Computadors

El Computador i els Seus Elements (2/2)

Grau en Ciència i Enginyeria de Dades

Facultat d'Informàtica de Barcelona (FIB)

Universitat Politècnica de Catalunya (UPC)

Creative Commons License

Aquest document utilitza Creative Commons Attribution 3.0 Unported License

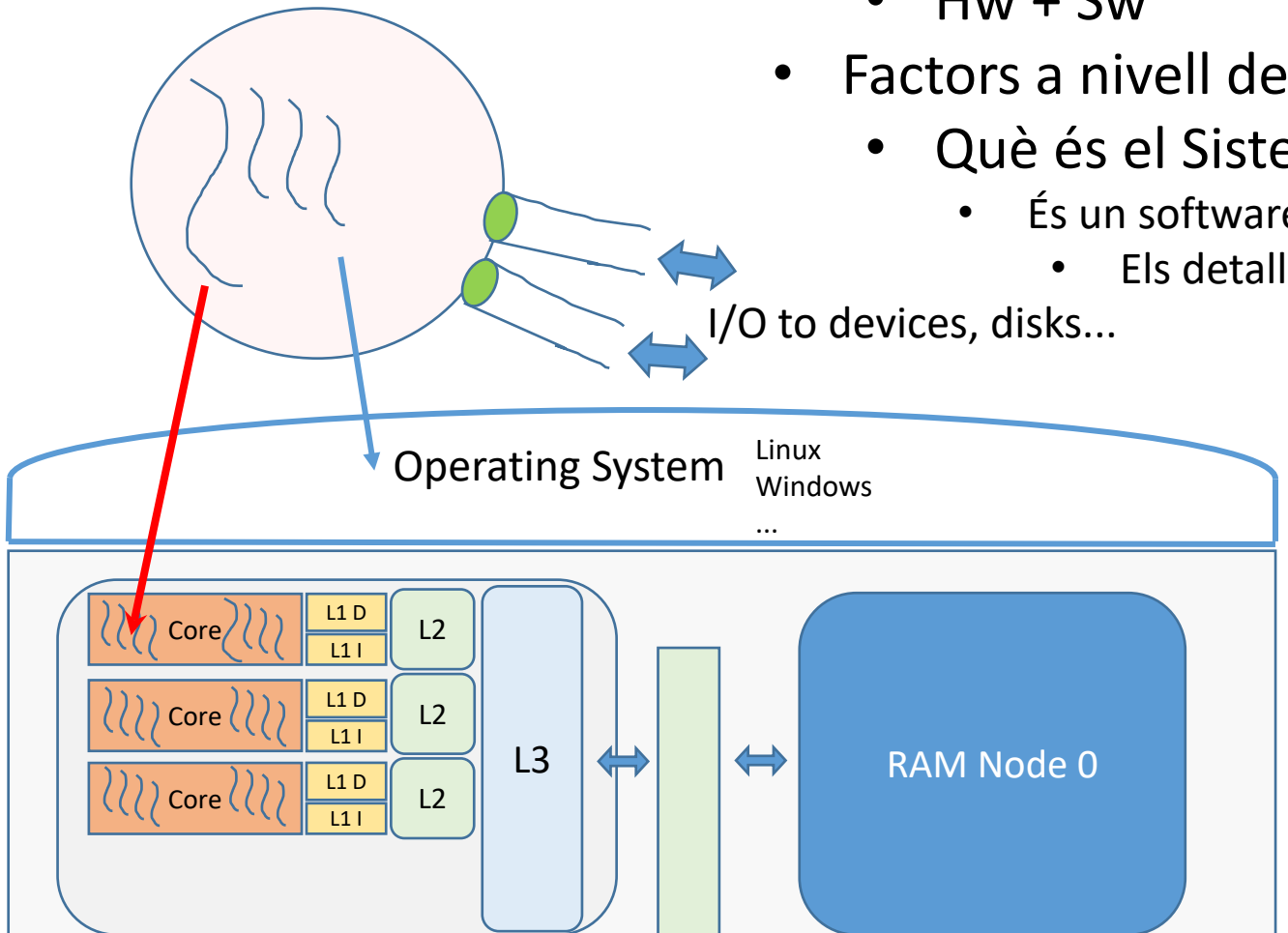
Els detalls d'aquesta licencia es poden trovar a
https://creativecommons.org/licenses/by-nc-nd/4.0

Index

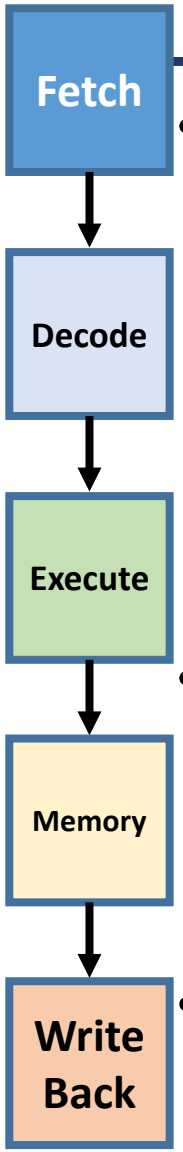
- Aquest tema té tres parts
  1) Components Bàsics i Jerarquia de Memòria
  2) **Fluxe d'execució i colls d'ampolla**

