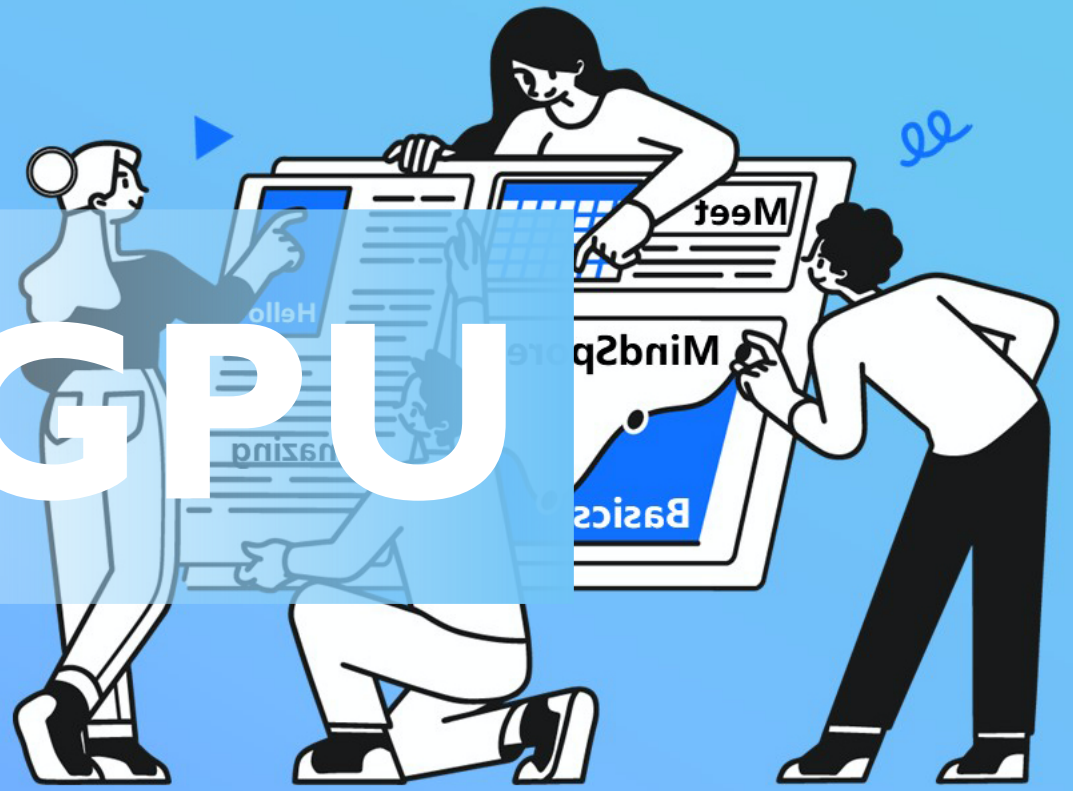
AI 芯片 – AI 芯片基础

图形处理器GPU



ZOMI

Talk Overview

1. AI 计算体系

- 深度学习计算模式
- 计算体系与矩阵运算

2. AI 芯片基础

- 通用处理器 CPU
- 从数据看 CPU 计算
- 通用图形处理器 GPU
- AI专用处理器 NPU/TPU
- 计算体系架构的黄金10年

Talk Overview

1. AI 计算体系
   - 深度学习计算模式
   - 计算体系与矩阵运算
2. AI 芯片基础
   - 通用处理器 CPU
   - 从数据看 CPU 计算
   - 通用图形处理器 GPU
   - AI专用处理器 NPU/TPU
   - 计算体系架构的黄金10年

1. 硬件基础
   - GPU 工作原理
   - GPU AI编程本质
2. 英伟达 GPU 架构
   - 从 Fermi 到 Hopper 架构
   - Tensor Code 和 NVLink 详解
3. GPU 图形处理流水线
   - 图形流水线基础
   - GPU 逻辑模块划分
   - 图形处理算法到硬件

