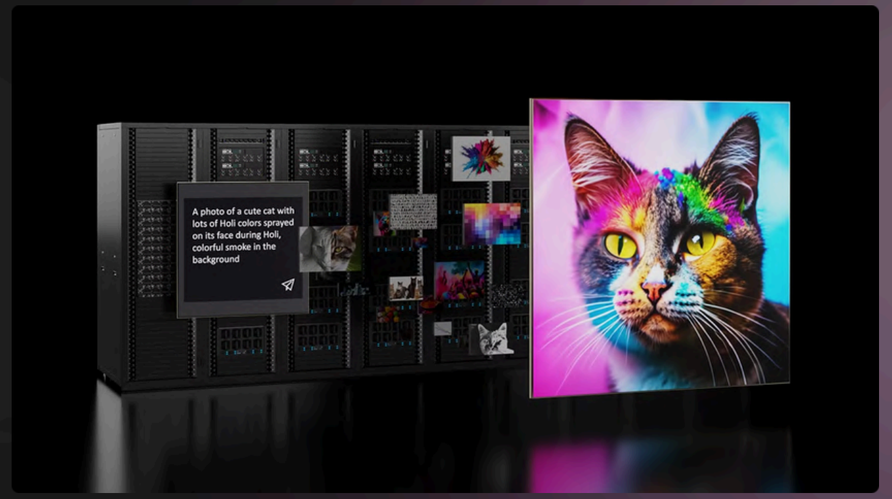
Utilização de GPUs para processamento de modelos de AI

“In 2009, I remember giving a talk at NIPS where I told about 1,000 researchers they should all buy GPUs because

GPUs are going to be the future of machine learning”

- Geoffrey Hinton

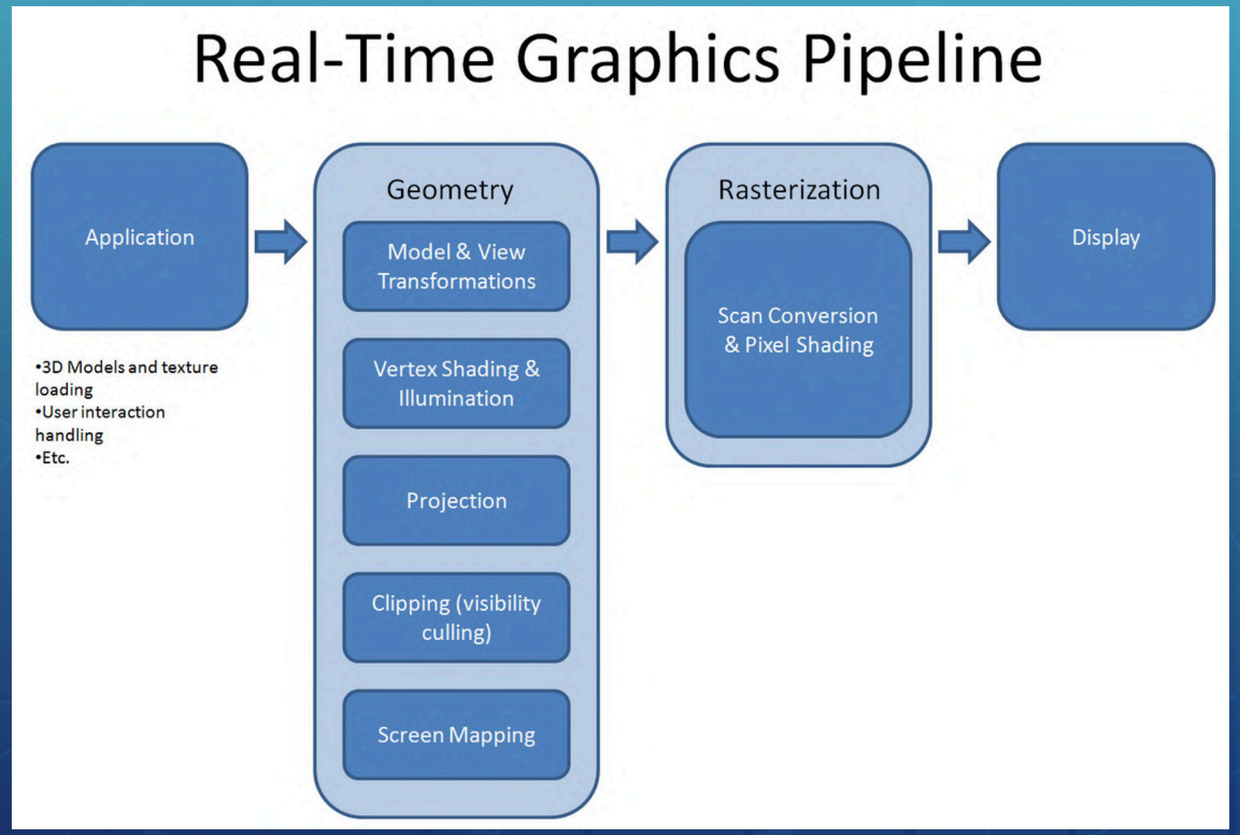


Grupo 03 | Turma A

Enzo Yasuo Hirano Harada Leonardo Ishida João Pedro Hamata

Victor Hugo Trigolo Amaral Miguel Bragante Henriques

Revolução 3D



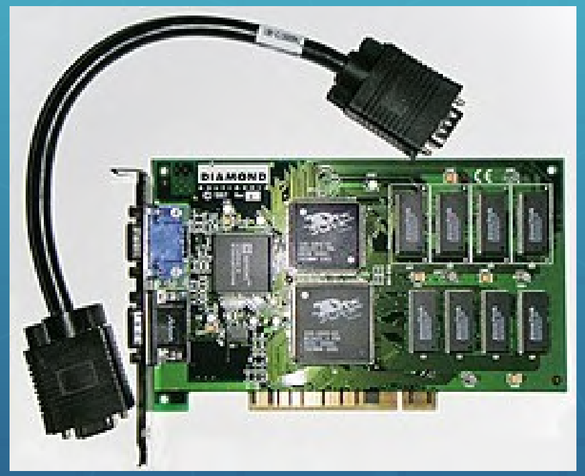
Na década de 90, com uma demanda de uma experiência mais imersiva na indústria dos jogos, gráficos 3D foram cada vez mais requisitados.

