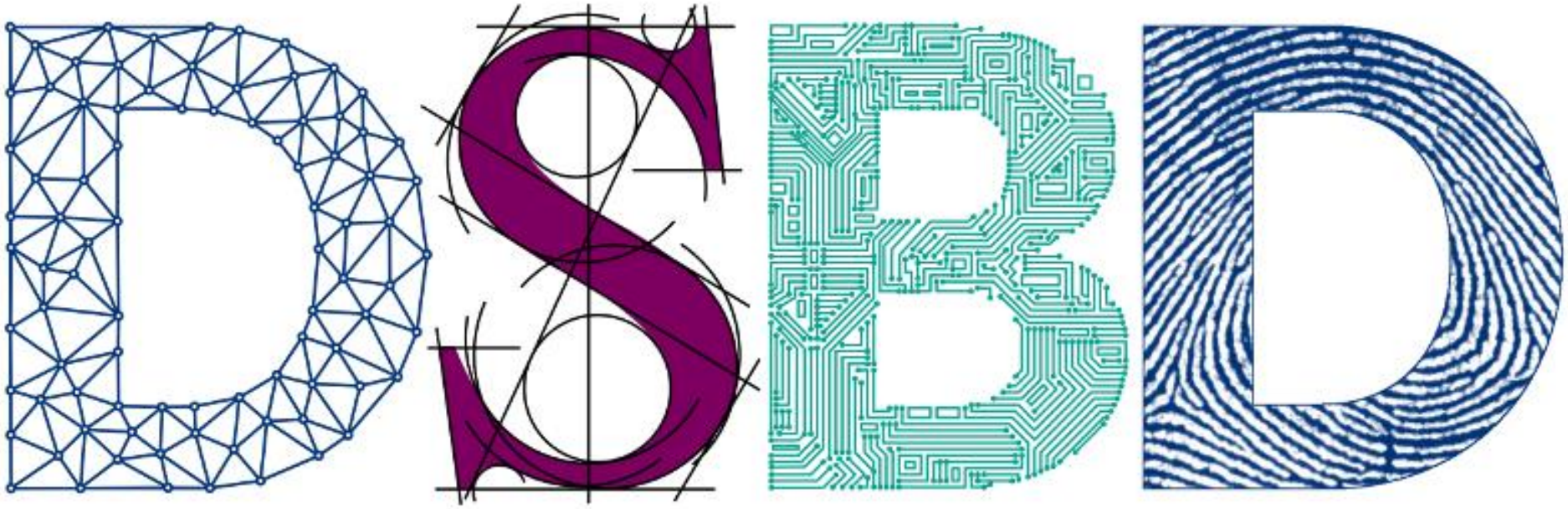
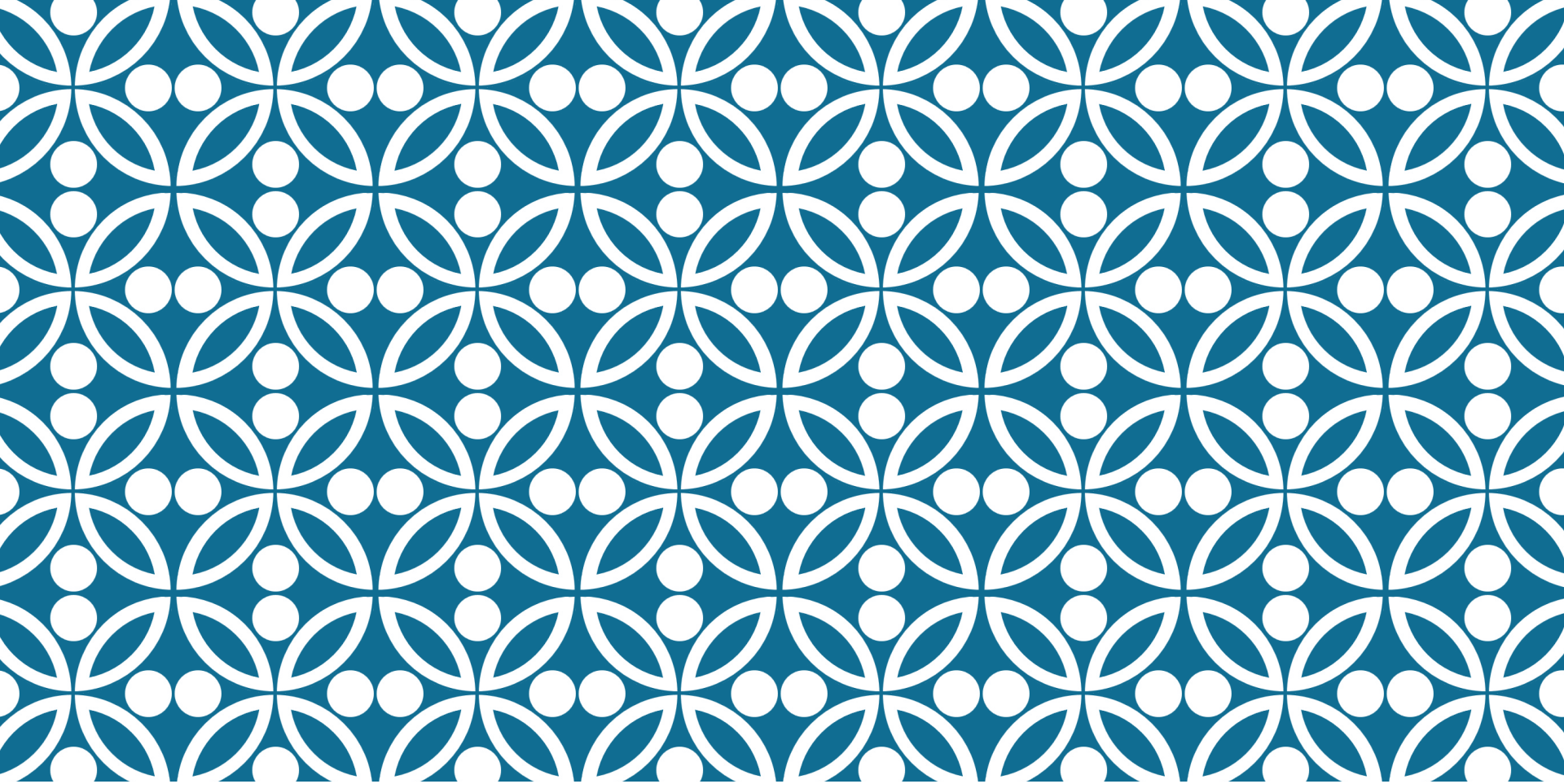


MEMÓRIAS CACHE

Infraesrutura Computacional Pt.3

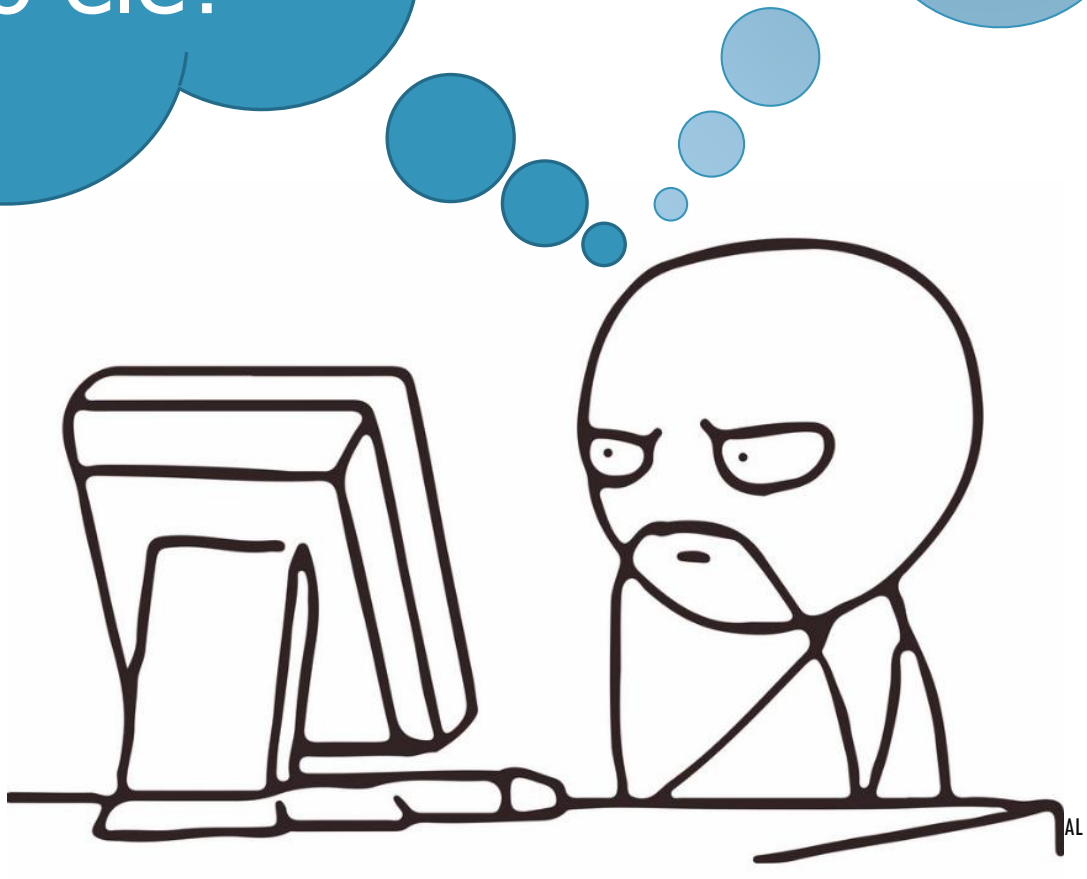
Marco A. Z. Alves

[Francesc Alted'11 - Blosc Author - Barcelona Music and Audio Technology (BMAT)]



O PROBLEMA DE ACESSO A DADOS POR QUE AS CPUS MODERNAS ESTÃO PASSANDO FOME?

Meu computador está com fome, como eu alimento ele?



CITAÇÃO DE 1993

“Continuamos a nos beneficiar de enormes aumentos na velocidade dos microprocessadores sem aumentos proporcionais na velocidade da memória. Isso significa que o desempenho "bom" está se aproximando mais dos bons padrões de acesso à memória e da reutilização cuidadosa dos operandos. Ninguém pode pagar um sistema de memória rápido o suficiente para satisfazer todas os acessos (de memória) imediatamente, de modo que os fornecedores dependem de caches, entrelaçamento e outros dispositivos para oferecer um desempenho razoável de memória ”.