Talk Overview

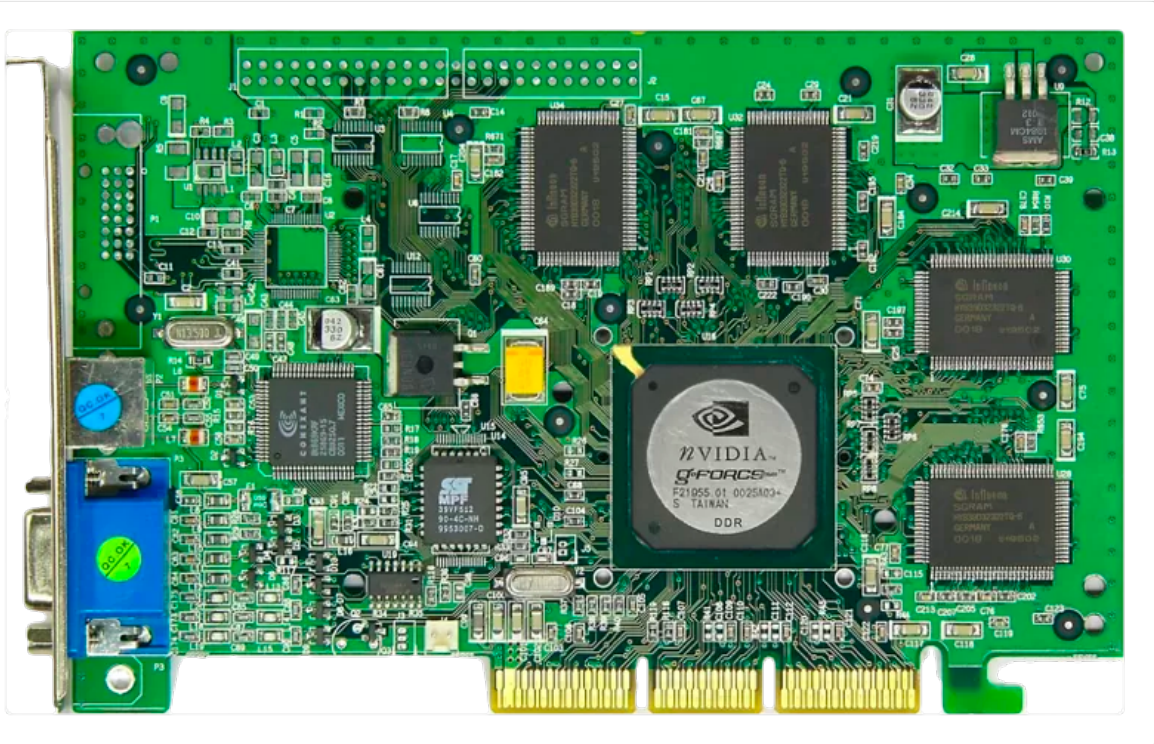
1. 图形处理器 GPU

- The History – GPU 发展历史和组成
- Difference GPU/CPU – GPU 和 CPU 的区别
- Why AI Need GPU – AI发展需要GPU
- Application – CPU的应用场景

GPU 发展历史

GPU 发展阶段 (I)

- 第一代GPU（ Before 1999 ）：部分功能从CPU分离， 实现针对图形图形处理的硬件加速。以几何处理引擎 GEOMETRY ENGINE 为代表，只能起到 3D 图像处理的加速作用，不具有软件编程特性。



GPU 发展阶段 (II)

- 第二代 GPU（ 1999-2005 ）， 实现进一步的硬件加速和有限的编程性。

- 1999年，英伟达发布了专为执行复杂的数学和几何计算的 GeForce256 图像处理芯片，将更多的晶体管用作执行单元， 而不是像 CPU 那样用作复杂的控制单元和缓存，将(TRANSFORM AND LIGHTING) 等功能从 CPU 分离出来，实现了图形快速变换，这成为 GPU 真正出现的标志。
- 2000-2005年， GPU 技术快速发展，运算速度迅速超过 CPU。2001年英伟达和ATI 分别推出 GEFORCE3 和 RADEON 8500，图形硬件的流水线被定义为流处理器，出现了顶点级可编程性，同时像素级也具有有限的编程性。但 GPU 的整体编程性仍然比较有限。

GPU 发展阶段 (II)

