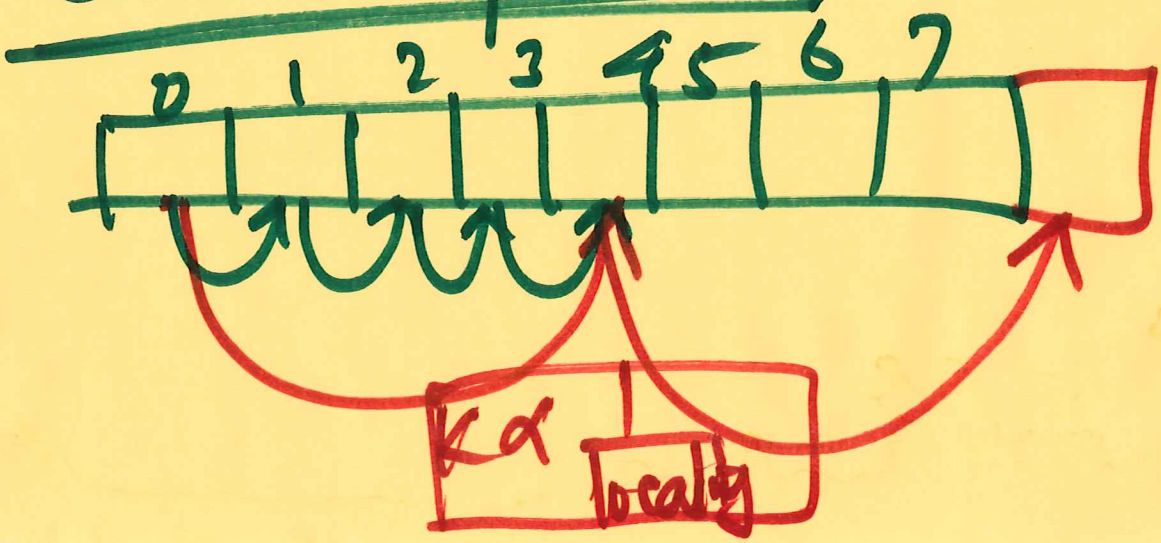
Review



Locality

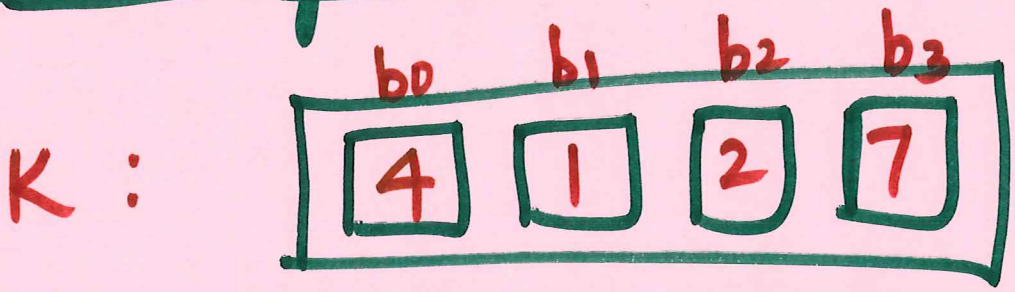
1. Temporal
2. Spatial

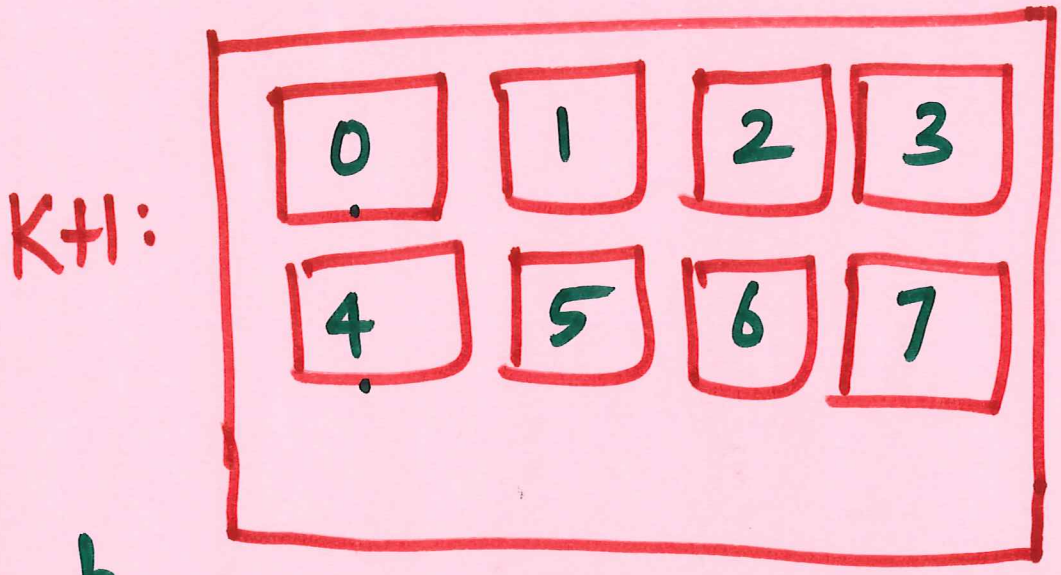
Stride - k patterns



Today

Caching and Caches





Block

32-bit computer.