– Kevin Dowd, after his book “High Performance Computing”, O’Reilly & Associates, Inc, 1993

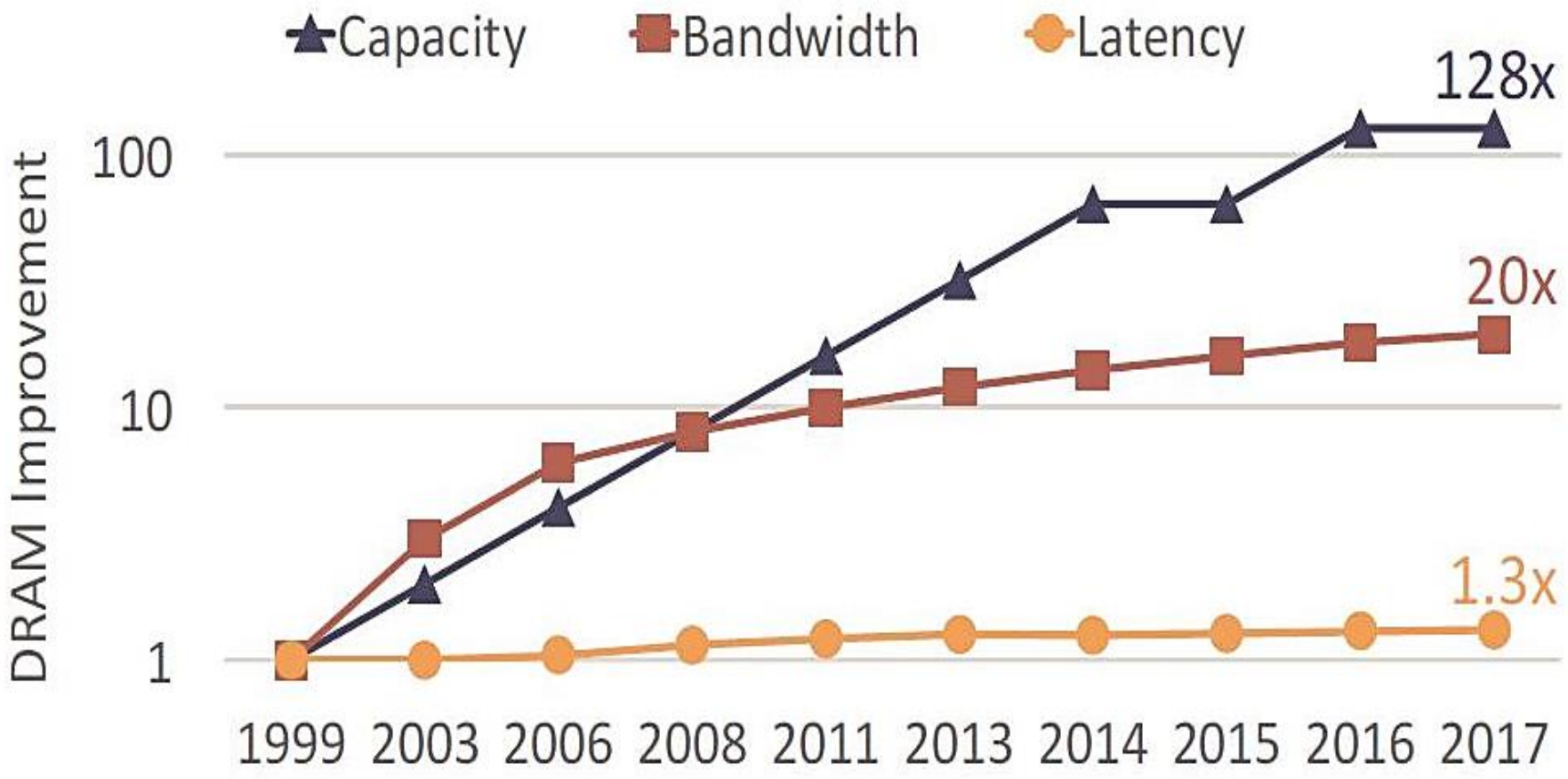
CITAÇÃO DE 1996

"Em todo os setores, os chips atuais são capazes de executar códigos mais rapidamente do que podemos alimentá-los com instruções e dados. Não há mais gargalos de desempenho no multiplicador de ponto flutuante ou em ter apenas uma única unidade inteira. A ação de design real está nos subsistemas de memória - caches, barramentos, largura de banda e latência."

"Na próxima década, o design do subsistema de memória será a única questão de design importante para os microprocessadores."

– Richard Sites, after his article "It's The Memory, Stupid!", Microprocessor Report, 10(10),1996

TENDÊNCIA DE MEMÓRIA



OGAWA, Tadashi "Understanding and Improving the Latency of DRAM-Based Memory Systems", PhD Thesis, 2017
Advisor (s): Onur Mutlu

O PROBLEMA DA FOME DA CPU

Conheça os fatos (em 2011):

- A latência de memória é muito mais lenta (cerca de 250x) do que os processadores e tem sido um gargalo essencial nos últimos quinze anos.
- A taxa de transferência de memória está melhorando em uma taxa melhor do que a latência de memória, mas também é muito mais lenta que os processadores (cerca de 25x).

O resultado é que as CPUs em nossos computadores atuais estão sofrendo de um sério problema de fome por dados: elas poderiam consumir (muito!) mais dados do que o sistema pode fornecer.

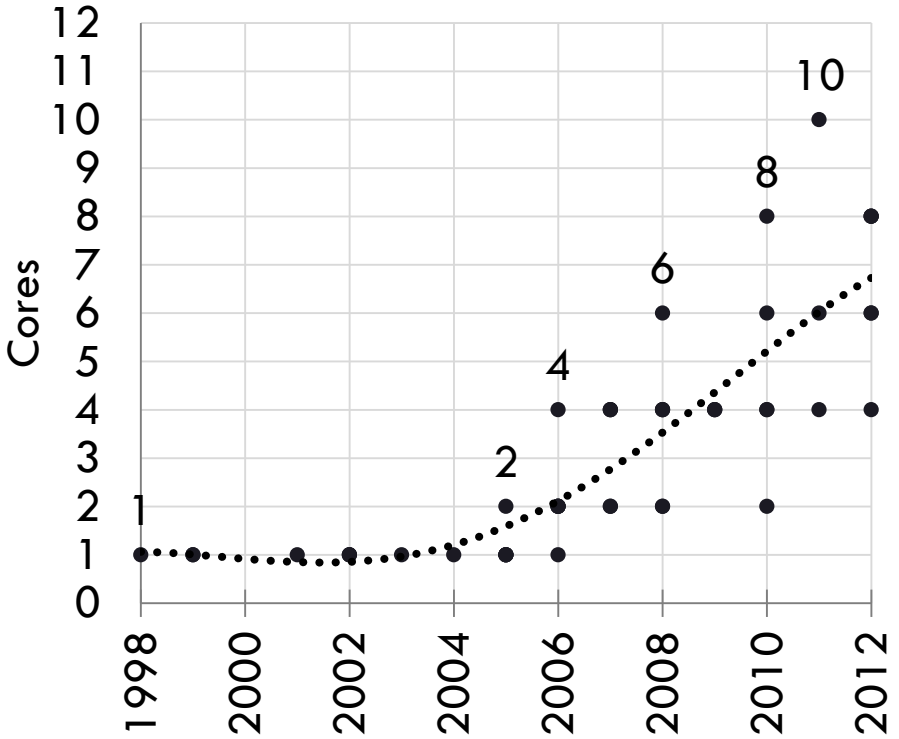
O QUE A INDÚSTRIA ESTÁ FAZENDO PARA ALIVIAR A FOME DA CPU?

Eles estão melhorando a largura de banda da memória: barato de implementar (mais dados são transmitidos em cada ciclo de clock).

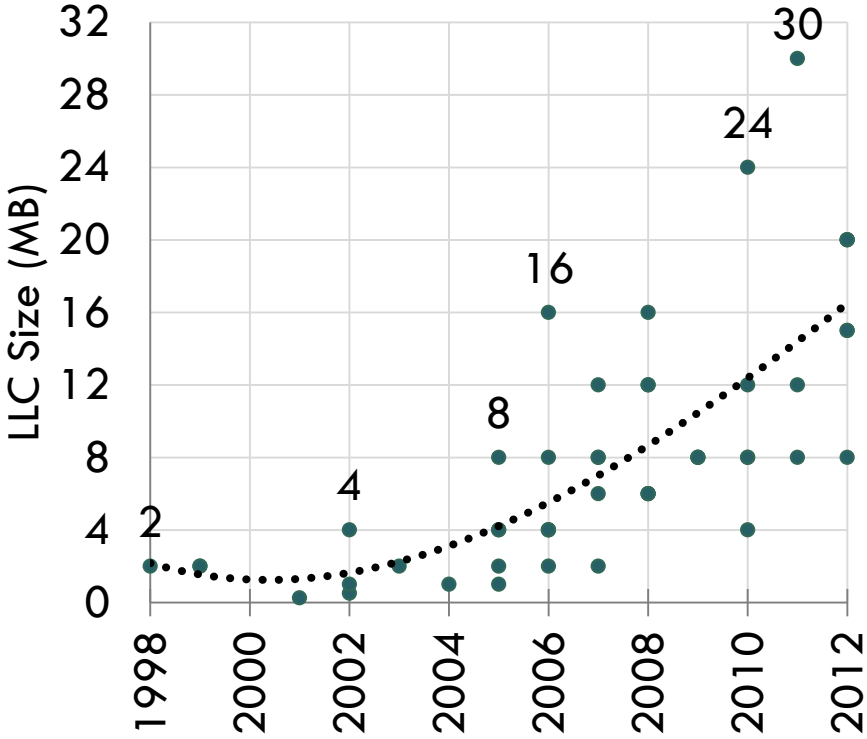
Eles estão adicionando grandes caches de dados na CPU.

EVOLUÇÃO DOS PROCESSADORES (INTEL XEON)

Número de Núcleos



Tamanho da Last Level Cache (LLC)



POR QUE UMA CACHE É ÚTIL?