- 第二代 GPU（ 1999-2005 ）， 实现进一步的硬件加速和有限的编程性。

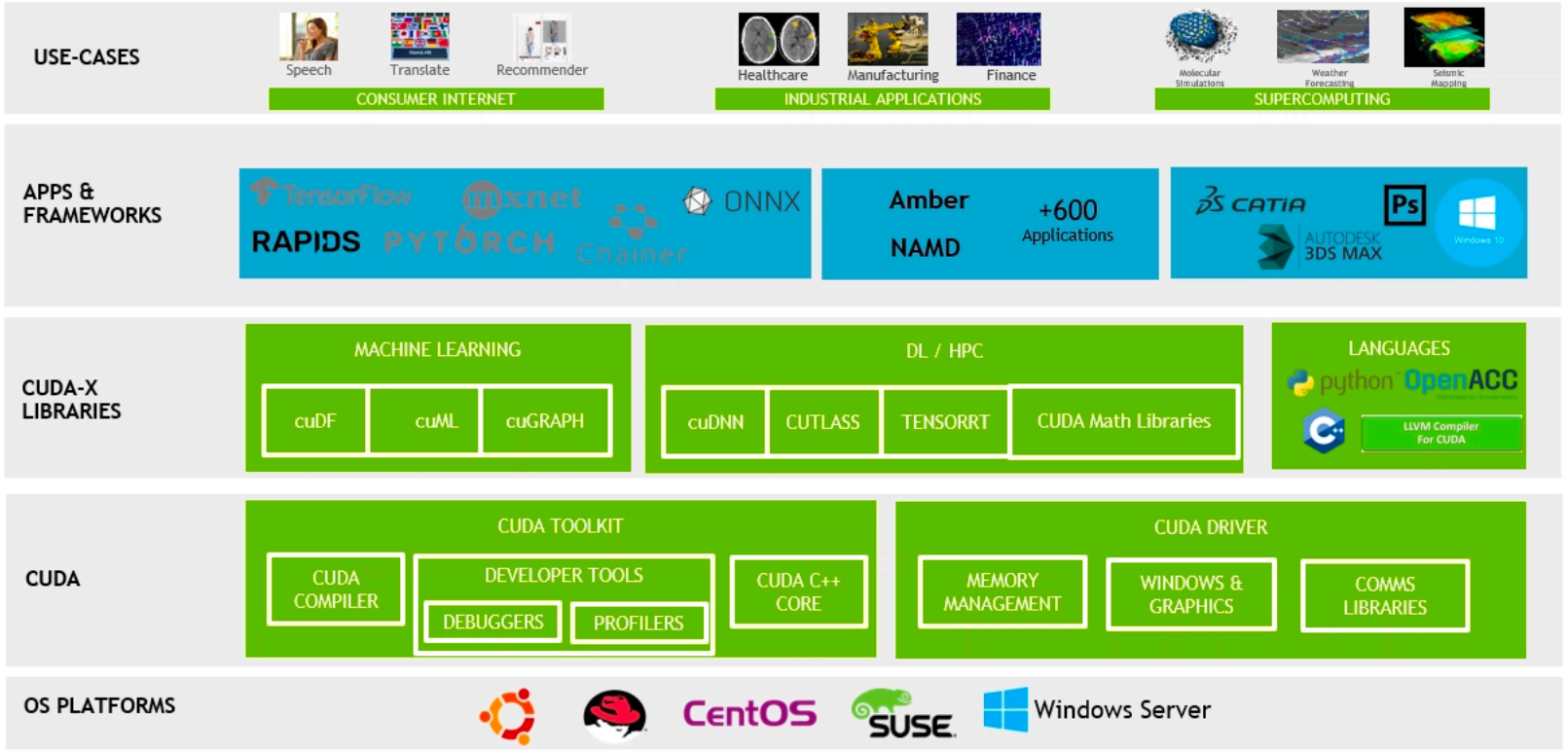
GPU 发展阶段 (III)

- 第三代 GPU（2006~），实现方便的编程环境创建， 可以直接编写程序。

- 2006 年，英伟达与 ATI 分别推出了CUDA（ Compute United Device Architecture ）和CTM（ CLOSE TO THE METAL ）编程环境， 使得 GPU 打破图形语言的局限成为真正的并行数据处理超级加速器。
- 2008 年，苹果公司提出一个通用的并行计算编程平台 OPENCL（开放运算语言），与 CUDA 绑定在英伟达的显卡上不同，OPENCL 和具体的计算设备无关，并迅速成为移动端GPU的编程环境业界标准。