Factors que afecten el rendiment

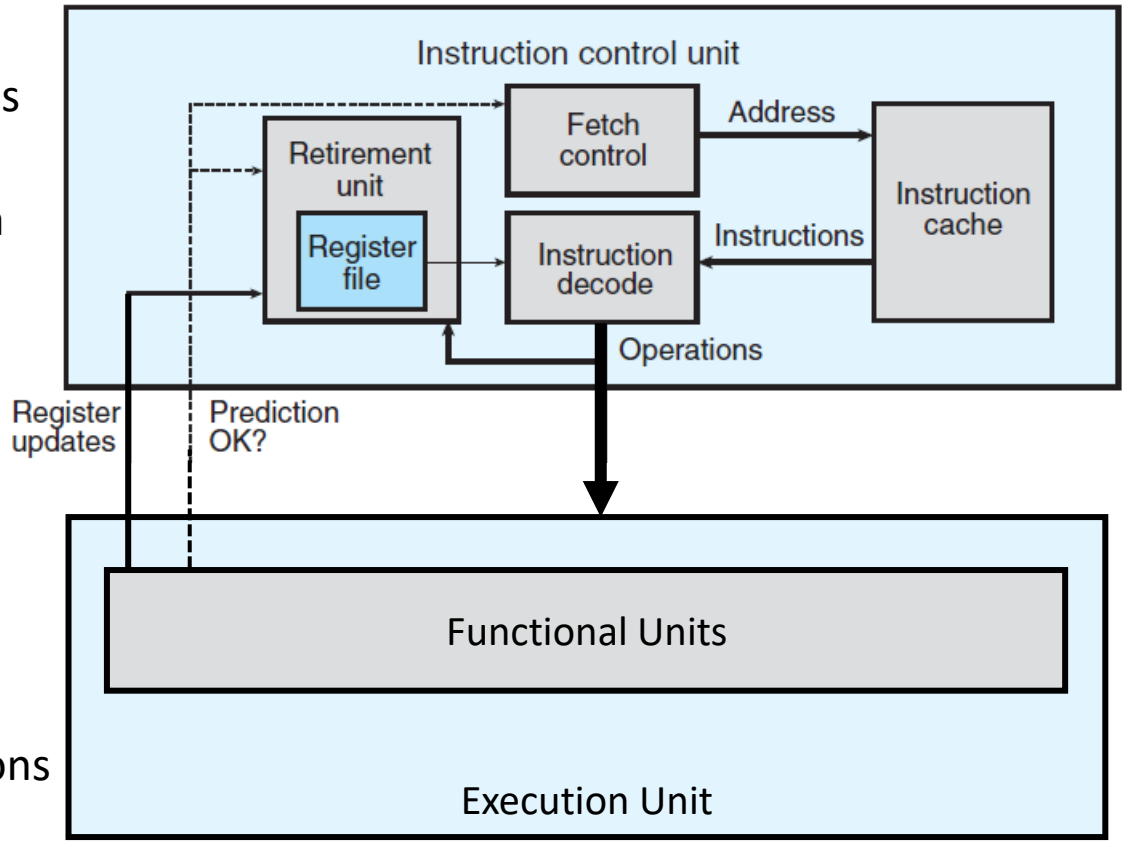
- Factors a nivell **d'Arquitectura**
  - Hw + Sw
- Factors a nivell de **Sistema Operatiu**
  - Què és el Sistema Operatiu (SO)?
    - És un software que gestiona Hw+Sw
      - Els detalls es veuran en l'últim tema



Execució d'Instruccions (*vist anteriorment...*)