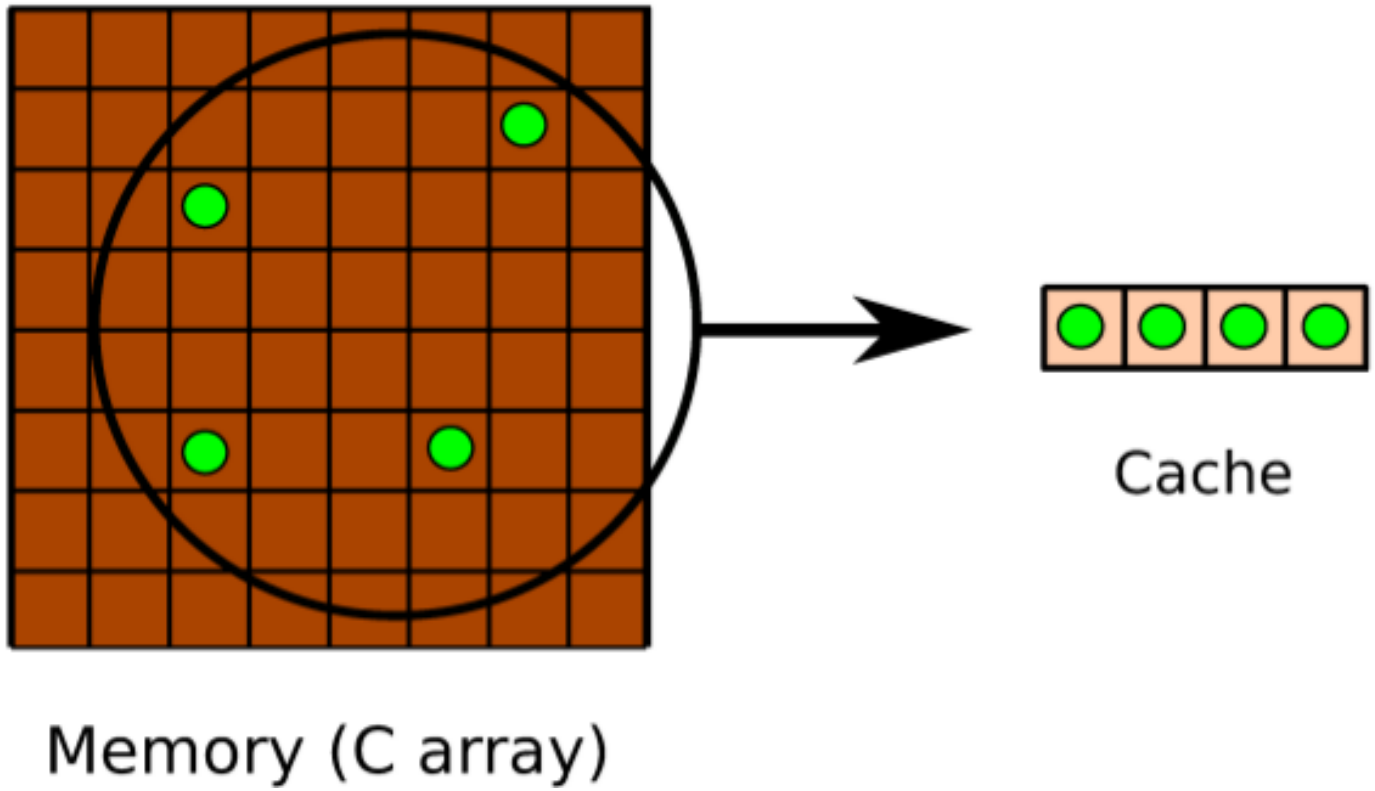
Os caches estão mais próximos do processador (normalmente no mesmo chip), portanto, tanto a latência quanto a taxa de transferência são melhoradas.

Essa caches são eficazes principalmente em alguns cenários:

- **Localidade temporal**: quando o conjunto de dados é reutilizado.
- **Localidade espacial**: quando o conjunto de dados é acessado seqüencialmente.

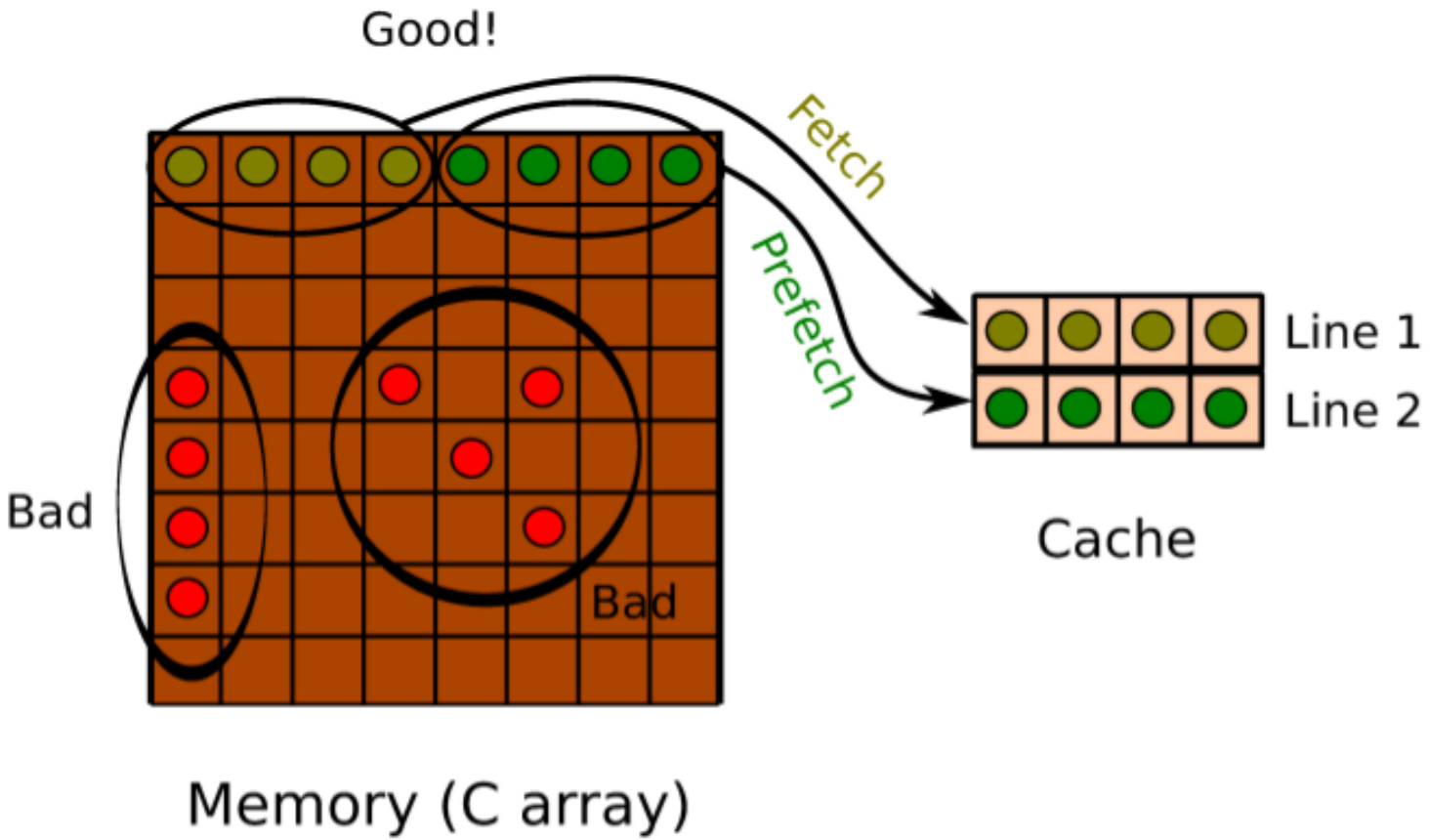
LOCALIDADE TEMPORAL

Partes do conjunto de dados são reutilizadas



LOCALIDADE ESPACIAL

O conjunto de dados é acessado sequencialmente



O MODELO DE MEMÓRIA HIERÁRQUICA

Introduzido pela indústria para lidar com problemas de fome de dados da CPU.

Consiste em ter várias camadas de memória com diferentes capacidades:

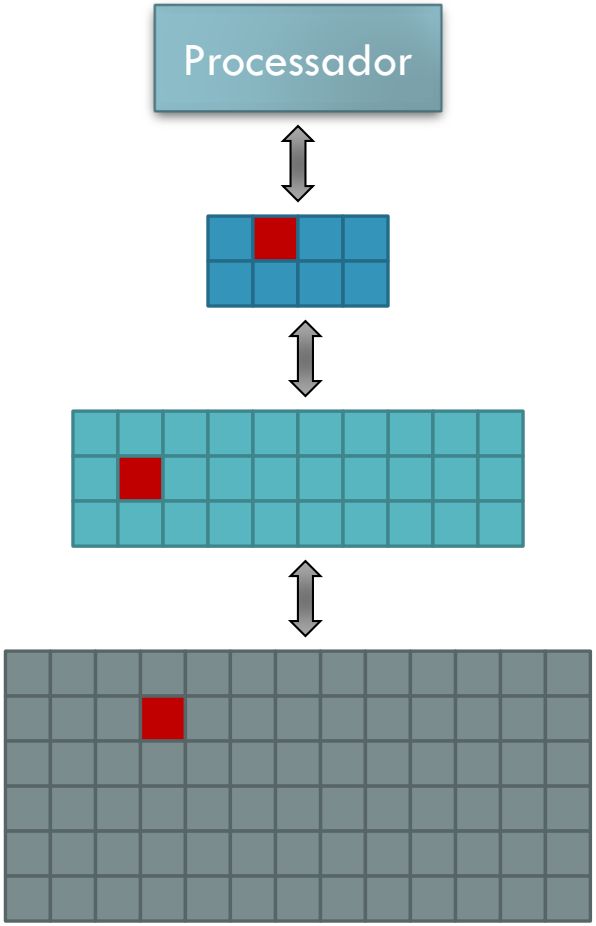
Níveis mais baixos (isto é, perdem para a CPU) têm maior velocidade, mas capacidade reduzida. Mais adequado para realizar cálculos.

Níveis mais altos têm velocidade reduzida, mas maior capacidade. Mais adequado para fins de armazenamento.