Quake (1996)

Wolfenstein 3D (1992)

Entretanto, os computadores de uso diário não possuíam hardware adequado ficando para trás dos consoles.
Devido a isso, vários jogos tinham softwares completamente renderizados apenas pela CPU, causando gráficos robustos e performances ruins.

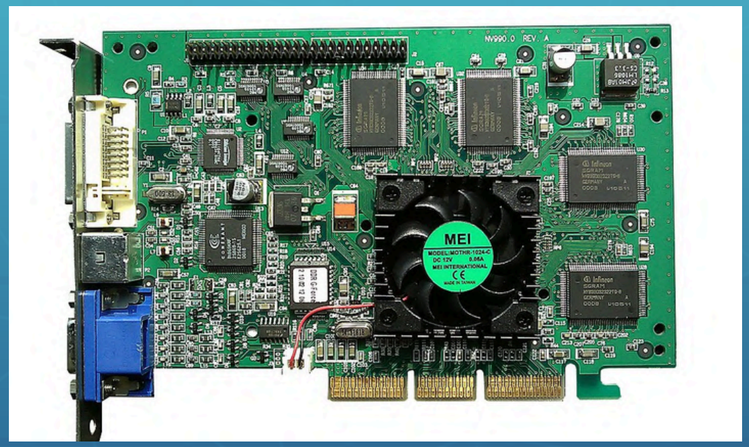
Revolução 3D: Destaques



Placa gráfica Voodoo da 3dfx é lançada, consistindo de uma placa utilizada apenas para renderizações 3D.

Cuidava das etapas finais do pipeline dos modelos 3D, cuidando da rasterização e processamento de pixels.

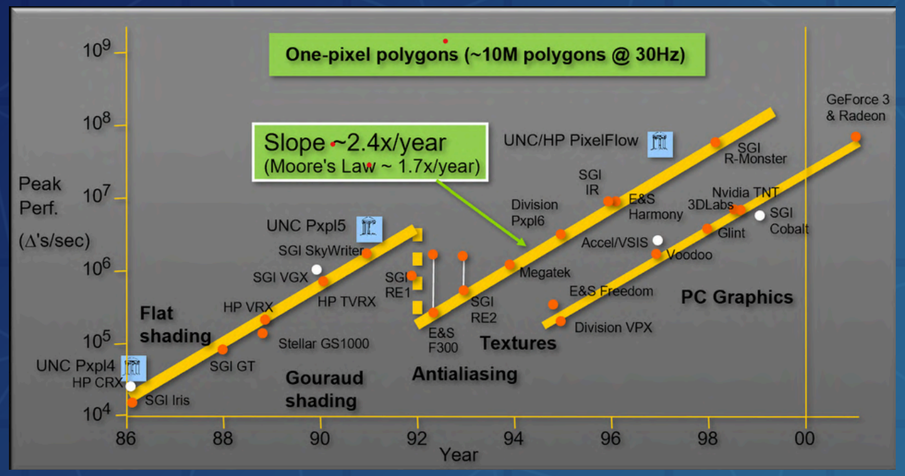
Chegou a representar 85% do mercado de aceleradores 3D



Destaque para as placas gráficas da NVIDIA, da série “GeForce”, mais especificamente a GeForce 256, considerada a primeira GPU moderna.

Ficava responsável por todas etapas da pipeline de gráficos 3D, deixando para a CPU apenas o trabalho de saber o que deve ser renderiza e aonde.

NVIDIA e a GPGPU

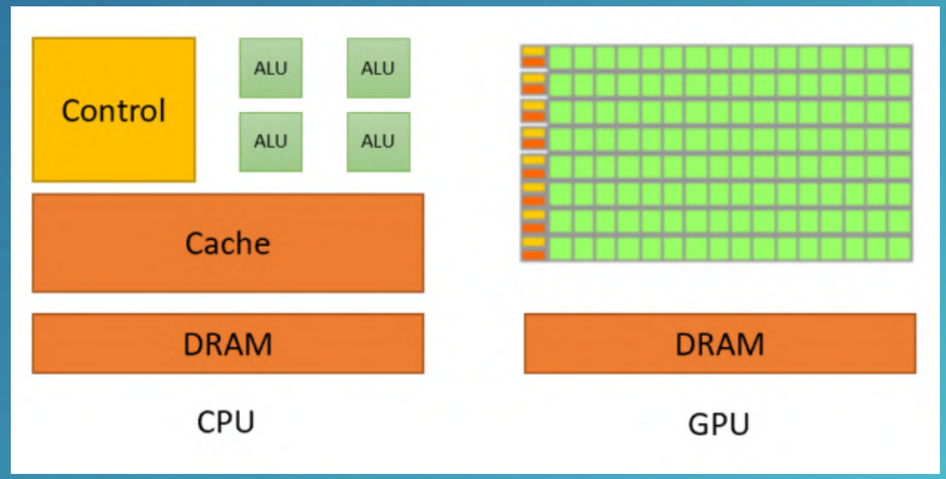


Hardware gráficos estavam desenvolvendo suas performances em mais de 2.4x/ano, mais do que previsto pela lei de Moore.