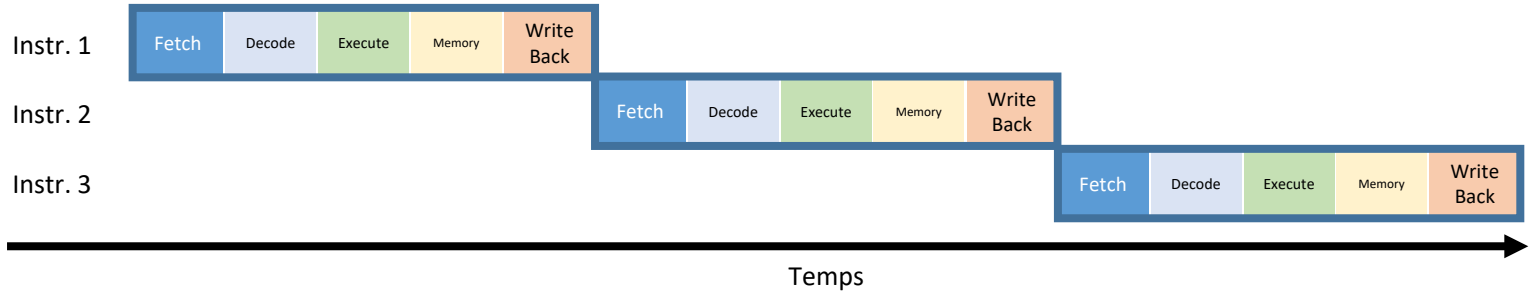


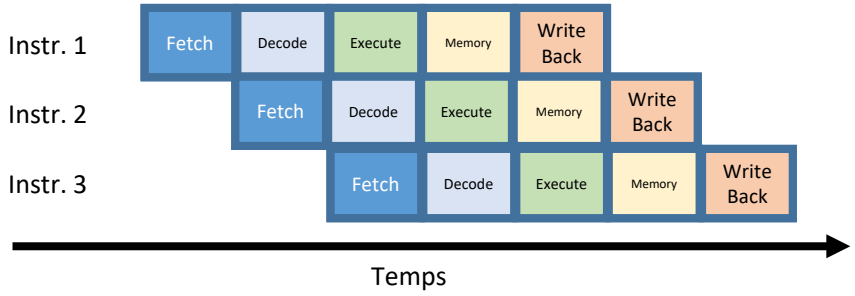
- **Instruction Control Unit...**
  - **Fetch Control:** porta la següent seqüència d'instruccions
    - Què passa si no està clar quina és la següent instrucció?
  - **Instruction Decode:** prepara l'instrucció per executar-la
    - Tradueix d'instruccions a operacions primitives o microoperacions
      - Depèn de l'ISA (Instruction Set Architecture)
    - És carreguen els operands des del **Register File**
      - És la unitat més petita i ràpida de la jerarquia de memòria
- **Execution Unit...**
  - Envia les operacions primitives a les unitats funcionals corresponents
  - Les **unitats funcionals** són les que realitzen les operacions
    - Load, Store, ALU, FP, ...
- **Tornada a la Instruction Control Unit...**
  - **Retirement Unit:** fa el seguiment de l'execució per guardar els resultats en el ordre correcte



Problemes en el Flux d'Execució

- Nivell d'Arquitectura
  - **Principi de Pipelining**
    - L'execució d'una instrucción es subdivideix en múltiples etapes
      - Els processadors actuals tenen un nombre gran d'etapes (>=15)

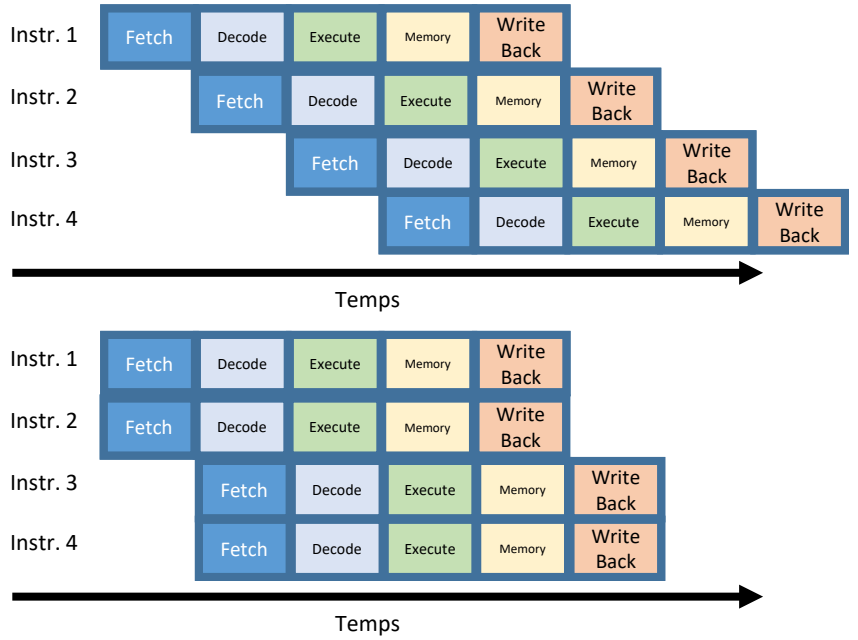




- Limitacions del concepte de pipelining
  - Particionat no-uniforme
    - El rendiment está limitat a l'etapa més lenta del pipeline
  - Cost inherent addicional
    - Comunicació entre dues etapes (registres de pipeline)

ILP: Instruction Level Parallelism

- Processador Escalar
  - Executa instruccions d'una en una
- Processador Superescalar
  - Preparar per executar >1 instruccions alhora



- ILP: Paral·lelisme a Nivell d'Instrucció
  - Executar més d'una instrucción a la vegada
    - Normalment instruccions que corresponen a la mateixa seqüència d'instruccions
    - Altres maneres d'explotar ILP pot ser executant instruccions de diferents threads

Explotar ILP

- Basic Block
  - Seqüència d'instruccions sense bifurcacions, excepte en el punt d'entrada i el punt de sortida
- Limitacions: Situacions en que una instrucción no pot ser executada quan li toca
  - Conseqüència: Introduir endarreriments
    - Intrs. NOP (software): El compilador insereix endarreriments amb instrs. que no fan res (Not OPeration)
    - Bubbles (hardware): El processador reté una o més instrs. en el pipeline fins que el perill està resolt
- Dos mètodes complementaris
  - Mètode Estàtic (realitzat pel compilador)
    - Reestructurar el codi per facilitar el paral·lelisme
  - Mètode Dinàmic (realitzat pel processador)
    - Tècniques per explotar dinàmicament ILP i resoldre dependències
      - E.g. execució fora d'ordre, prediccions

Problemes en el Flux d'Execució

- Perills (Hazards)
  - Limitacions de la pròpia organització del pipeline
  - Limitacions per restriccions estructurals: augmentar el nombre de recursos???