GPU 发展阶段 (III)

- 第三代 GPU（2006~），实现方便的编程环境创建， 可以直接编写程序。



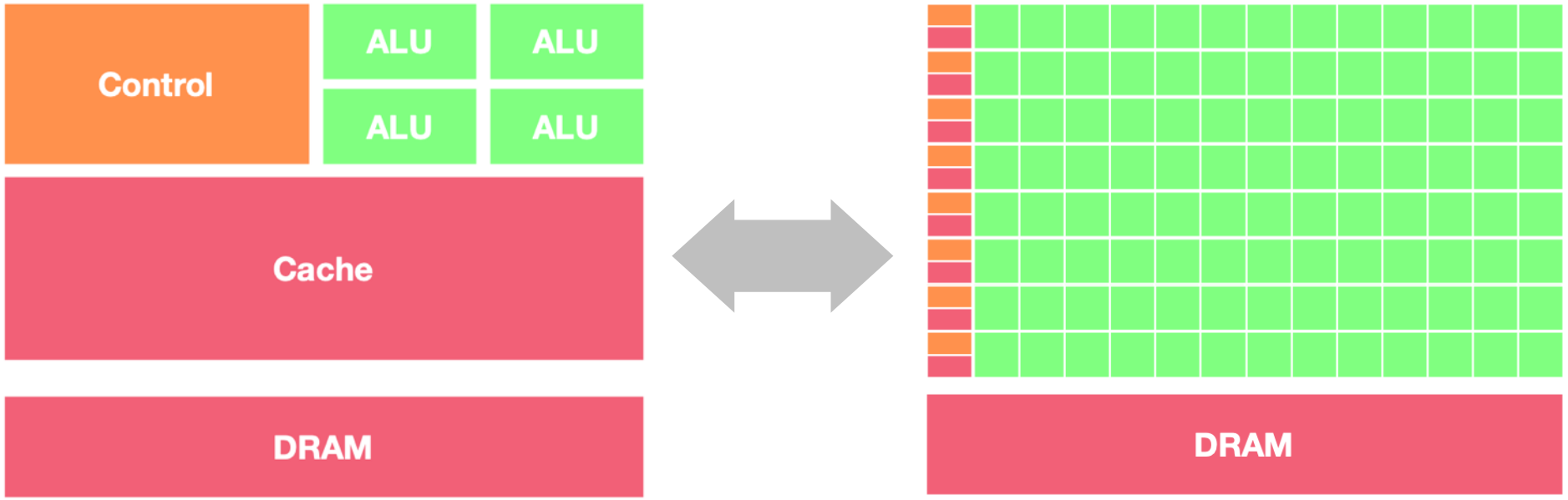
GPU vs CPU

图形计算单元

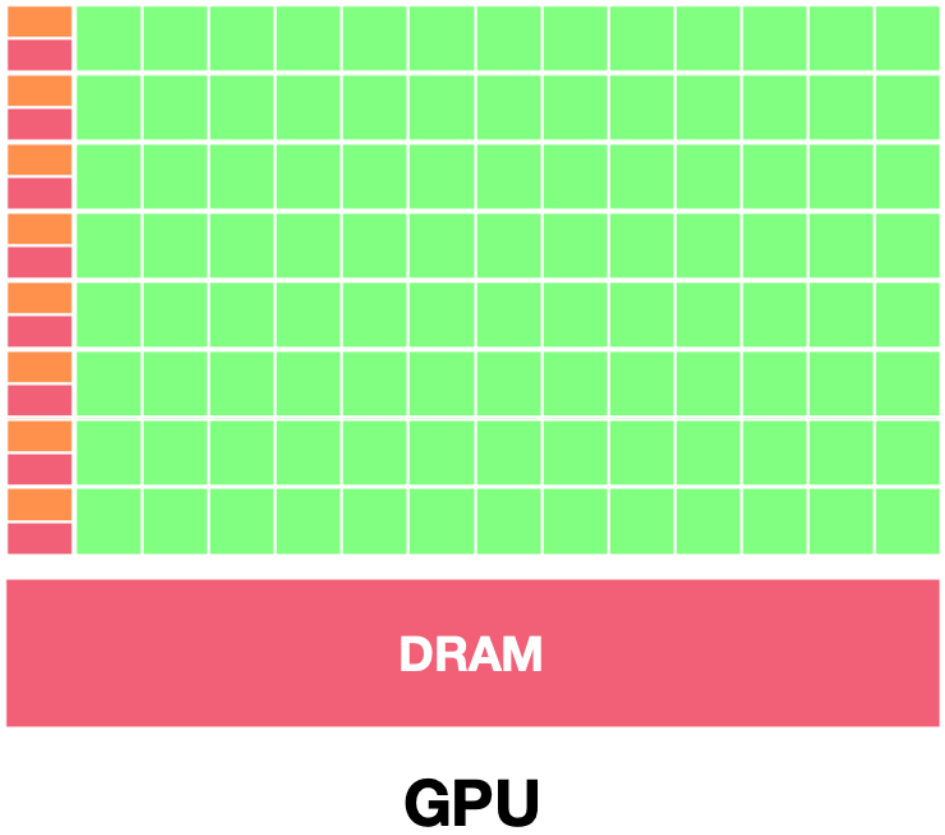
- 正如其全称“图形计算单元”，GPU的初衷主要是为了接替CPU进行图形渲染的工作。因为图像上的每一个像素点都需要处理，这项任务计算量相当大。尤其遇上一个复杂的三维场景，就需要在一秒内处理几千万个三角形顶点和光栅化几十亿的像素。不过，由于每个像素点处理的过程和方式相差无几，这项艰巨的任务可以靠并行计算来化解。