Incentivo a aproveitar a capacidade computacional da GPU em outras áreas

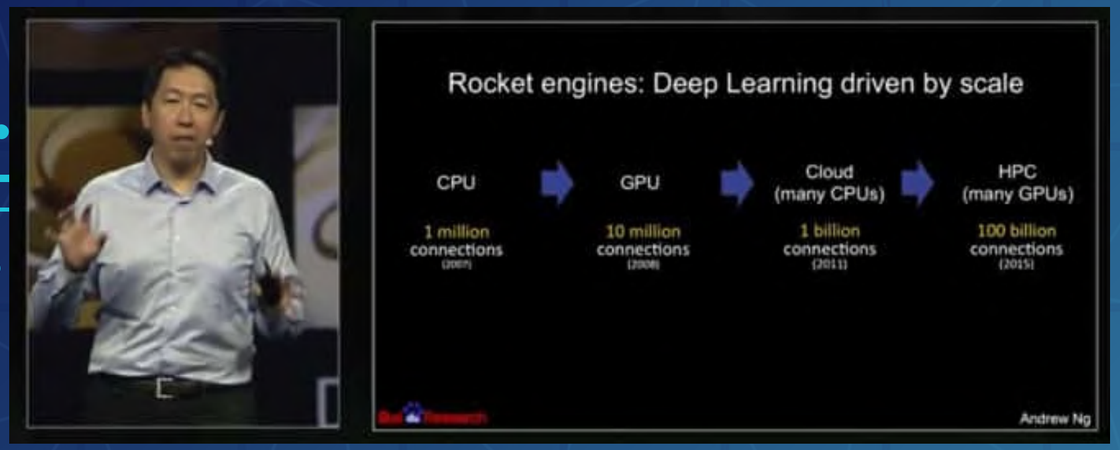
NVIDIA CUDA

Linguagem CUDA ajudou a simplificou o processo de desenvolvimento da programação paralela utilizando GPUs. Além disso, possui um modelo que facilta a escalabilidade de softwares que a utilizam



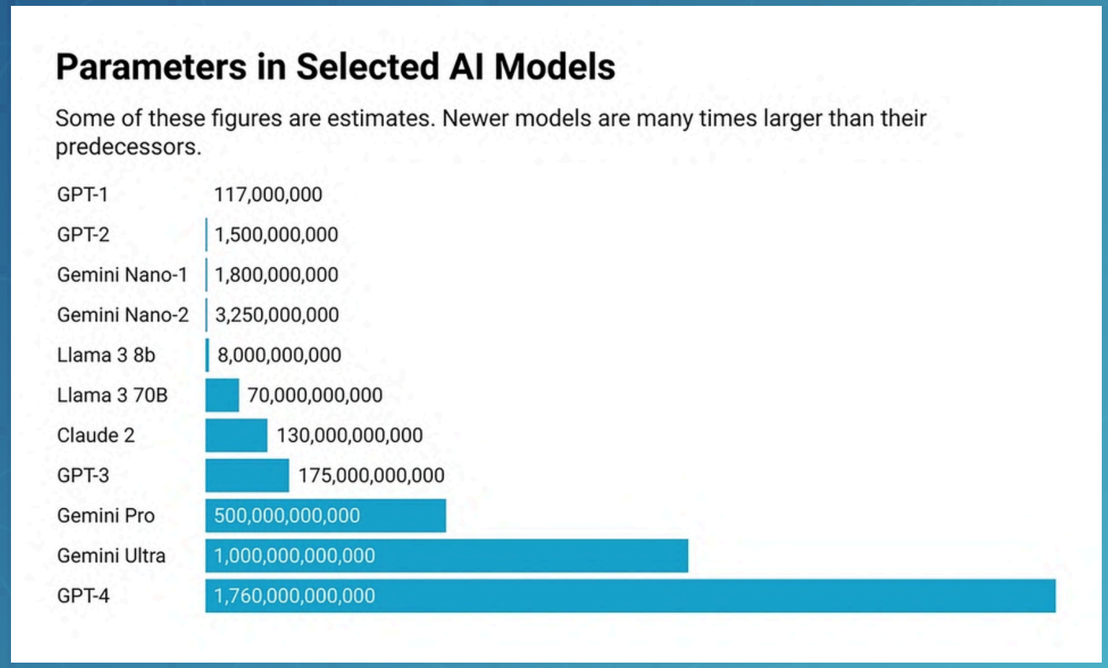
GPU é bem mais adequada para processamento de dados em grande escala, devido a sua paralelização e modelo SIMT.

GPU NA IA



Pesquisa feita organizada por Andrew NG, um dos principais nomes da área de IA, em Stanford de 2008, confirmou que a GPU em relação a CPU teve um speedup de 70x!

Operando com milhares de threads ativas e com baixo custo de escalonamento, foi possível processar um modelo de 100 milhões de parâmetros com a GPU em apenas um dia, se comparado a semanas com a CPU



A complexidade de modelos de IA tem crescido em cerca de 10x / ano, com o GPT-4 atingindo quase 1.8 trilhões de parâmetros.

GPU

Vimos que o processamento de modelos de IA pode ser caro e demorado, pois com o aumento de modelos, surgem limitações de memória e desempenho.