- Dependències
  - Limitacions del codi dels programes que s'executen
    - **Dependències de dades**: un operand depèn del resultat del càlcul anterior
    - **Dependències de control**: saber quina és la següent instrucción a ser executada depèn d'una comparació anterior
    - **Dependències de nom**: una instrucción utilitza el mateix registre o adreça de memòria que una altra instrucció, però no tenen dependència de dades

- No totes les dependències es converteixen en Hazards
  - Es converteix amb Hazard si el valor produït per una operació NO està a punt quan la instrucció que presenta la dependència intenta llegir el valor
    - Si el valor produït ja está a punt, NO es converteix amb hazard

Dependències de dades

*Assumint que l'instrucción **X** s'executa abans que l'instrucción **Y***

- RAW (Read After Write)
  - L'instrucción Y vol llegir un operand abans que l'instrucción X l'escrigui
    - Dependència real

```
ADD   R0, R1, R2     R1+R2→R0
SUB   R4, R3, R0     R3-R0→R4
```

- WAR (Write After Read)
  - L'instrucció Y genera un resultat sobre un registre que encara no ha llegit l'instrucción X
    - Anti-dependencia *(dependència de nom)*

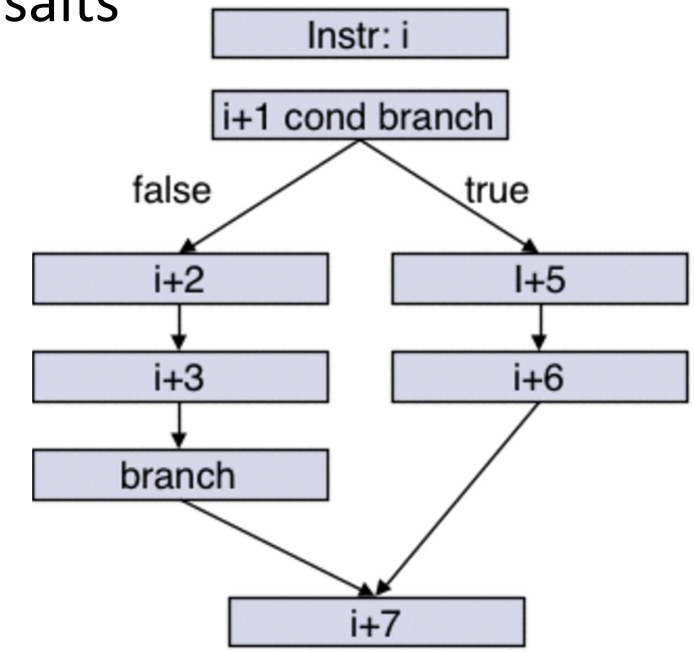
```
ADD   R0, R1, R2     R1+R2→R0
SUB   R2, R3, R4     R3-R4→R2
```

- WAW (Write After Write)
  - Les dues instruccions generan un resultat sobre el mateix registre
    - Dependència de resultat *(dependència de nom)*

```
ADD   R0, R1, R2     R1+R2→R0
SUB   R0, R3, R4     R3-R4→R0
```

Dependències de Control

- Generades per instruccions de bifurcació (Branch), com ara salts



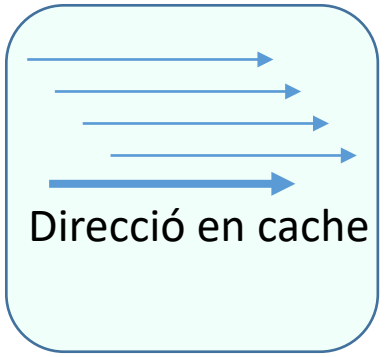
- Salts condicionals
  - Predicció del resultat de la condició
- Salts incondicionals
  - Què passa amb les instruccions que estan colocades just a continuació?
  - Introducció de NOP instruccions per afegir endarreriments des del compilador

Altres Aspectes que afecten a l'ILP

- Contenció a nivell de Hardware degut a recursos compartits entre diferents threads
  - E.g.: una unitat de coma flotant (FPUnit) compartida entre varios hardware threads en un core
- A nivell de multiprogramació (varios programas vius alhora)
  - El SO planifica quins programes poden usar la CPU i quins programes han d'esperar
    - En la sessió de laboratori es veurà una pincellada d'aquest efecte
    - Es veurà en més detall en l'últim tema de teoría
- A nivel de paral·lelisme dins d'un mateix codi
  - Esperes entre software threads
- Impacte dels fallos de cache en la jerarquia de memòria
  - Penalització per Miss: temps adicional degut a una errada en el nivel i-éssim de cache
  - Tipus d'errades de cache: **Cold, Conflict, Capacity**

Exemple del rendiment de la cache (1/6)

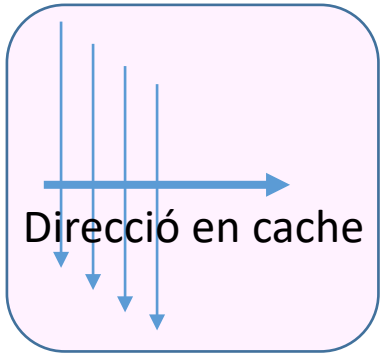
- ACCÉS(i,j)



Accés per files

- Aprofita la localitat espacial

ACCÉS(j,i)