GPU vs CPU

- GPU几乎主要由计算单元ALU组成，仅有少量的控制单元和存储单元。GPU采用了数量众多的计算单元和超长的流水线，但只有非常简单的控制逻辑并省去了Cache。
- CPU不仅被Cache占据了大量空间，而且还有有复杂的控制逻辑和诸多优化电路，相比之下计算能力只是CPU很小的一部。



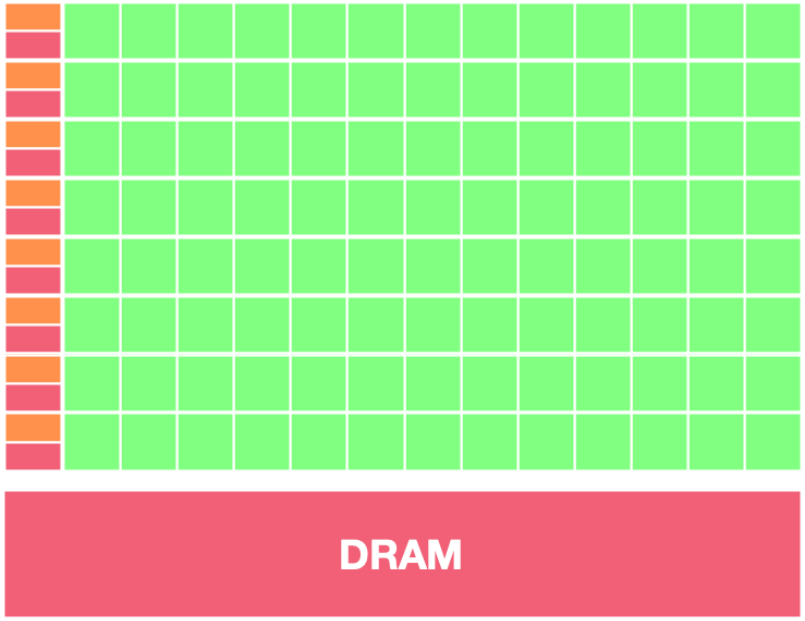
GPU Architecture



- **Small Caches**
  - boost memory throughput
- **Simple Control**
  - no branch prediction
  - no data forwarding
- **Energy Efficient ALUs**
  - Long latency but high throughput
- **Massive threads tolerate latencies**