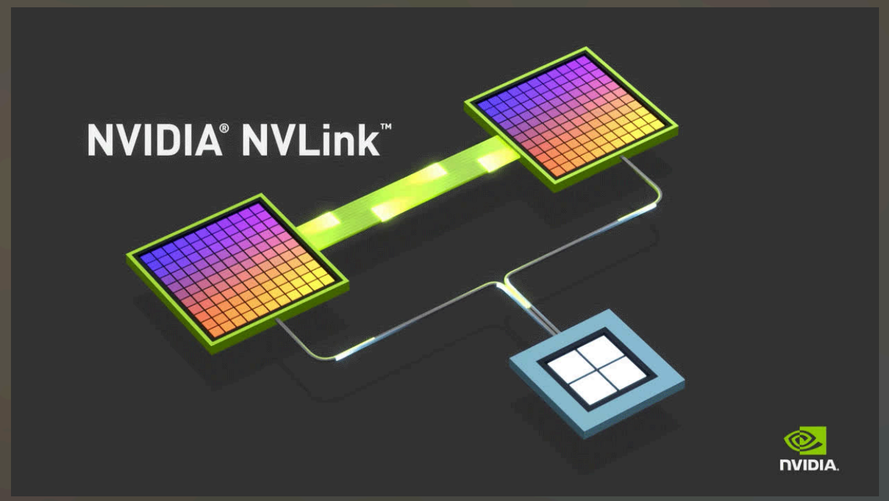
Como melhorar esse cenário?

Treinamento Distribuído

A ideia é utilizar, geralmente, **clusters** de GPUs interconectadas, distribuindo o processamento do modelo.

Nesse cenário, a **Cloud** fornece uma infraestrutura escalável e distribuída, para quem não pode investir em hardware.

Mas como funciona no hardware?



NVLINK

1. Tecnologia desenvolvida pela NVIDIA que fornece uma **comunicação** rápida entre GPUs, otimizada para a arquitetura CUDA.
2. Utilizada pela OpenAI, Meta AI e Google.
3. Resolve as limitações do barramento PCIe, com maior **largura de banda** e **menor latência**.

Técnicas

...

01

Paralelismo de dados:
Os dados são divididos entre as GPUs. Os gradientes são calculados localmente e depois sincronizados.

02

Parelelismo de modelo:
O modelo é dividido entre as GPUs, ao invés de copiado. Cada GPU processa, geralmente, parte de uma camada do modelo.