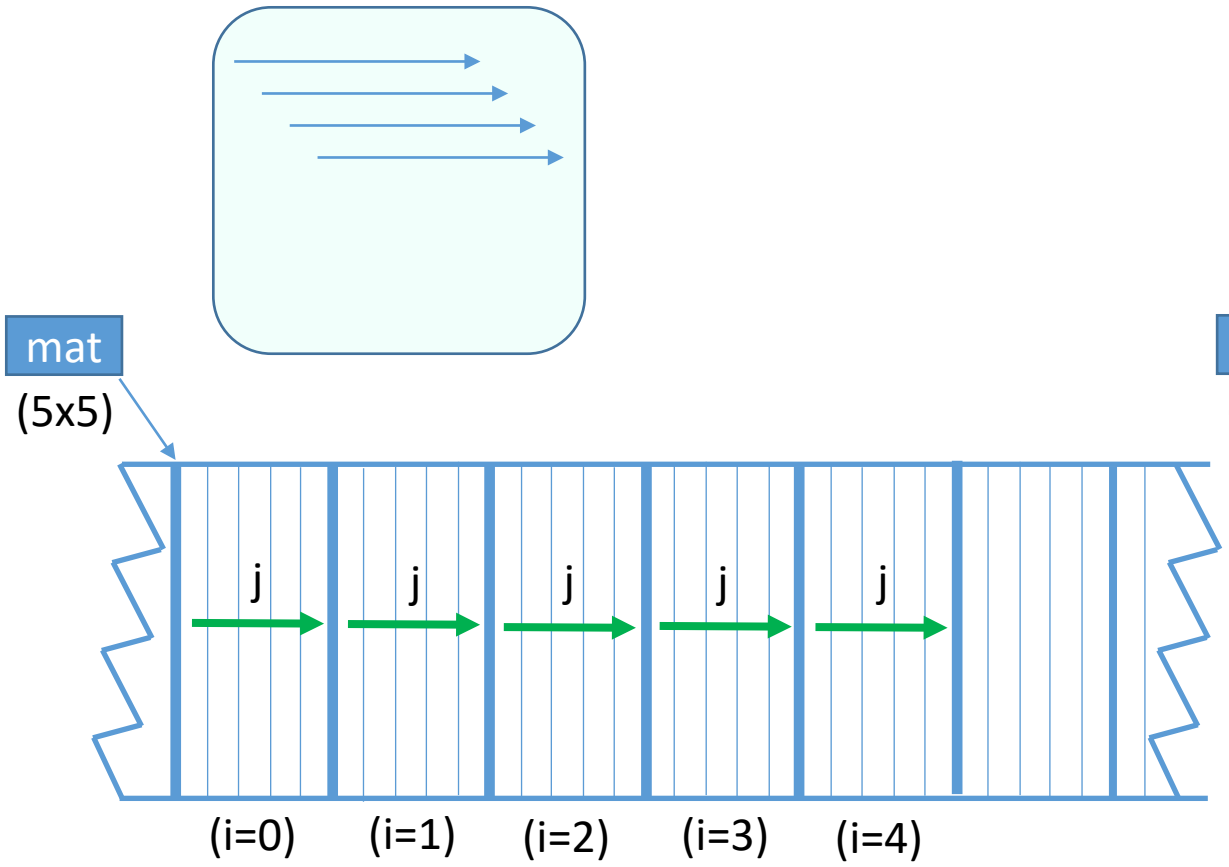


Accés per columnes
Matriu transposada

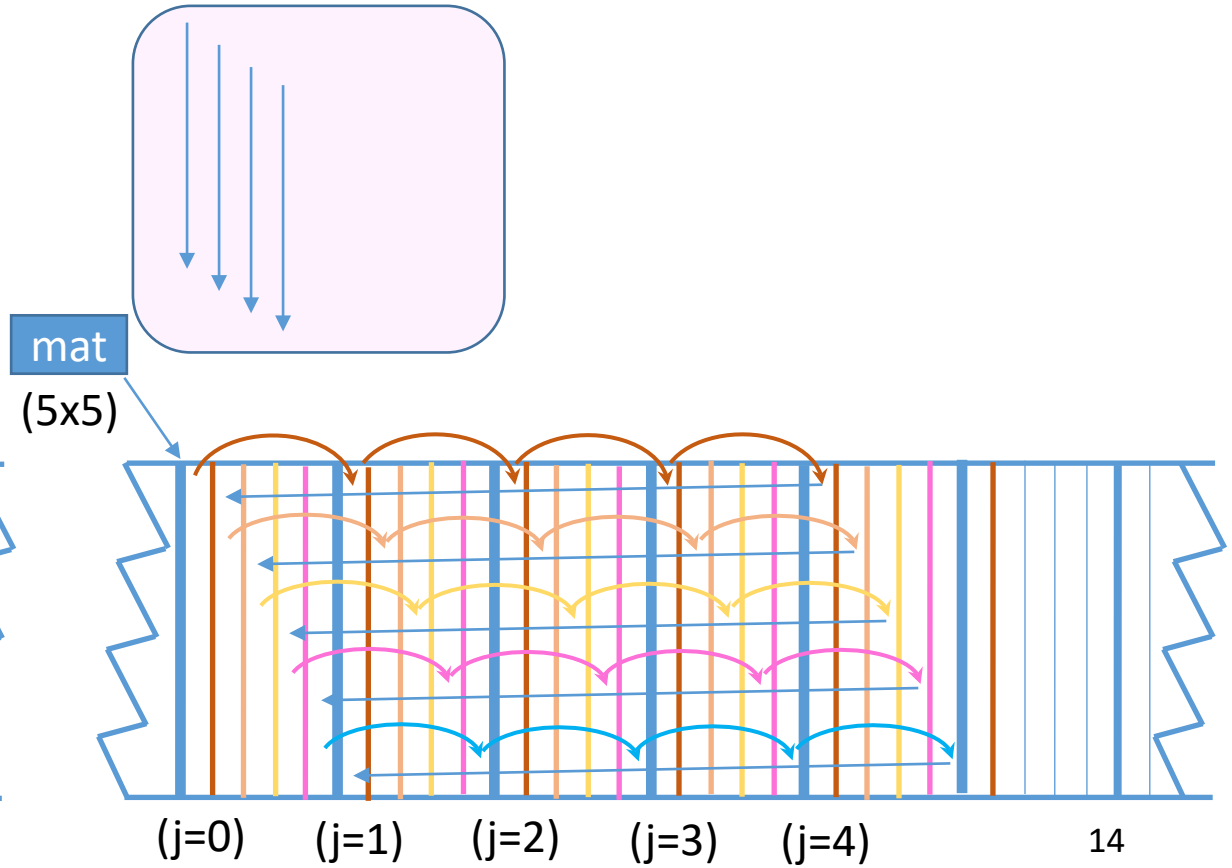
- Accessos no consecutius:
  baixa localitat espacial