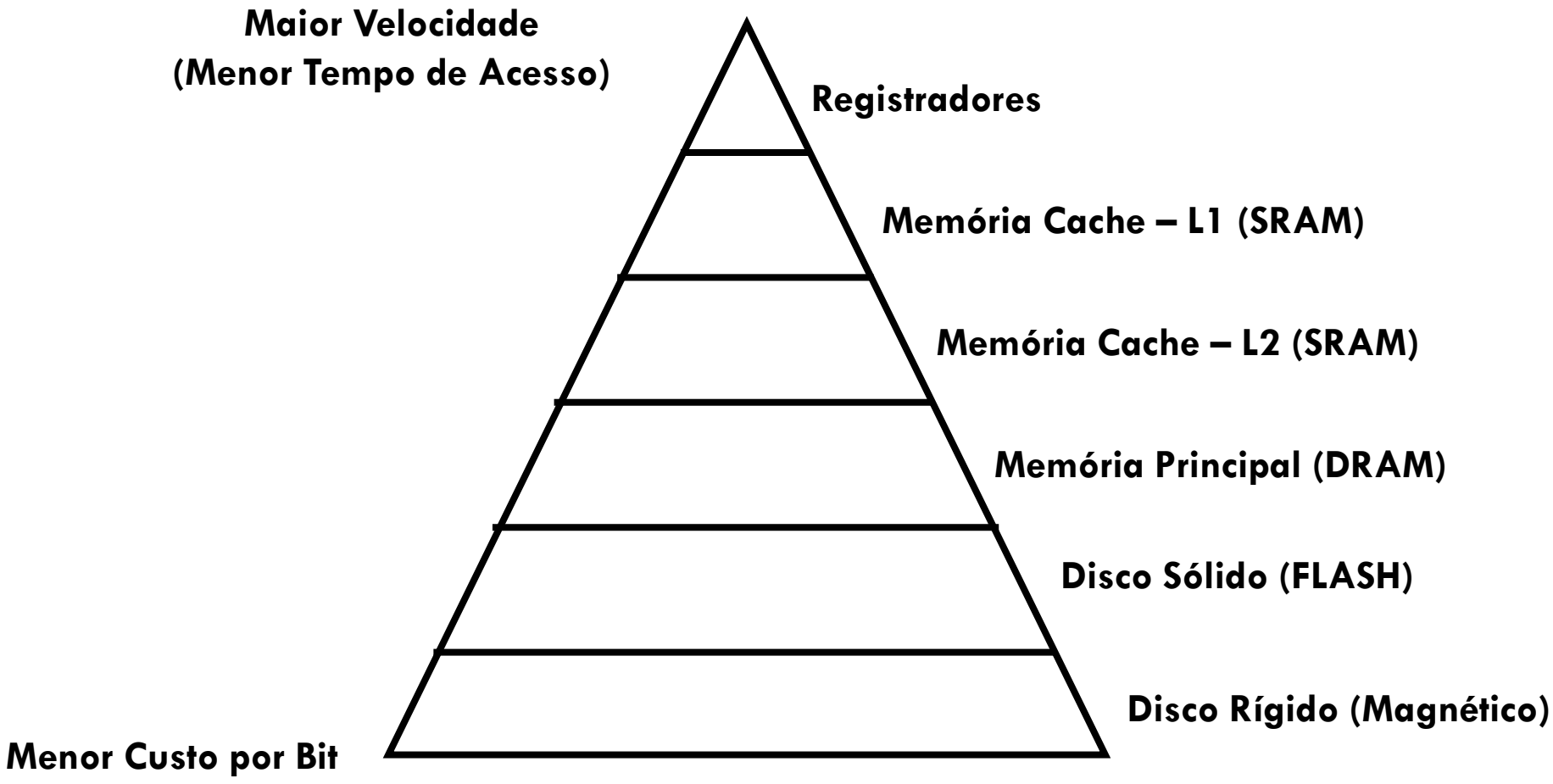
HIERARQUIA DE MEMÓRIA

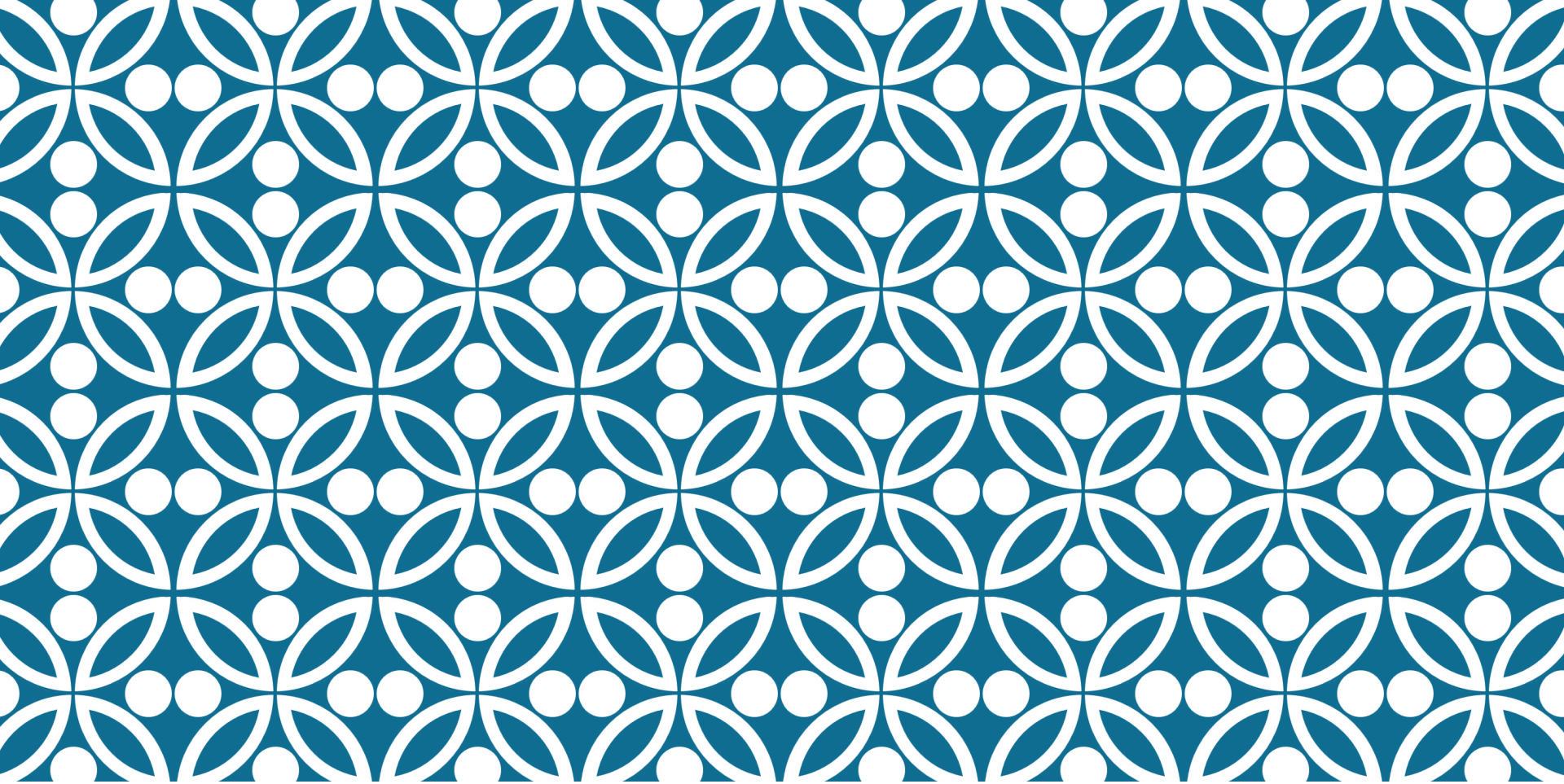
Objetivo: Oferecer ilusão de máximo tamanho de memória, com custo mínimo e velocidade máxima

Cada nível pode conter uma cópia de parte da informação armazenada no nível superior seguinte



NÍVEIS DA HIERARQUIA DE MEMÓRIA





TÉCNICAS PARA COMBATER A FOME DE DADOS

ERA UMA VEZ...

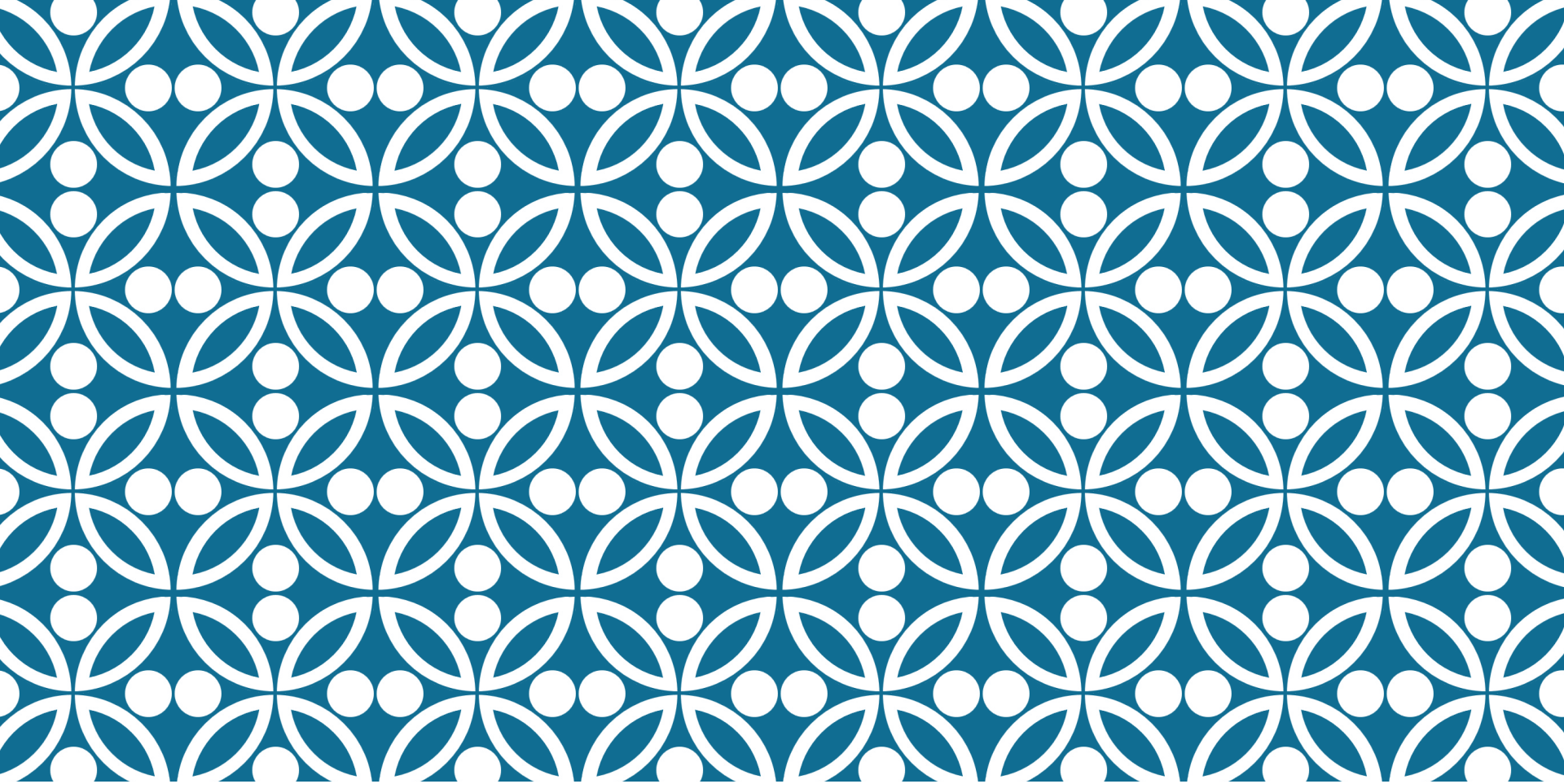
Nos anos 70 e 80, muitos cientistas computacionais tiveram que aprender a linguagem assembly para extrair todo o desempenho de seus processadores.

Nos bons e velhos tempos, o processador era o principal gargalo.

HOJE EM DIA...

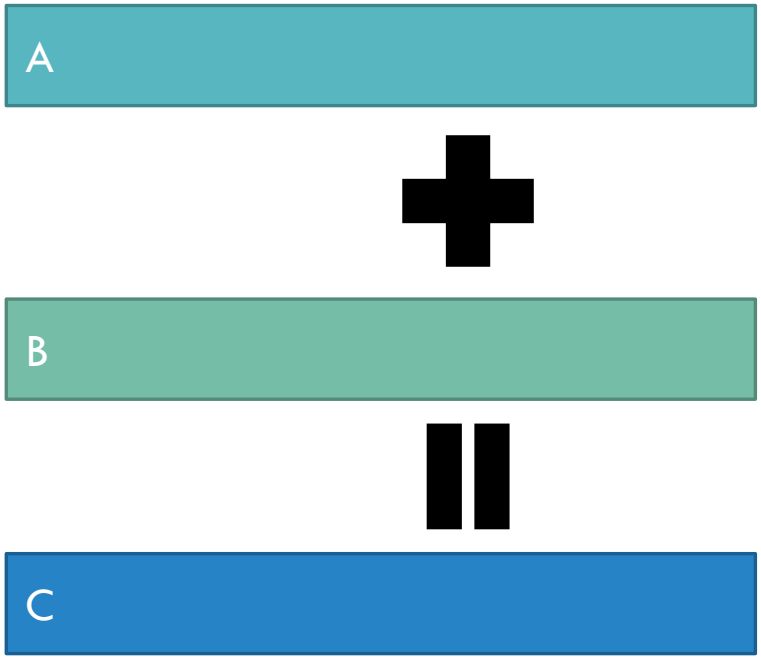
Todo cientista ~~da computação~~ deve adquirir um bom conhecimento do modelo de memória hierárquica (e suas implicações) se quiser que seus aplicativos sejam executados a uma velocidade decente (ou seja, eles não querem que suas CPUs parem demais).