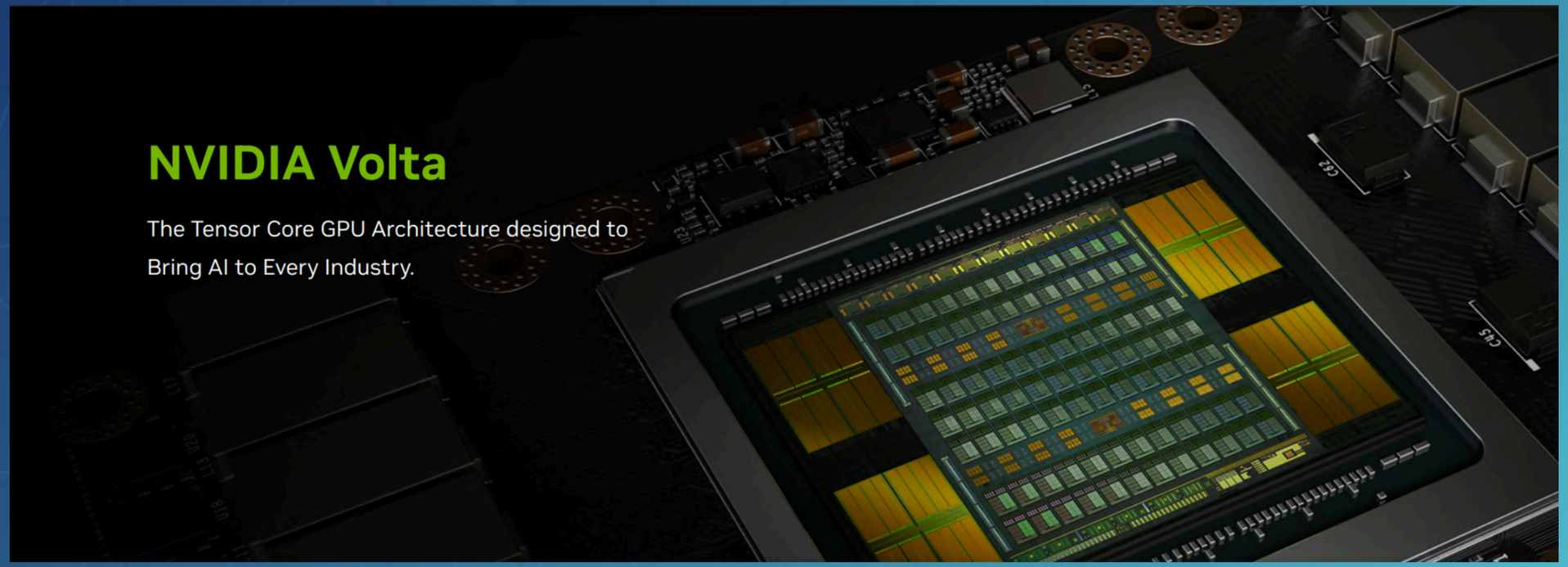
03

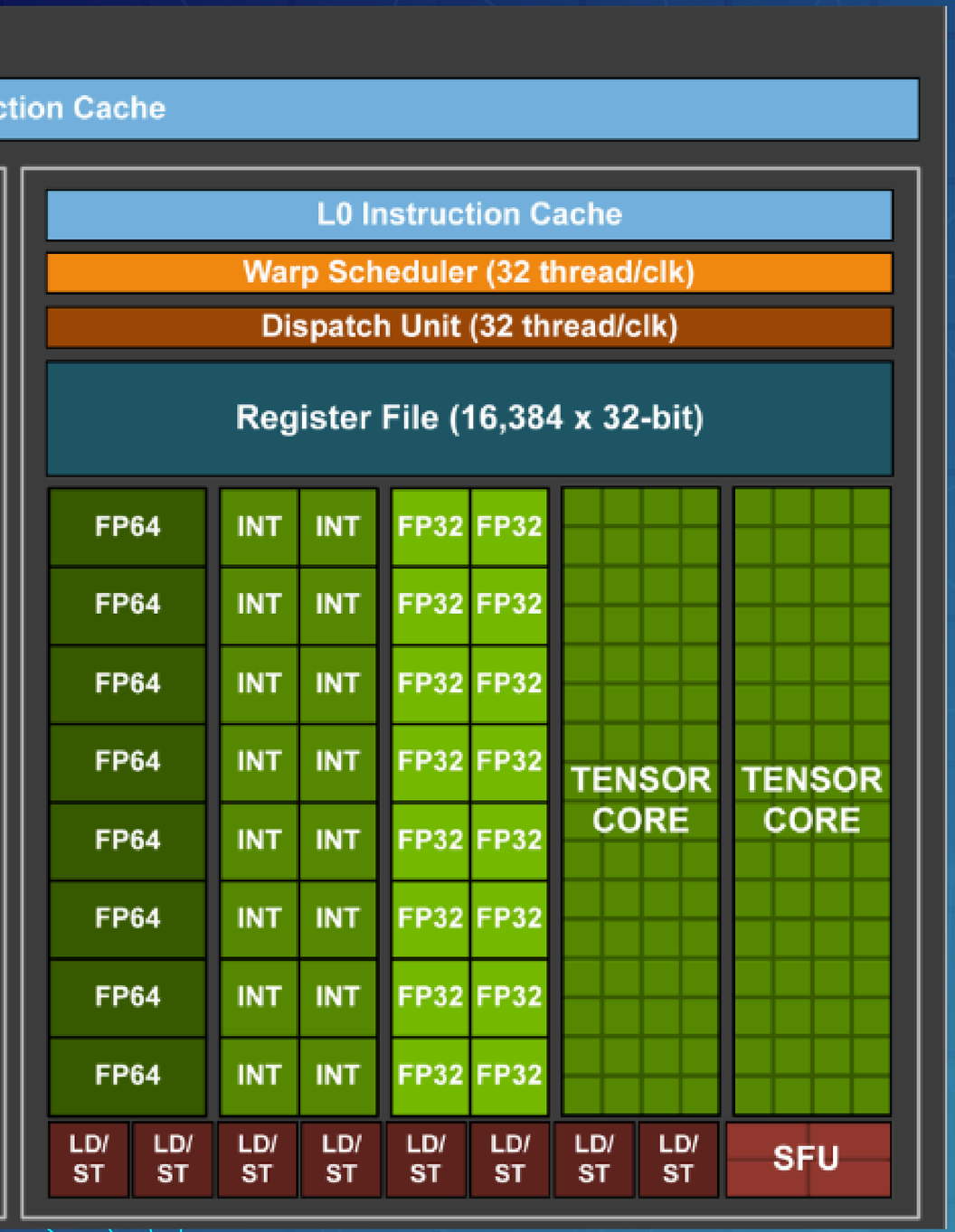
Paralelismo de pipeline:
O modelo é dividido entre estágios, os dados são processos sequencialmente como uma linha de montagem.

Tensor Core

Arquitetura volta

Arquitetura que revolucionou o treinamento de modelos de inteligência artificial, utilizando os novos processadores NVIDIA Tensor Core.
Cada GPU possui 640 Tensor Core.





Tensor core: o que são e como funcionam

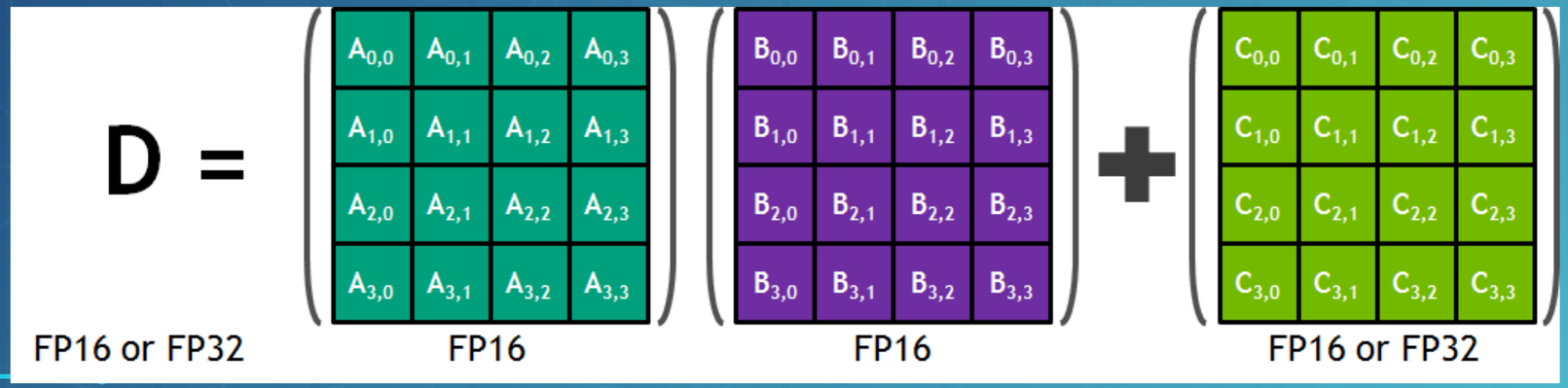
Tensor cores são unidades de hardwares especializados que computam rapidamente operações de matrizes utilizando precisão mista, a fim de obter aceleração no treinamento e inferênciade modelos de inteligência artifical e high performance computing (HPC).