Exemple del rendiment de la cache (2/6)

- ACCÉS(i,j)

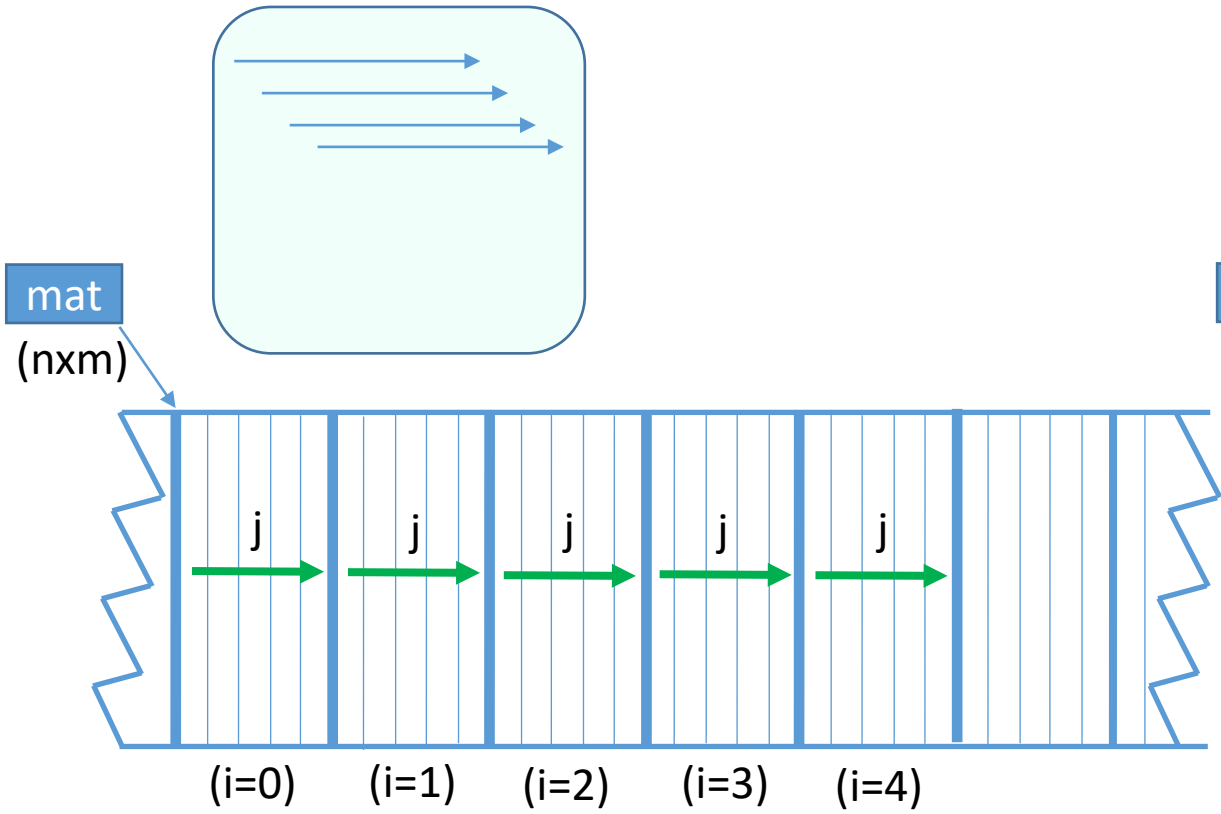


ACCÉS(j,i)