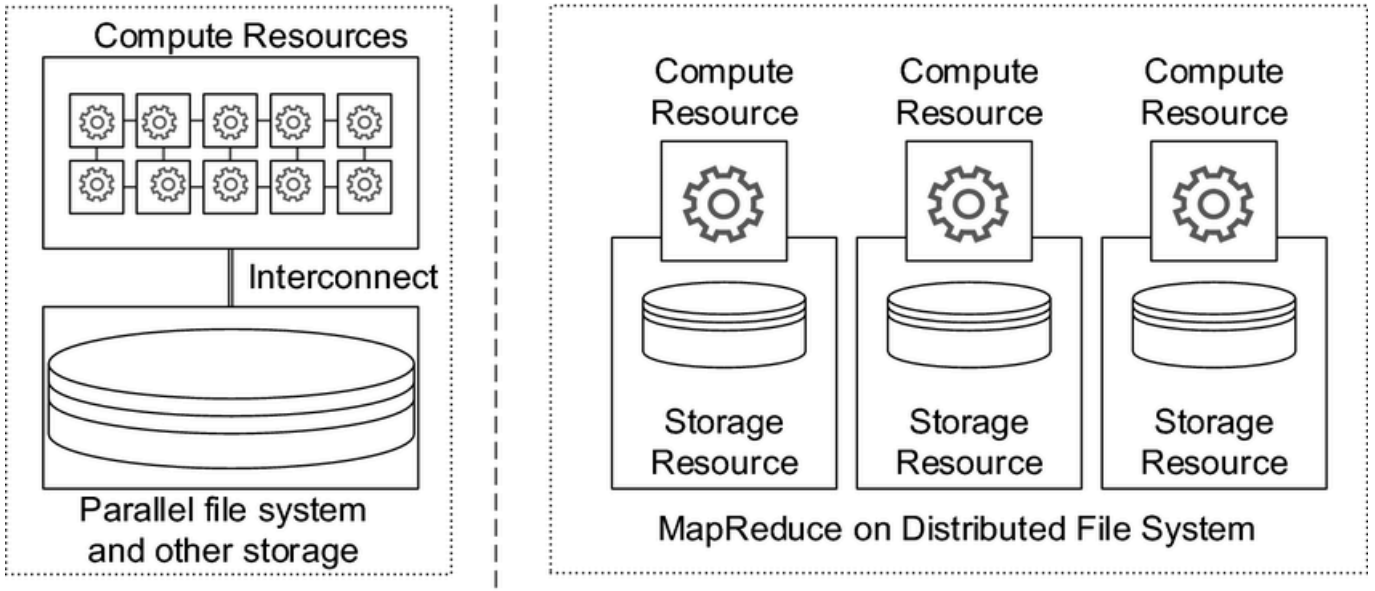
GPU Architecture

- **GPU Cache**：GPU 特点是有很多 ALU 很少 Cache，缓存目的不是保存后面需要访问的数据，与CPU不同，而是为 Threads 提供服务。
- **Reason**：如果有很多 Threads 线程需要访问同一段数据，缓存会合并这些访问，然后再去访问 DRAM，获取数据后 Cache 会统一转发该数据到对应 Threads 线程。



GPU 适合处理的程序

1. **计算密集型程序**：所谓计算密集型 (Compute-intensive) 程序，大部分运行时间消耗在寄存器运算上，寄存器的速度和处理器的速度相当，从寄存器读写数据几乎没有延时。
2. **易于并行程序**：GPU 虽然叫 SIMT，其实为特殊的 SIMD(Single Instruction Multiple Data) 架构，拥有成百上千个核 CUDA Core，每一个核在同一时间能执行同样指令。



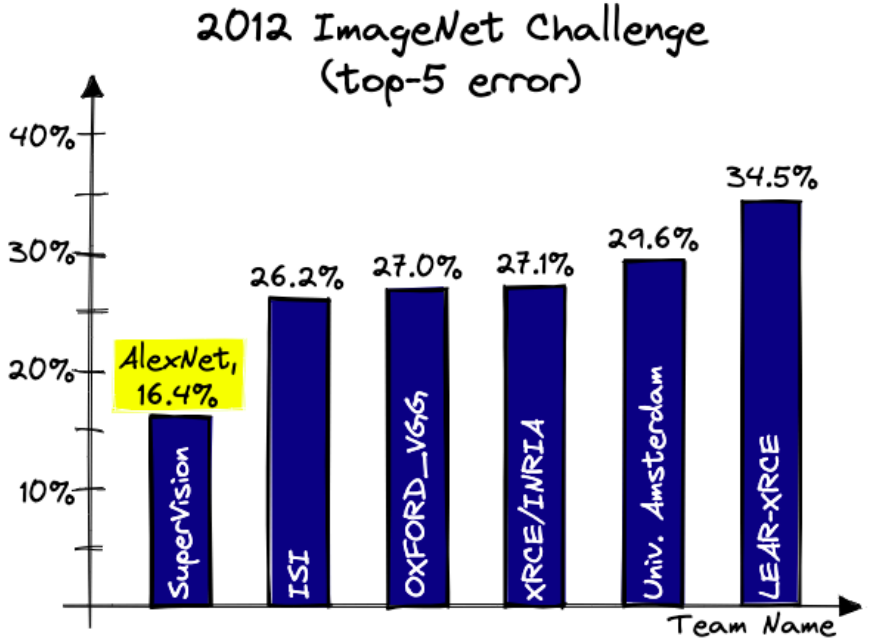
(a). Compute-intensive processing paradigm