each addr — 32 bits (4 bytes)

0x FA BC D1 C2

word size = 32 bits (4 bytes)

block = group of words

eg. 1 block = 8 words

= 8 × 4 = 32 bytes

$L_0 \leftrightarrow L_1$: 1 block = 1 word

$L_1 \leftrightarrow L_2$:

$L_2 \leftrightarrow L_3$

$L_3 \leftrightarrow L_4$

Main Memory

1 block = 8 or 16 words

at this level = 8 × 4 = 32 bytes

〃

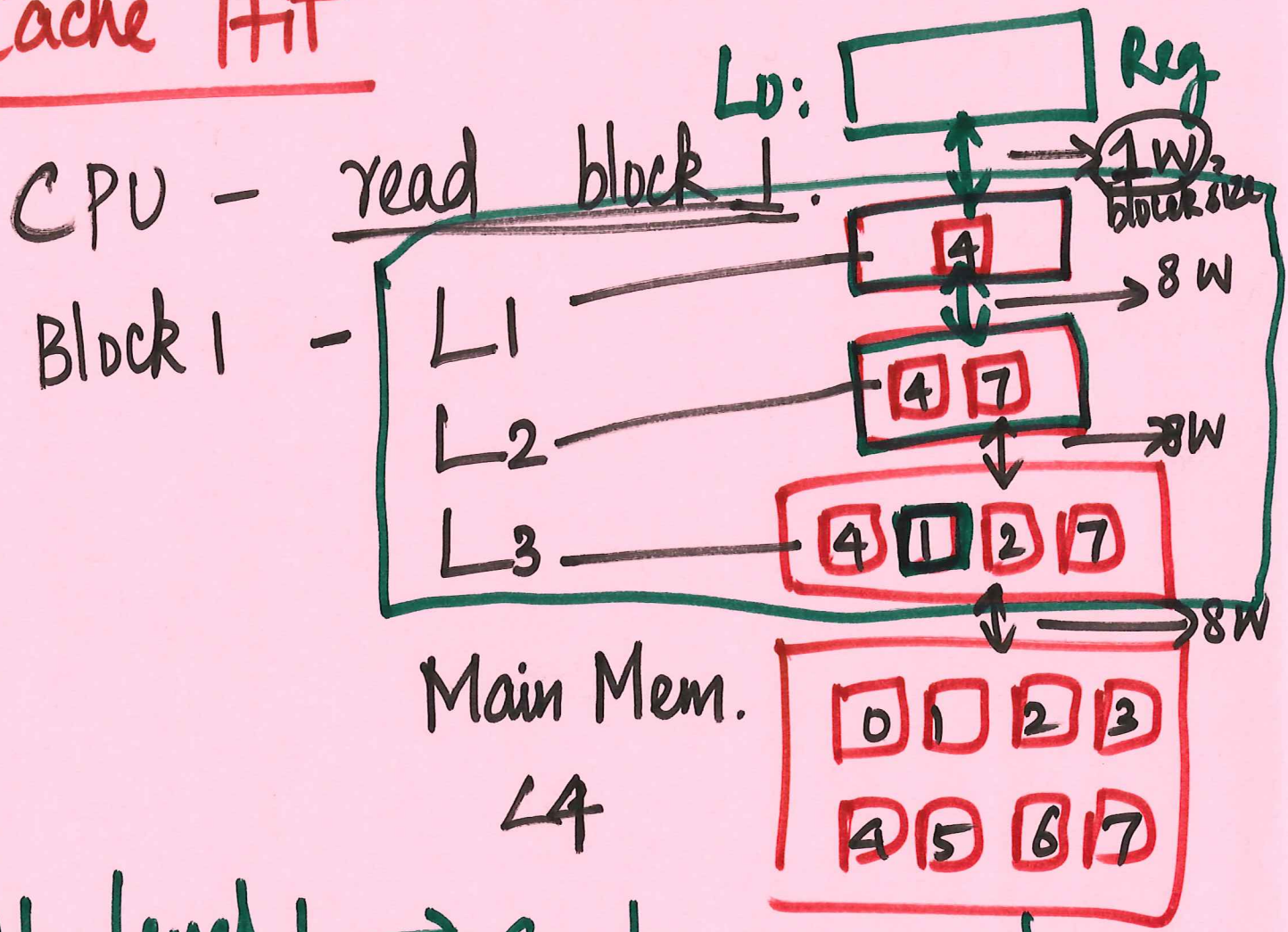
〃

$L_4 \leftrightarrow L_5$: HDD

1 block = 1000's of bytes

eg. 1KB = 1024 bytes.