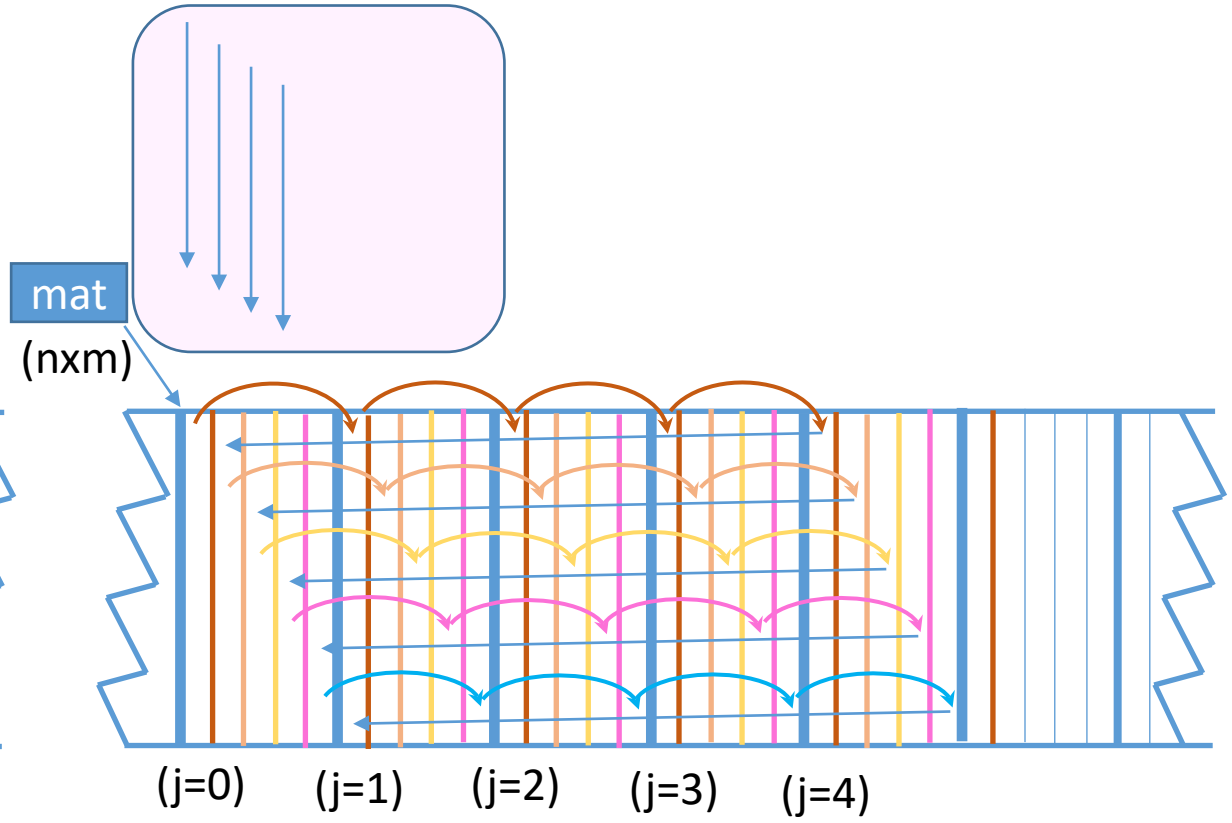
Exemple del rendiment de la cache (3/6)

- ACCÉS(i,j)



X misses

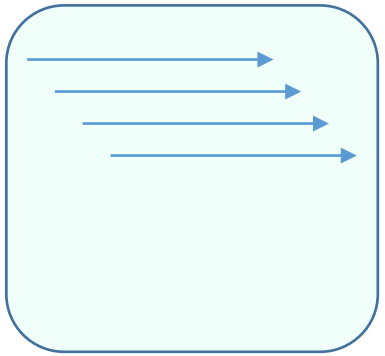
ACCÉS(j,i)



>X misses (capacity)

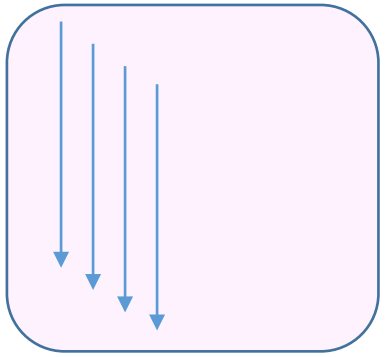
Exemple del rendiment de la cache (4/6)

- ACCÉS(i,j)



- Matriu 10240 x 10240, single precision floating point
  - Casi 105 Milions de números → 400MB
- MN4 processor, Intel(R) Xeon(R) Platinum 8160 CPU @ 2.10GHz
- Temps d'inicialització
  - loop =0: 148 ms
  - loop >0: 51ms
- Favorable a la Cache
  - Amb localitat espacial

ACCÉS(j,i)



loop =0: 465 ms

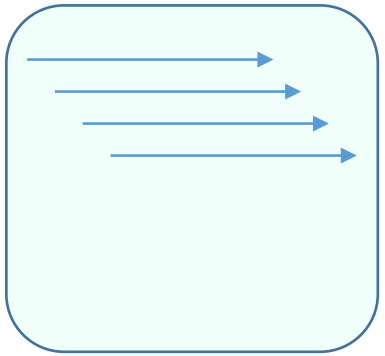
loop >0: 361ms

NO favorable a la Cache

Sense localitat espacial

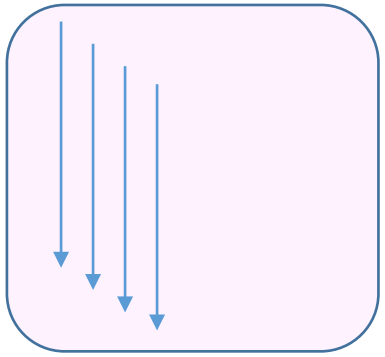
Exemple del rendiment de la cache (5/6)

- ACCÉS(i,j)



- Matriu 10240 x 10240, single precision floating point
  - Casi 105 Milions de números → 400MB
- MN4 processor, Intel(R) Xeon(R) Platinum 8160 CPU @ 2.10GHz
- Temps d'inicialització
  - **loop =0: 148 ms**
  - **loop >0: 51ms**
- Favorable a la Cache
  - Amb localitat espacial