A organização da memória tornou-se agora o fator chave para otimizar.

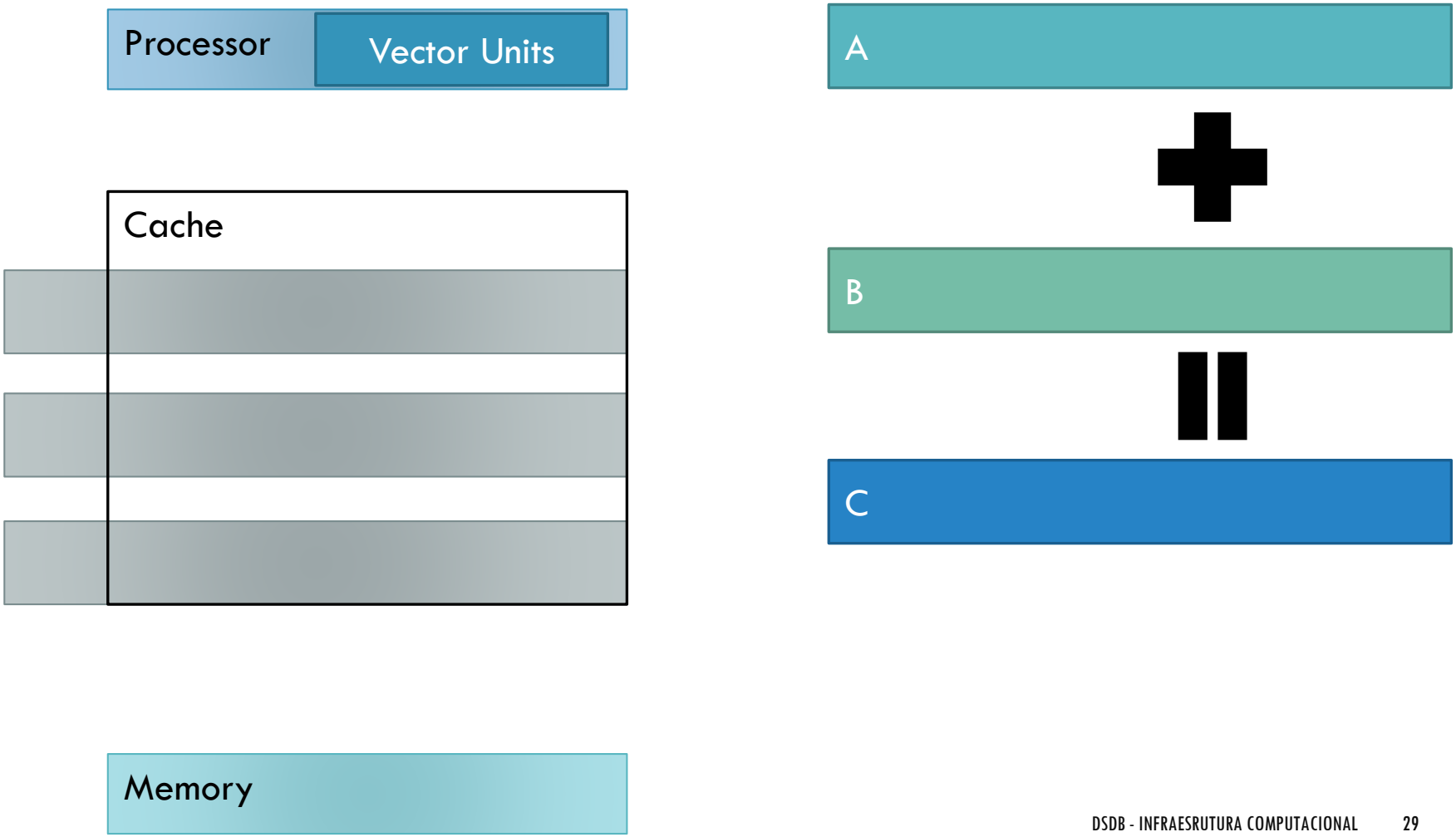


EXEMPLO DE FUNCIONAMENTO DA CACHE

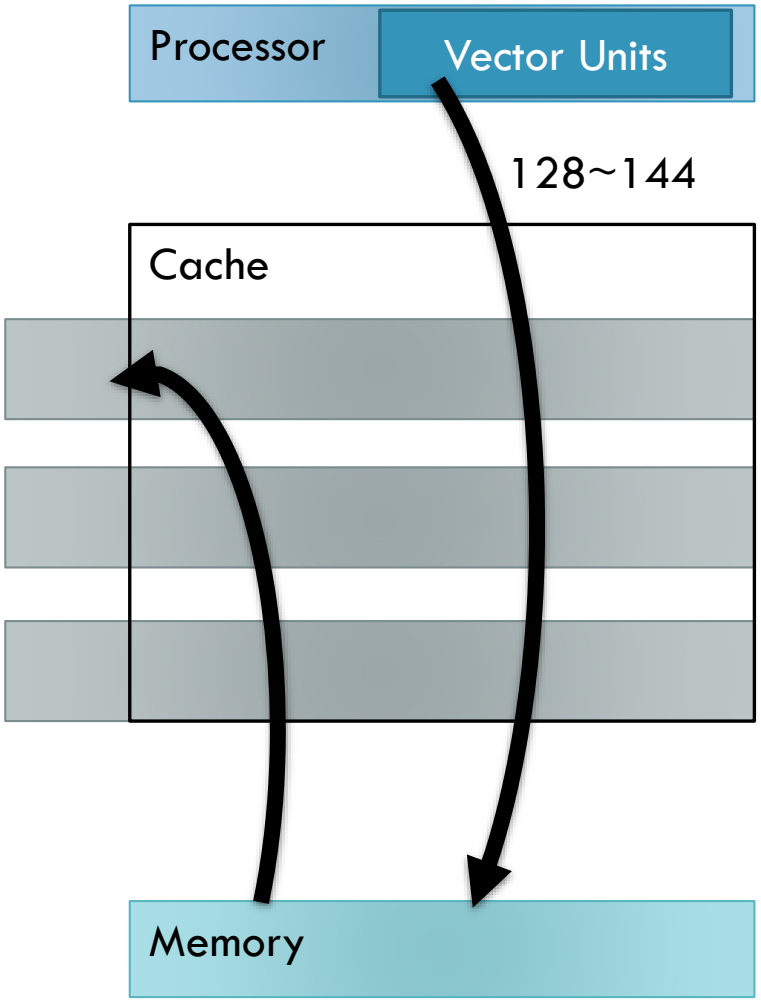
FUNCIONAMENTO – SOMA VETORIAL

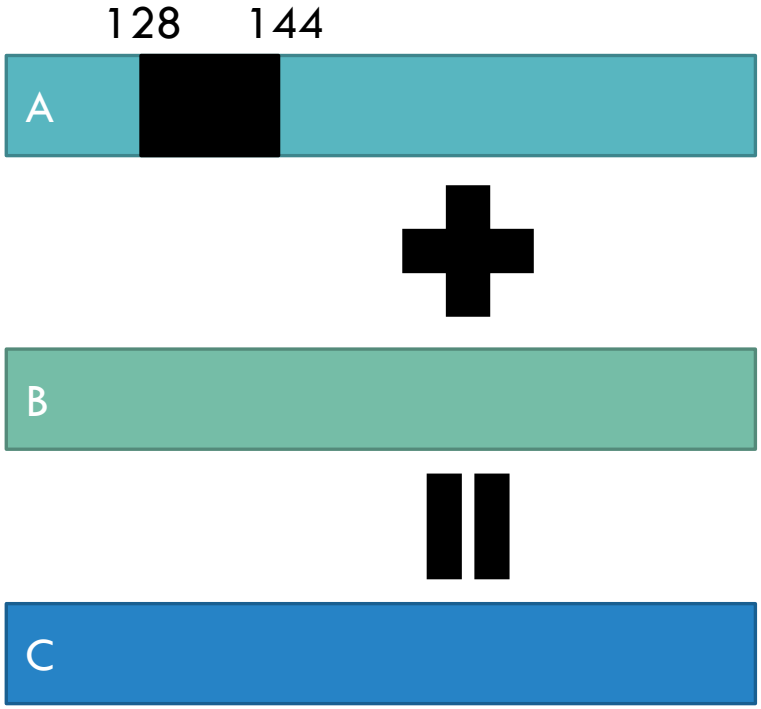


FUNCIONAMENTO – SOMA VETORIAL

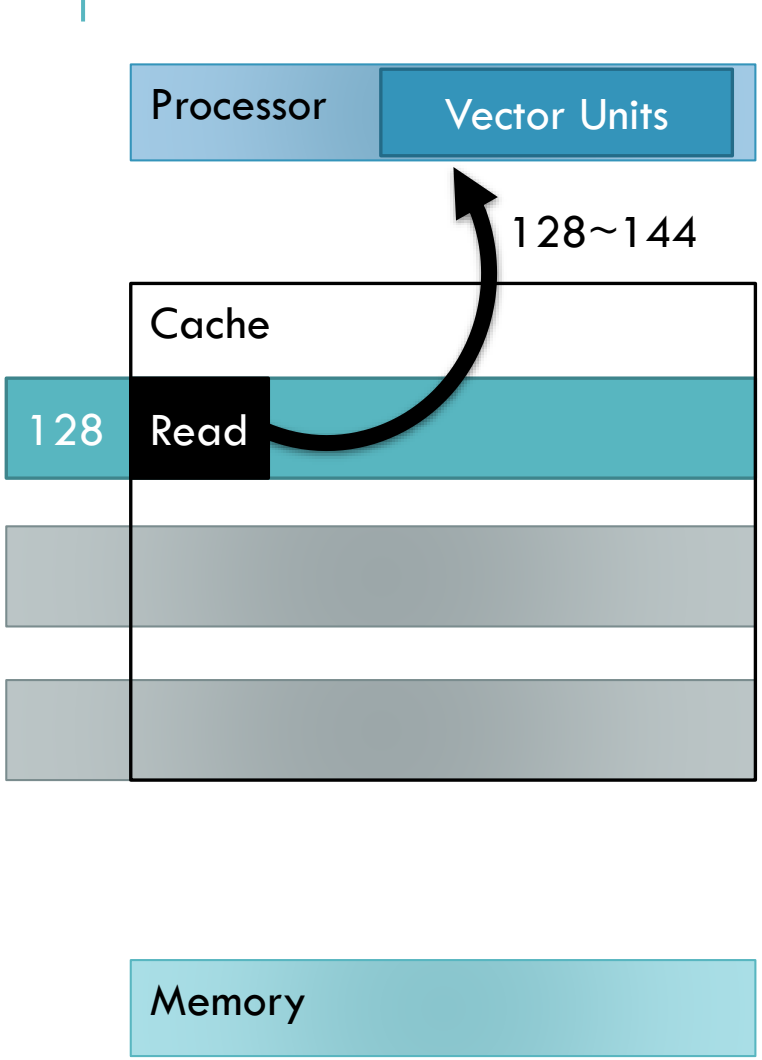


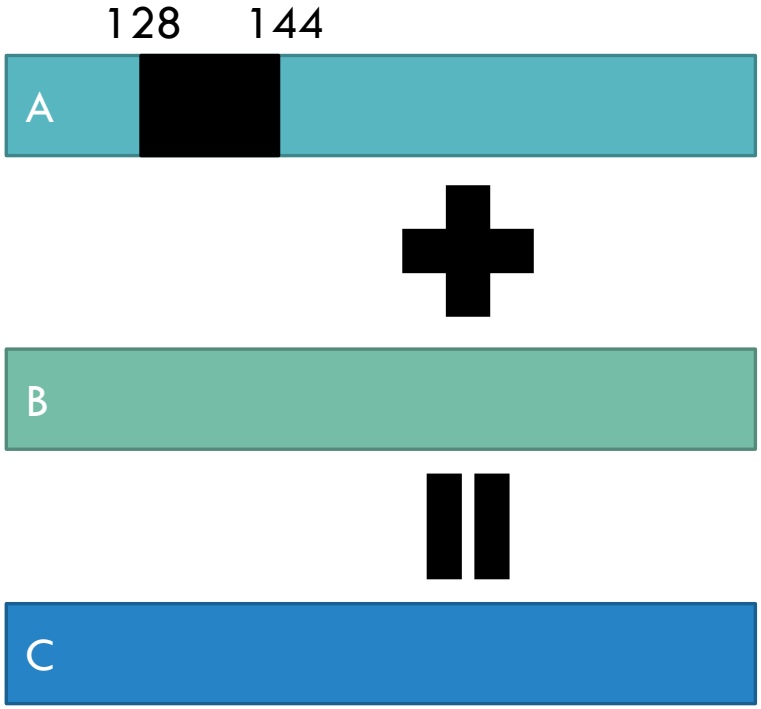
FUNCIONAMENTO – SOMA VETORIAL



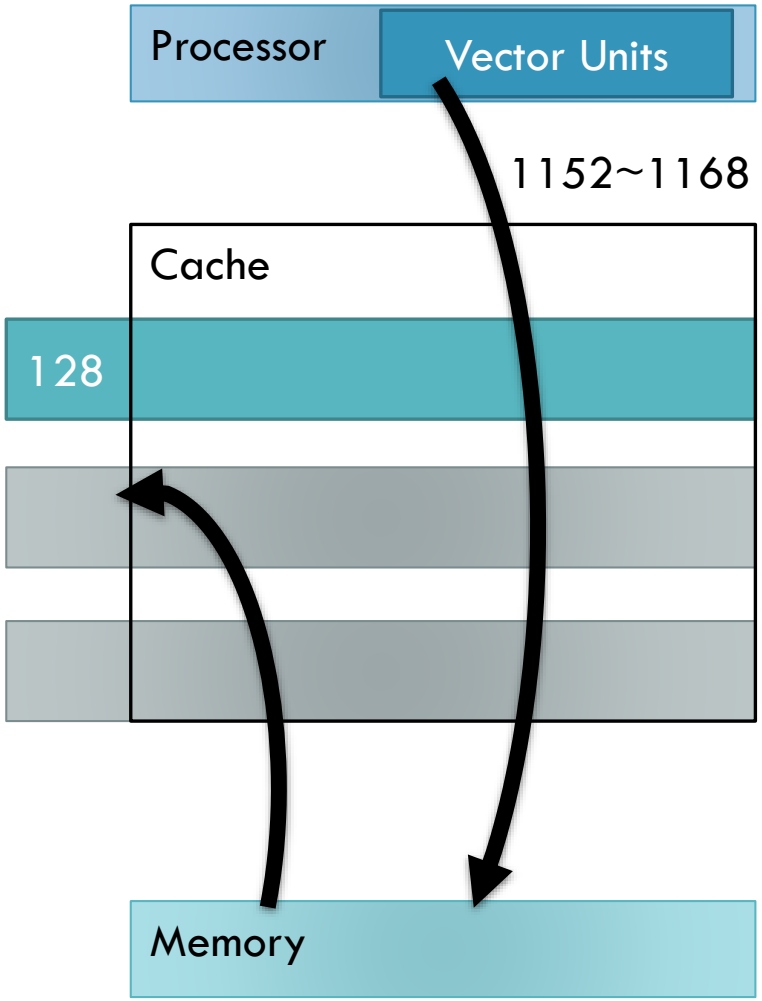