Cache Hit



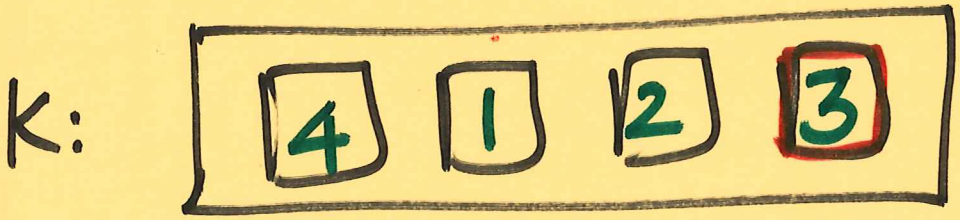
At level 1 → Cache miss at L1.

At L2 → " " " L2.

L3 → Yes! Cache hit @ L3.

Read [3]

1. Cache miss at L3.
2. Read from MM (main mem) ⟹ Cache hit @ L4.
3. Copy the block to L3. (prev level cache).



[7] → victim block

How did we choose the victim?

Cache Replacement Policy!

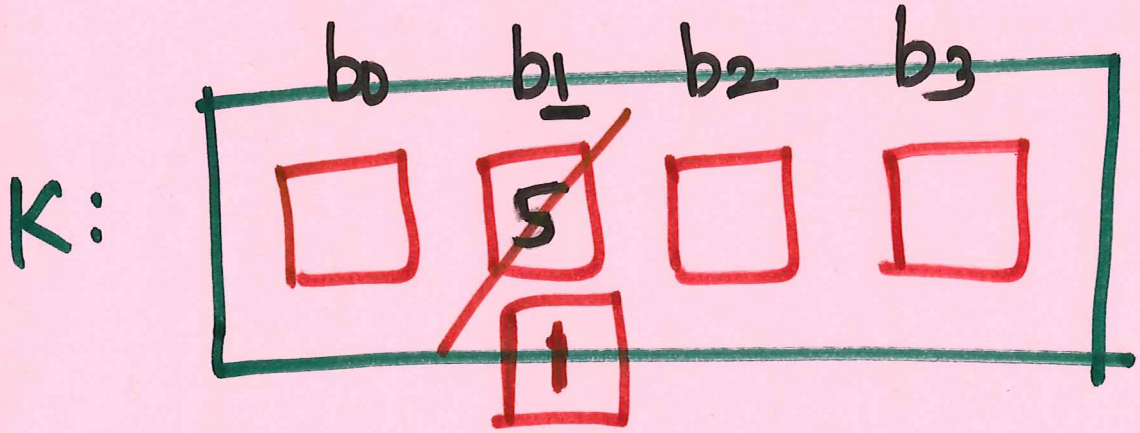
eg. LRU

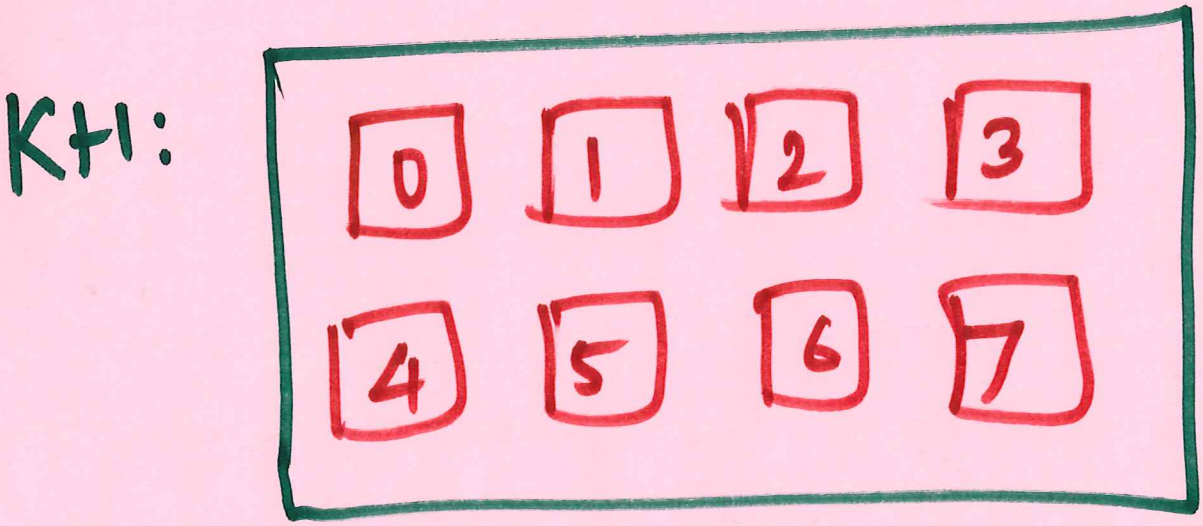
eg. Random.

Placement Policy

1. Flexible - any block for K+1 can be placed anywhere in K.
2. Restrictive.

eg. blocknum % 4





5 % 4 = 1 → b1

1. cold misses
2. conflict misses.
3. capacity miss. if working set > cache size.

Read blocks 0, 1, 2, 3 & 4 in the same order multiple times