ACCÉS(j,i)



loop =0: 465 ms

loop >0: 361ms

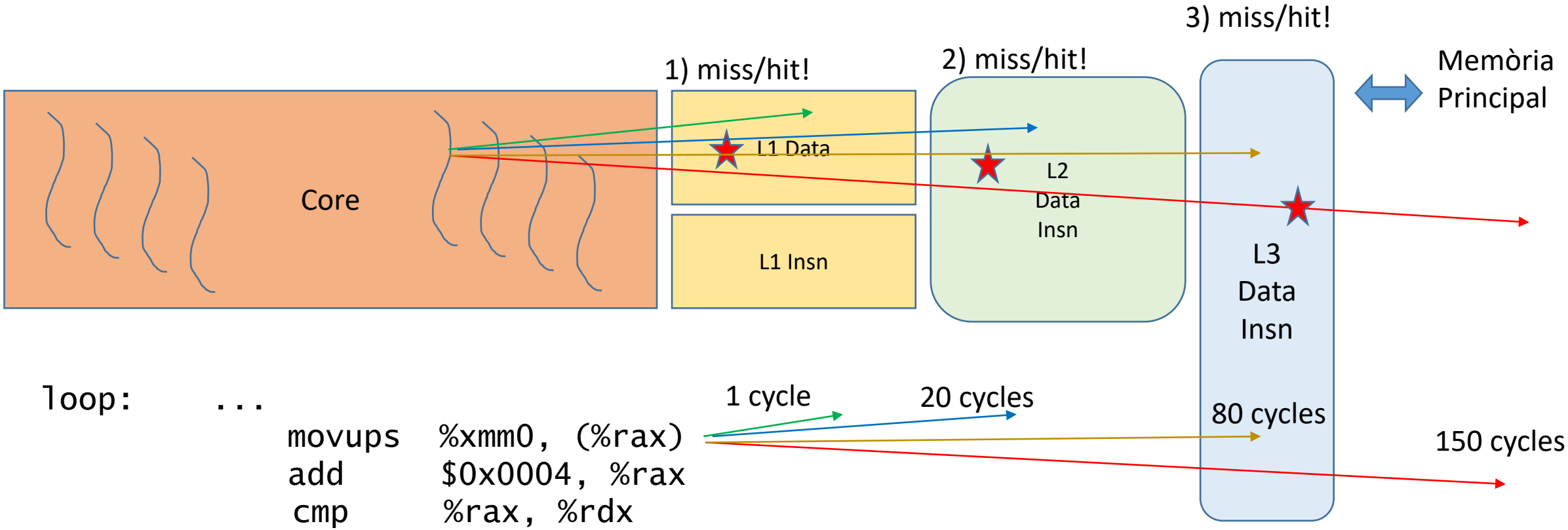
NO favorable a la Cache

Sense localitat espacial

La primera iteració del loop presenta un temps superior a la resta d'iteracions degut a aspectes de la gestió de memòria per part del Sistema Operatiu

Exemple del rendiment de la cache (6/6)

- Accessos a cache




```
loop:    ...
         movups   %xmm0, (%rax)
         add      $0x0004, %rax
         cmp      %rax, %rdx
         jne      loop
         ...
```

Mètriques de Rendiment

- Basat en l'arquitectura
  - CPI: cicles per instruccció
    - CPI = $(C_i+C_b) / C_i = 1 + C_b/C_i$ → CPI = 1 + penalitzacions
      - C_i: instruccions
      - C_b: bubbles
      - C_b/C_i: promig de bubbles per instrucció
  - IPC: instruccions per cicle
    - IPC = #instruccions/cicle = 1/CPI

Mètriques de Rendiment

- Temps d'execució
  - Temps mesurat amb un temporitzador (rellotge)
  - Qualsevol esdeveniment Hardware pot fer augmentar el temps gastat per l'aplicació
    - Requereixen serveis del SO per tractar-los (e.g. moure el ratolí)
  - ... Així com també esdeveniments del SO
    - **Multiprogramació**
- Temps de CPU
  - Quantitat de temps utilitzant la CPU (hardware thread), en mode usuari i/o sistema
    - Mode sistema: codi del sistema operatiu (e.g. escriure un caràcter per pantalla)
- Augment de rendiment (Speed-up)
  - Relació entre el temps d'execució original i el temps d'execució amb alguna optimització

Mètriques de Rendiment

- Ampla de Banda (bytes/s)
  - Relació entre la quantitat de dades transmeses per unitat de temps
- Latència (s)
  - Quantitat de temps empleat per completar operacions o comunicacions
- Rendiment: Throughput (elements/s)
  - Nombre d'operacions (o aplicacions) completades per unitat de temps
- **Eficiència energètica (Operacions / Watt)**
  - **Nombre d'operacions completades per cada watt d'energia consumit**
- I moltes altres més...

- **Millora en el rendiment: speed-up**

Estadístiques

- Mitjana
  - Mètrica més estable i realista
  - Aritmètica:
    - Suma de les mostres i dividit pel nombre de mostres
  - Altres mitjanes:
    - Mitjana Harmònica (promig de ratis)
    - Mitjana Geomètrica (quan es comparen diferents elements amb diferents rangs numèrics)
- Desviació Estàndar
  - Indica quanta variació hi ha entre els resultats
  - $s = \sqrt{\frac{\sum_{i=0}^{N-1} (x_i - \bar{x})^2}{N-1}}$

Bibliografia

- Computer Systems – A Programmer's perspective (3rd Edition)
  - Randal E. Bryant, David R. O'Hallaron, Person Education Limited, 2016
  - https://discovery.upc.edu/permalink/34CSUC_UPC/11q3oqt/alma991004062589706711
    - Several Chapters

- Computer Organization and Design (5th Edition)
  - D. Patterson, J. Hennessy, and P. Alexander
  - http://cataleg.upc.edu/record=b1431482~S1*cat
    - Several Chapters