Multiplicação de matrizes e acumulação (FMA)

Cada Tensor Core utiliza dados de uma matriz 4x4, e realiza a seguinte operação aritmética:

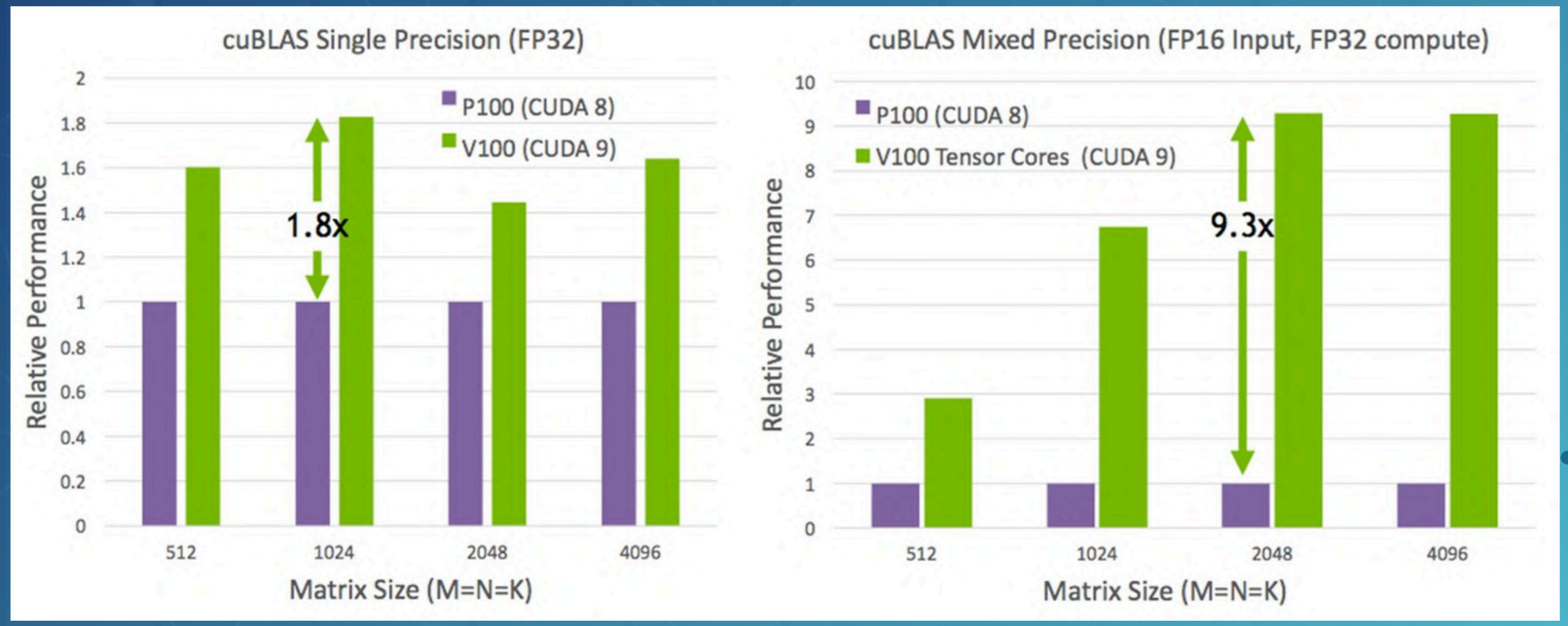
$$D = A \times B + C$$

Onde A, B, C e D são matrizes 4x4. A multiplicação de matrizes utiliza dados dos inputs A e B, que são matrizes FP16, e a acumulação de matrizes C e D podem ser do tipo FP16 ou FP32, depende da precisão desejada.
Esse tipo de operação faz apenas um arredondamento, para obter resultado mais preciso.