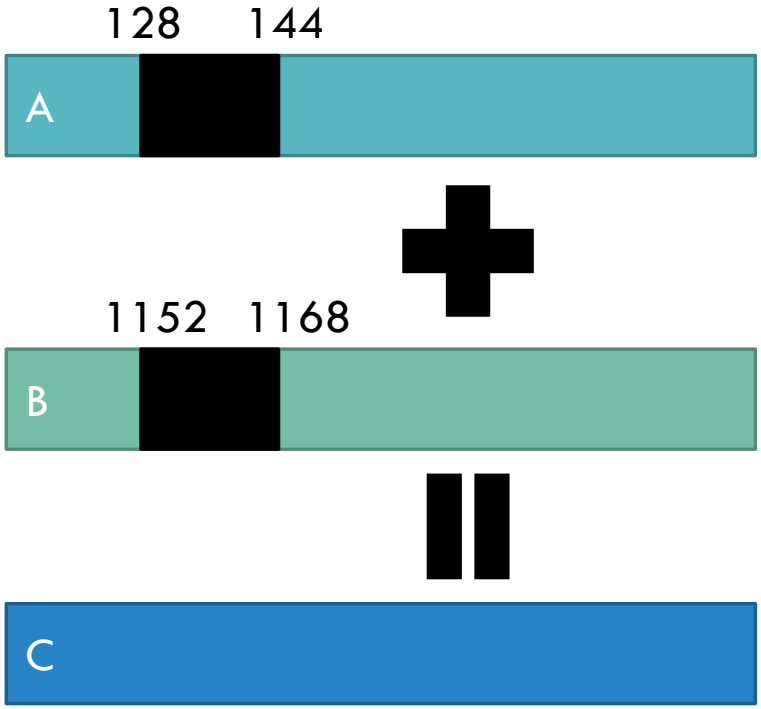
FUNCIONAMENTO – SOMA VETORIAL



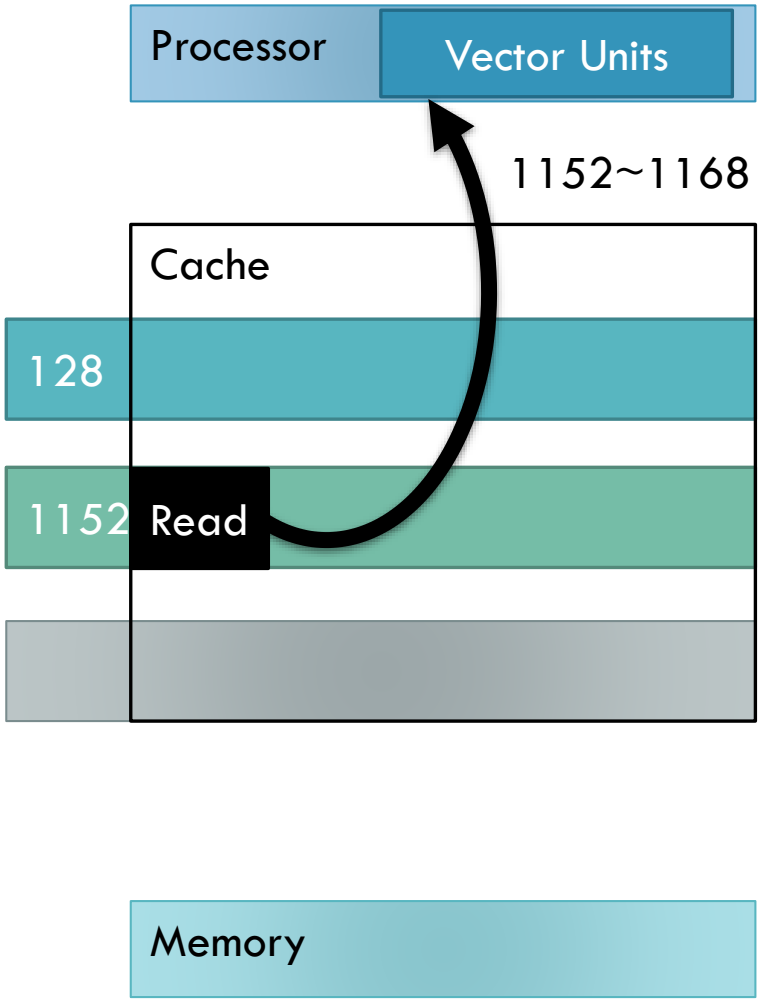


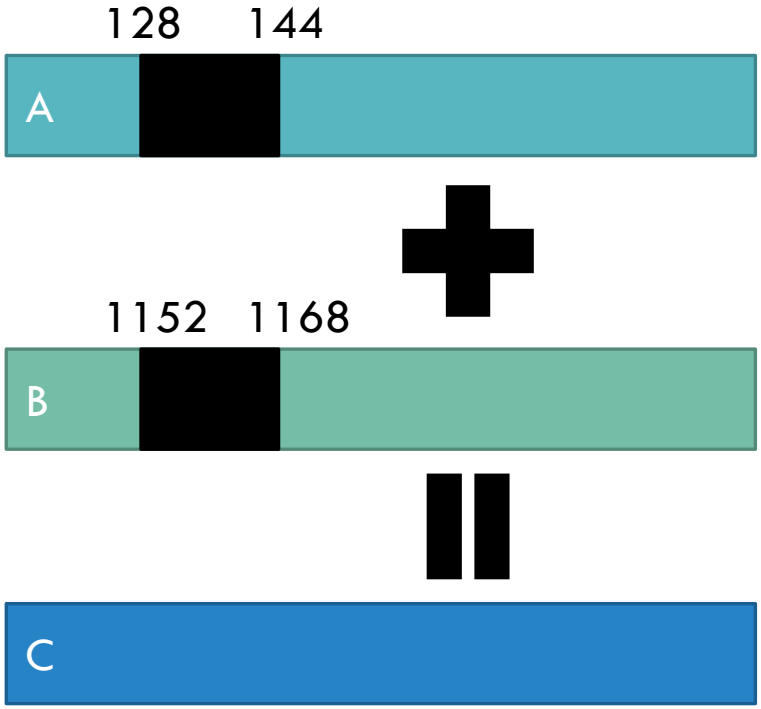
FUNCIONAMENTO – SOMA VETORIAL



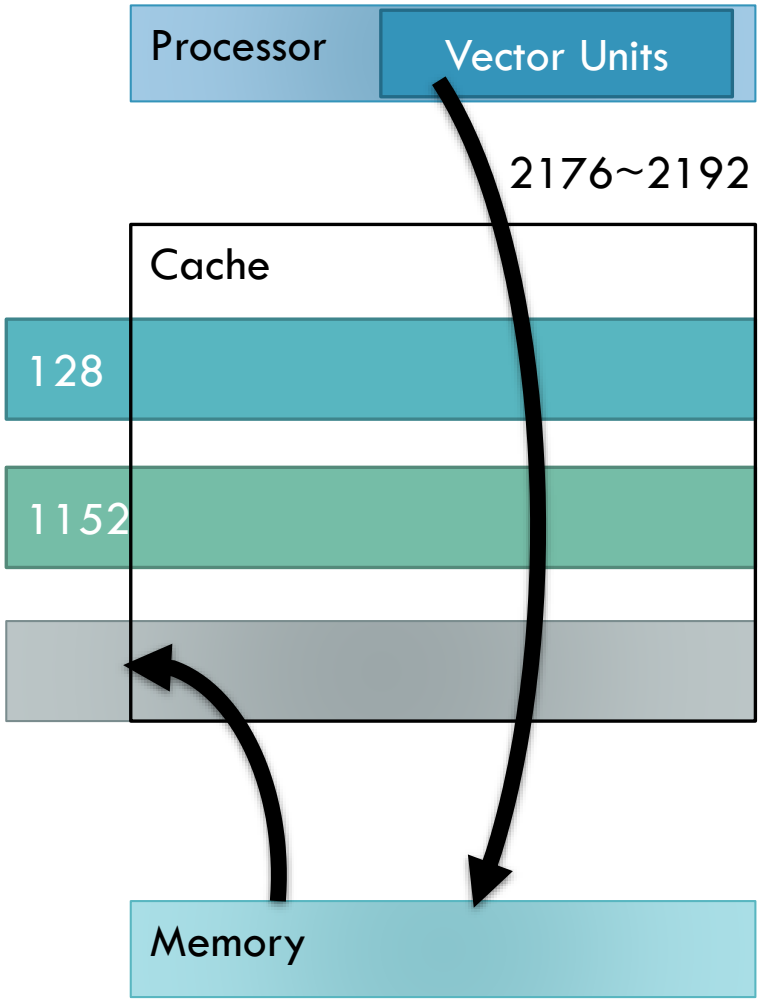


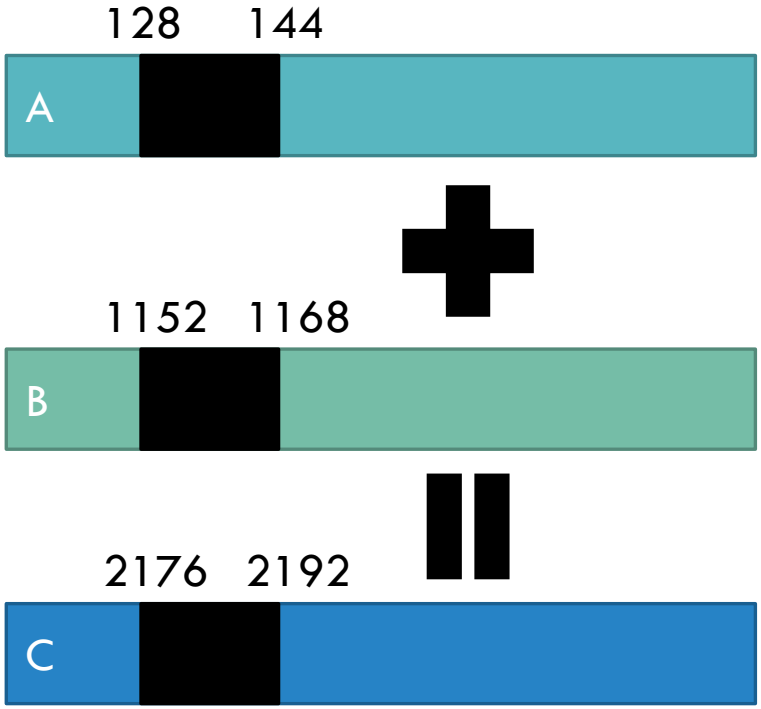
FUNCIONAMENTO – SOMA VETORIAL



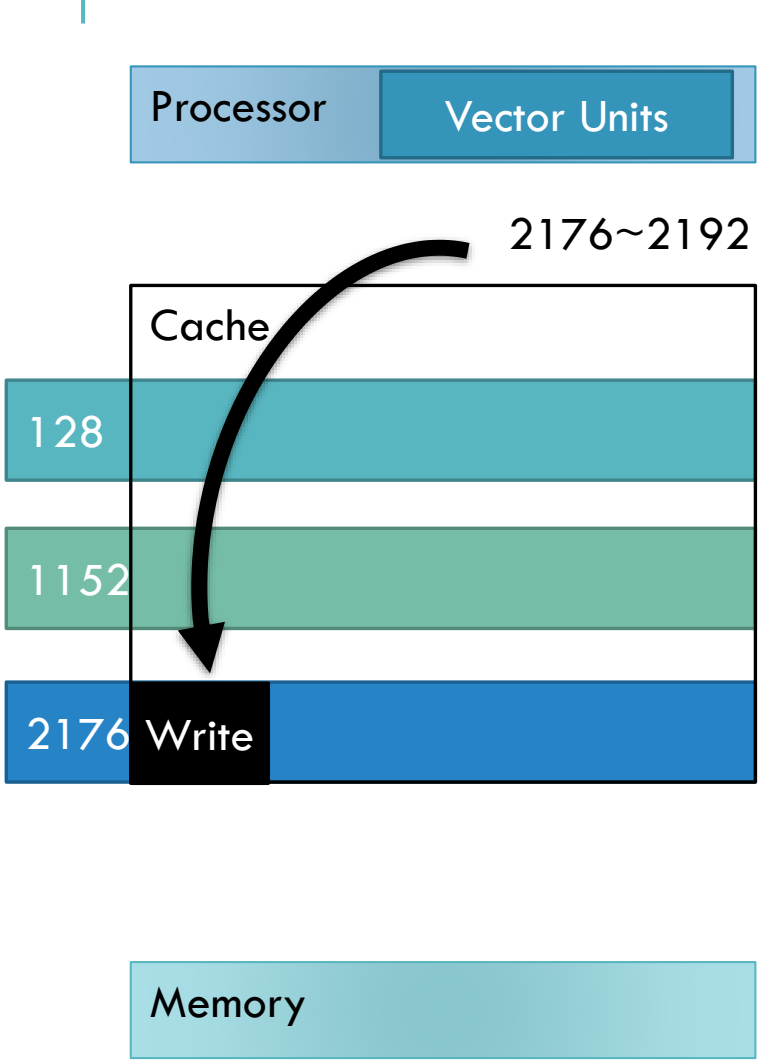


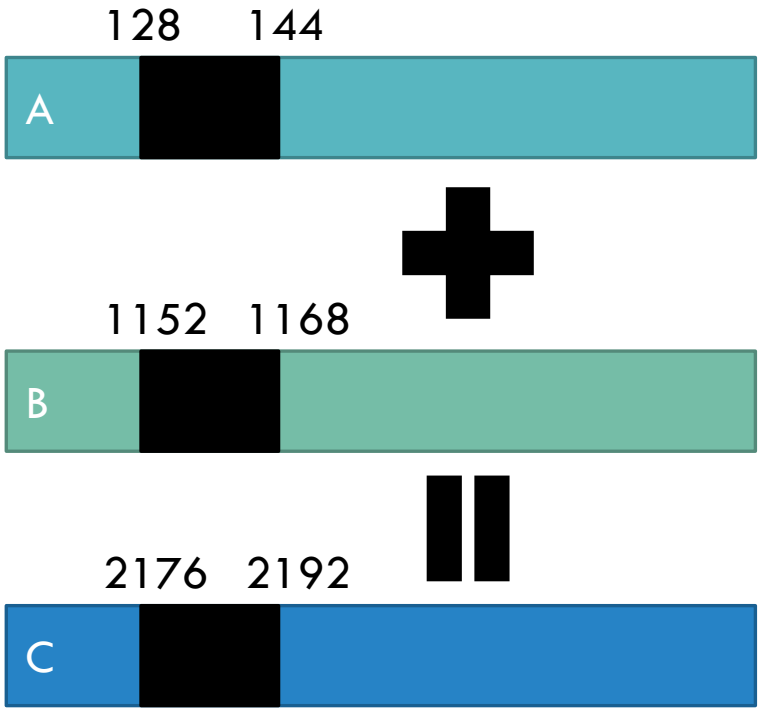
FUNCIONAMENTO – SOMA VETORIAL



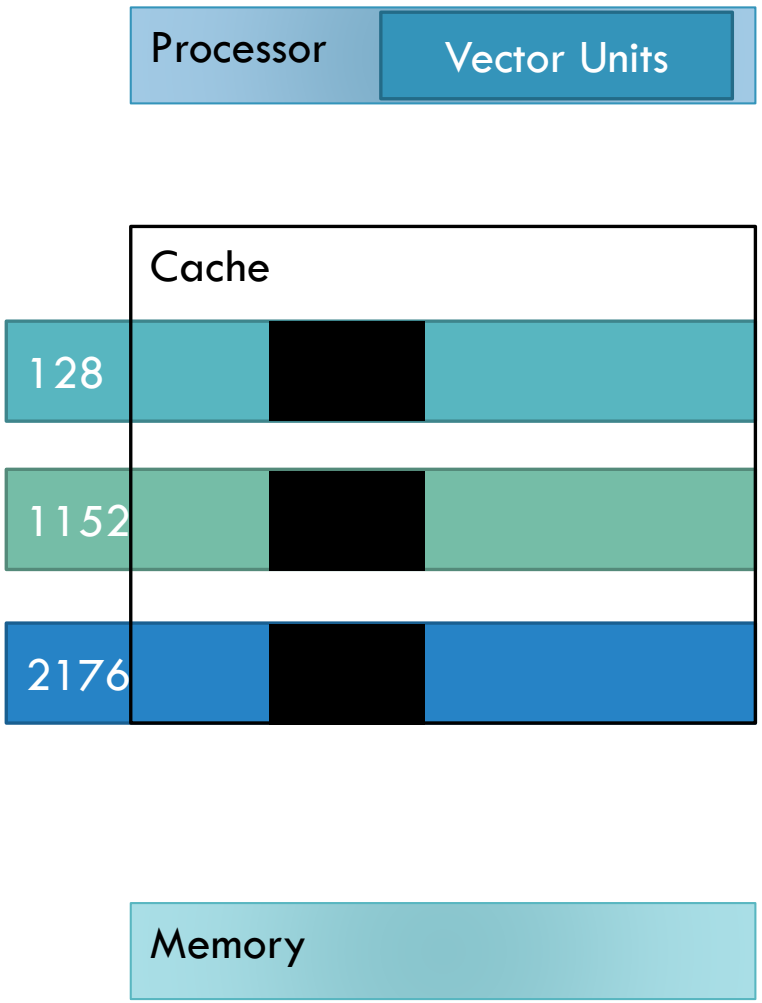