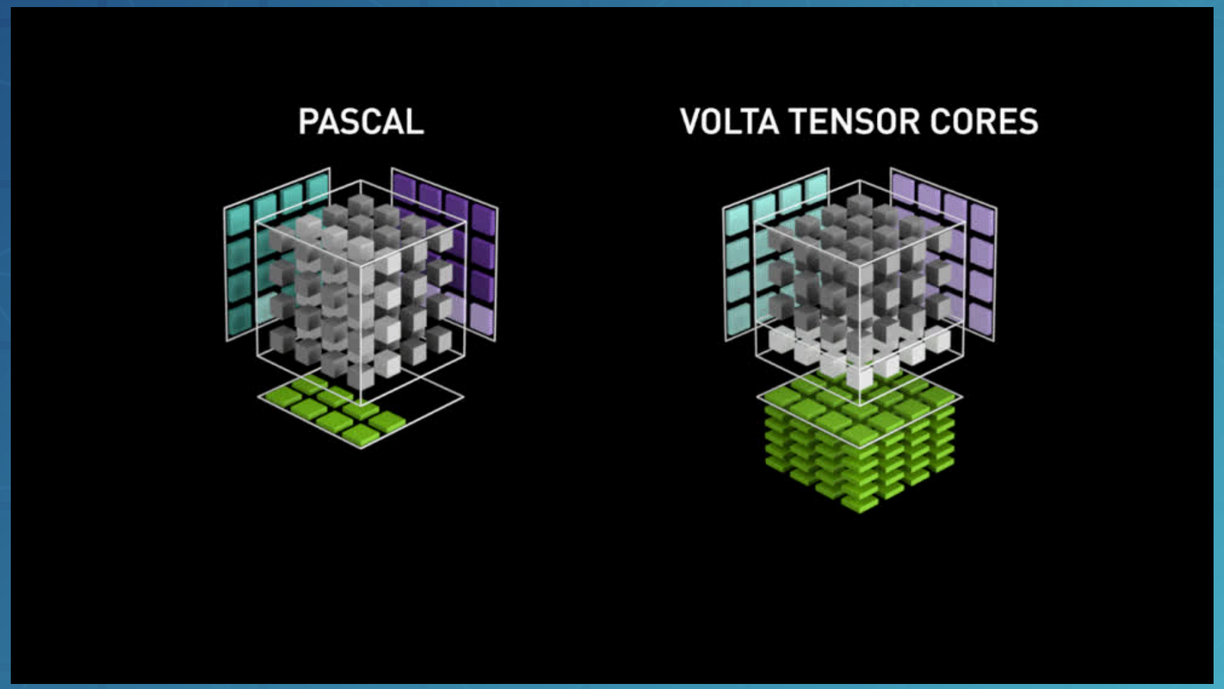
Comparação: Cuda core x Tensor core

Tensor cores conseguem fazer 64 operações Fused Multiply-Add (FMA) por ciclo de clock, o que resulta em uma matriz 4x4. Enquanto isso, CUDA cores conseguem fazer apenas uma operação aritmética por ciclo de clock.



Comparação: Cuda core x Tensor core

Ao utilizar Tensor cores para realizar as operações FMA, o potencial de vazão de dados (throughput) dessas GPUs pode ser aumentada em até 12x, em termos de teraFLOPs (floating point operations).



TPU’s e NPU’s

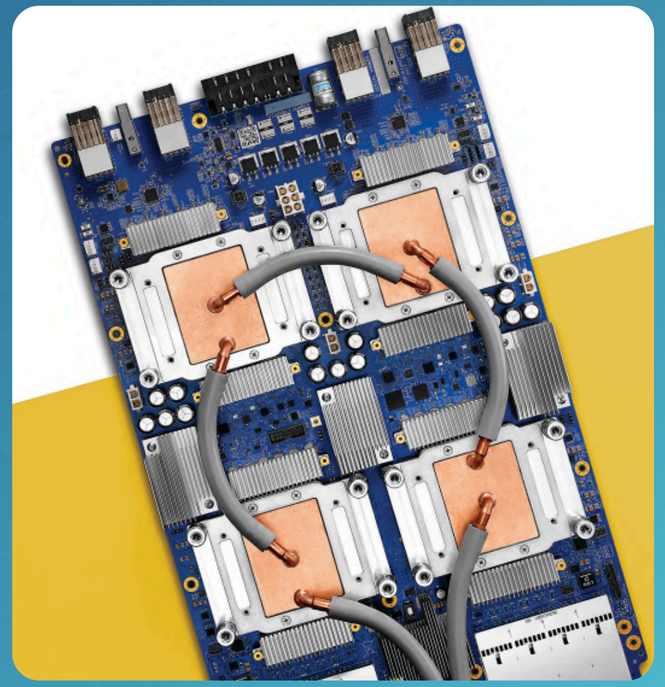
TPU

Tensor Processing Unit

Unidade de processamento paralelo baseado em máquinas SIMD/**SIMT** (como a GPU).

Especializada em processamento de tensores. Utilizada no treinamento e inferência de modelos de **redes neurais**.

Como ela é mais eficiente do que a GPU em processamento de redes neurais?



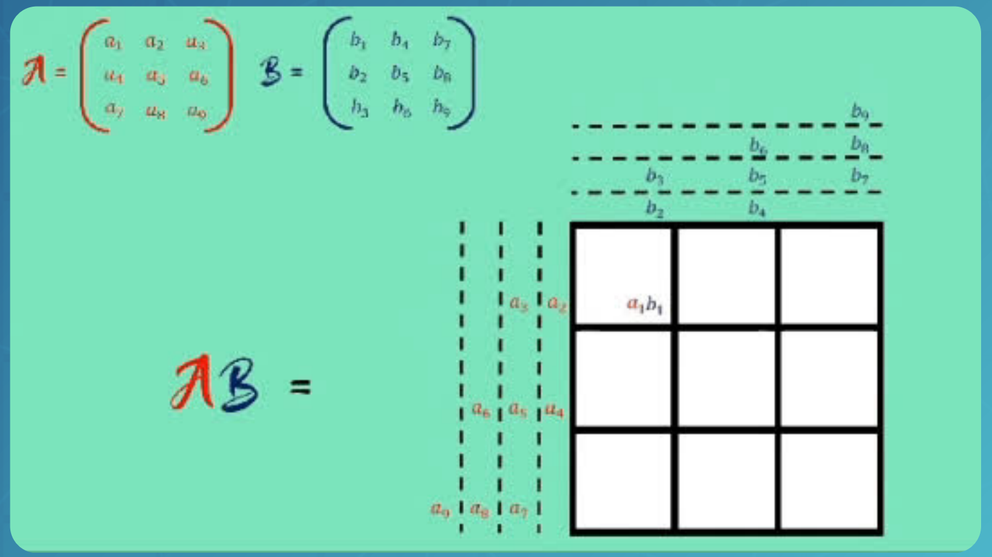
Google TPU 3.0

TPU

Tensor Processing Unit

Otimização das operações sobre tensores (multiplicação de matrizes e convolução):

- **Systolic arrays**: matriz de unidades de processamento com transferência de resultados entre células adjacentes sem acessar memória global.



TPU

Tensor Processing Unit

Precisão das operações tendem a ser menores do que as da GPU (podemos ter precisão de apenas 8 bits), o que diminui o consumo de energia e latência.

Buffer on-chip: armazenamento de pesos e ativações eficientemente, sem o uso da memória.

NPU

Neural Processing Unit

Unidade de processamento paralelo também baseado em máquinas SIMD/**SIMT.** Porém seu uso é direcionado a modelos de IA em dispositivos mobile, IoT e veículos autônomos.

O que a torna adaptada a modelos nesses dispositivos?

Samsung Exynos 9 processador com NPU integrada

NPU

Neural Processing Unit

Possui uma grande **eficiência energética** e **baixa latência**:

Foca apenas na inferência de modelos de Machine Learning. Diferente da TPU, que é utilizada também para o treinamento de grandes modelos.

Uso de systolic arrays

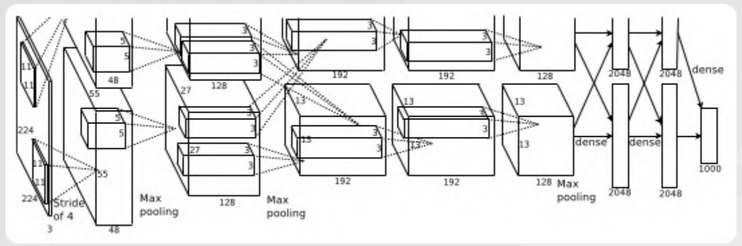
Fusão de operações em uma única instrução.

Precisão dinâmica, podendo ser ainda menor que 8 bits, reduzindo uso energético.

NPU

Neural Processing Unit

O desempenho de algumas NPU’s chega a ser **100 vezes melhor** do que o de uma GPU comparável, de equivalente consumo de energia.



AlexNet

Contexto histórico:

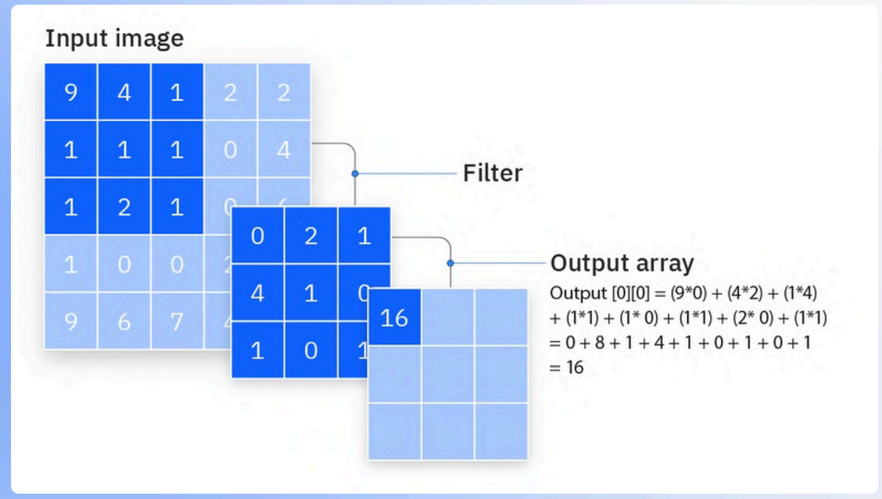
- Participação no desafio ILSVRC-2012.
- Classificação de 1,2 milhão de imagens em 1.000 categorias.

Detalhes técnicos:

- Divisão da rede entre duas GPUs GTX 580 devido à limitação de memória (3GB por GPU).
- Uso de comunicação otimizada entre GPUs em camadas específicas.

Impactos:

- Redução do tempo de treinamento em comparação com CPUs.
- Popularizou as GPUs para tarefas de IA.



CNNs

Paralelização de operações convolucionais:

- Transformação de convoluções em multiplicações matriciais (GEMM).
- GPUs processam múltiplos patches e canais simultaneamente.

Avanços proporcionados pelas GPUs:

- Implementação eficiente de **ReLUs**:
  - Computacionalmente leves.
  - Paralelizáveis e fundidas com operações anteriores.
- Pooling e normalização:
  - Otimizações específicas para reduzir transferências de memória.

Transformers

Cálculos de atenção requerem alto paralelismo e memória.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Em GPU, esta operação é altamente paralelizável:

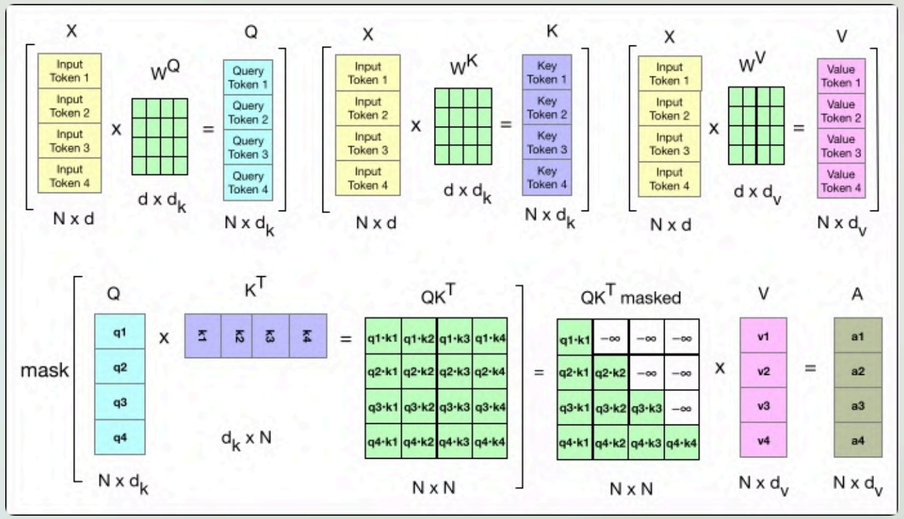
- As multiplicações QK^T são computadas simultaneamente
- O processamento ocorre em paralelo para todas as cabeças de atenção
- O softmax é implementado com otimizações para GPU

1

2

A arquitetura da GPU permite:

- Processamento simultâneo de tokens
- Paralelização por batch e por cabeça de atenção
- Otimização de memória para sequências longas



Obrigado!