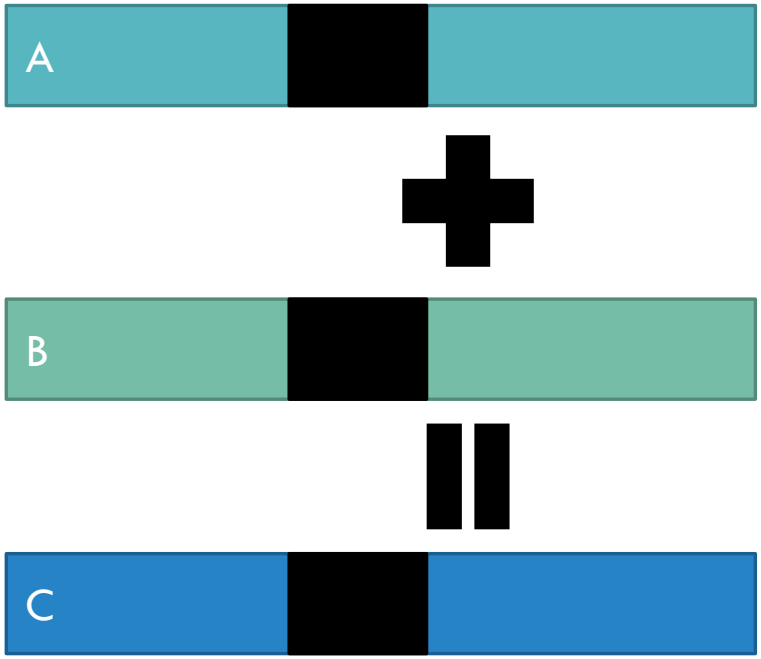
FUNCIONAMENTO – SOMA VETORIAL



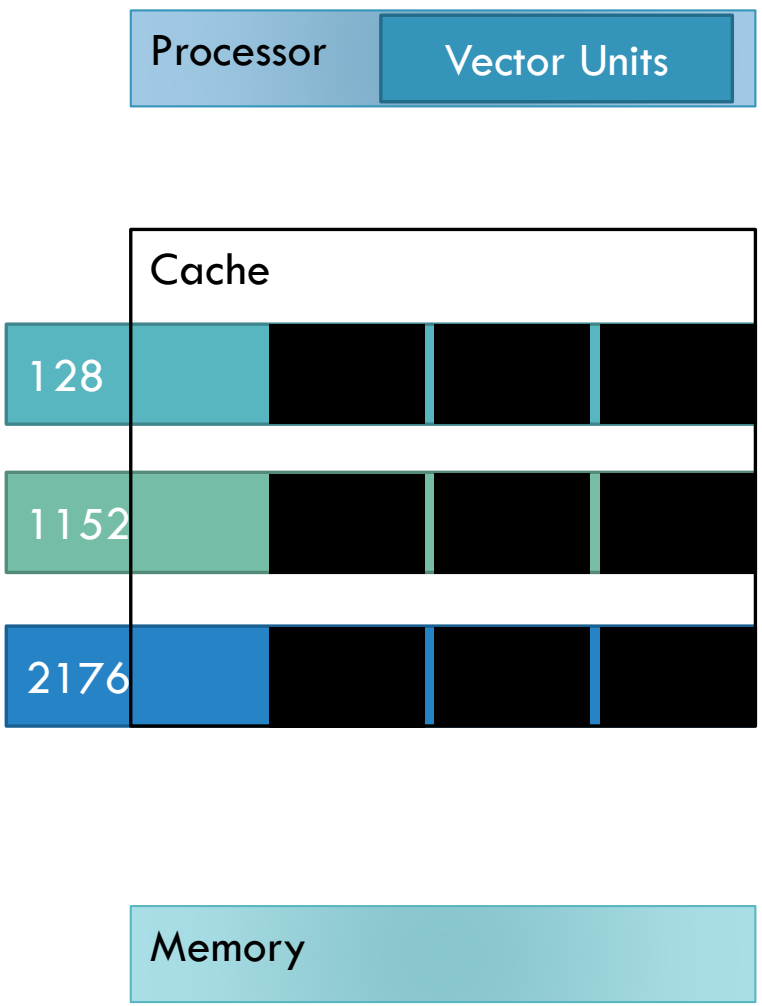


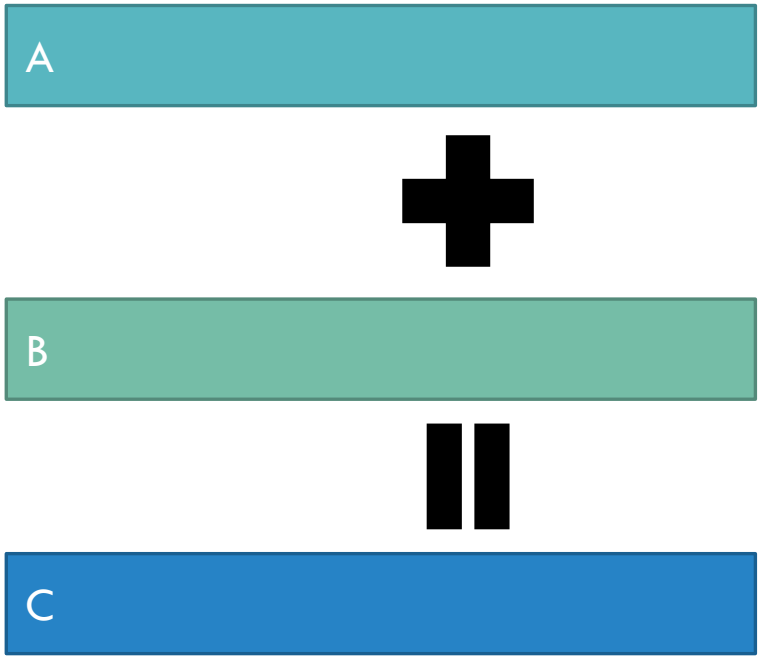
FUNCIONAMENTO – SOMA VETORIAL



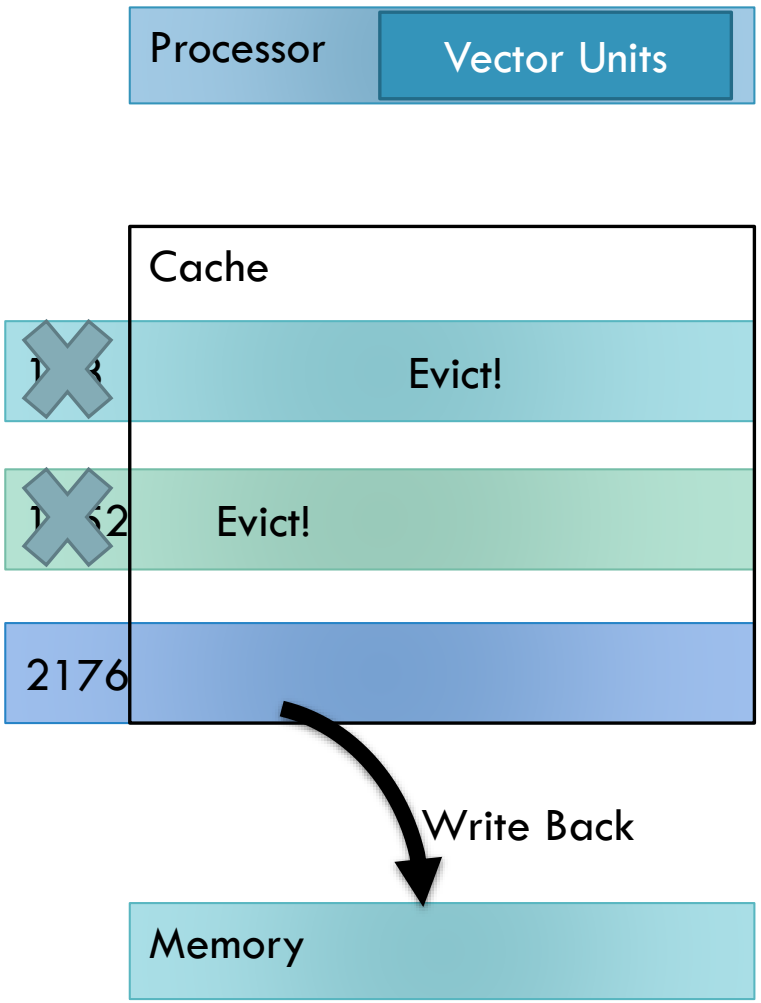


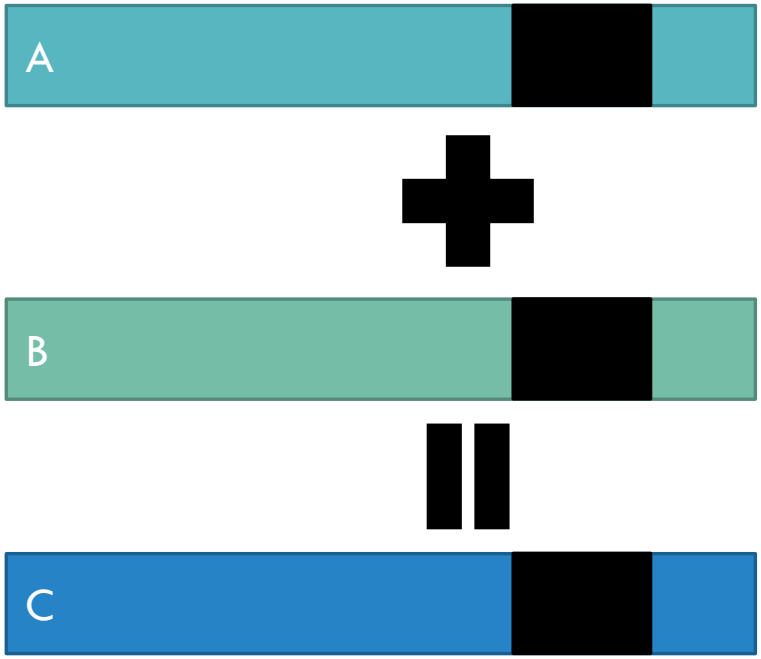
FUNCIONAMENTO – SOMA VETORIAL





FUNCIONAMENTO – SOMA VETORIAL





CONCLUSÕES / OBSERVAÇÕES

O programador pode otimizar o código para tirar desempenho da cache

- Como as estruturas de dados estão organizadas
- Como os dados são acessados
- Estrutura de laços aninhados
- Blocking é uma técnica geral

Todos sistemas favorecem “cache friendly code ”

- Obter o máximo desempenho requer conhecimento da plataforma especifica
  - Cache sizes, line sizes, associativities, etc.
- Podemos obter grande parte do desempenho com um código genérico
  - Mantendo o conjunto de dados pequeno (localidade temporal)
  - Usar pequenos pulos/strides (localidade especial)