(b). Data-intensive processing paradigm

AI的规模化发展
急需GPU

AI 遇到 GPU

- 2012年，Hinton 和 Alex Krizhevsky 设计了 AlexNet，使用了两块英伟达 GTX 580 训练了两周 AlexNet，将计算机图像识别的正确率提升了一个数量级，并获得了 2012 年 ImageNet 竞赛冠军，充分展示了GPU在AI计算中的巨大潜力。



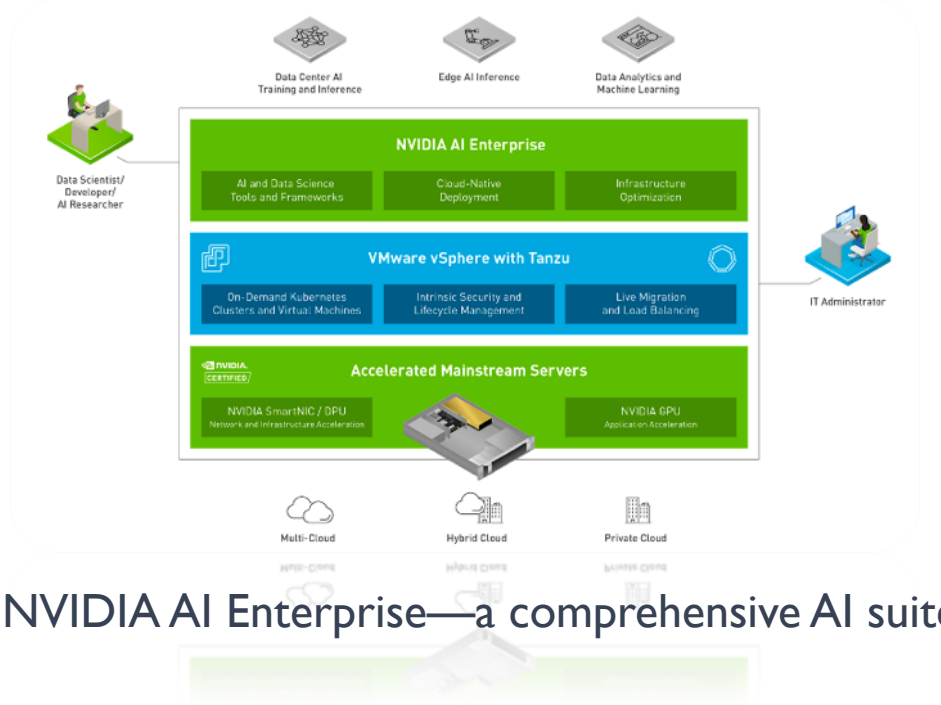
AI 遇到 GPU

- 谷歌使用 1000 台 CPU 服务器，基于Google YouTube 视频同的数据完成了猫狗识别的任务，而 2012 年吴恩达等采用 3 台 GTX680-GPU 服务器完成了同样的任务。
- 毋庸置疑，AlexNet 和吴恩达等工作在业界和学界都产生了良好的示范效应。或许从这段时间开始，学术界关于AI相关的研究逐渐更多的采用了GPU，互联网头部厂商也陆续开始引入GPU到各自的生产研发环境。

AI 爆发期

- 从2005/2006年开始有人尝试用GPU进行AI计算，到2012/2013年GPU被更大范围的接受，以及到2016/2017年GPU称为AI计算的标配，具有一定的偶然性，发现深度学习网络层次越深、网络规模越大，GPU的加速效果越显著。

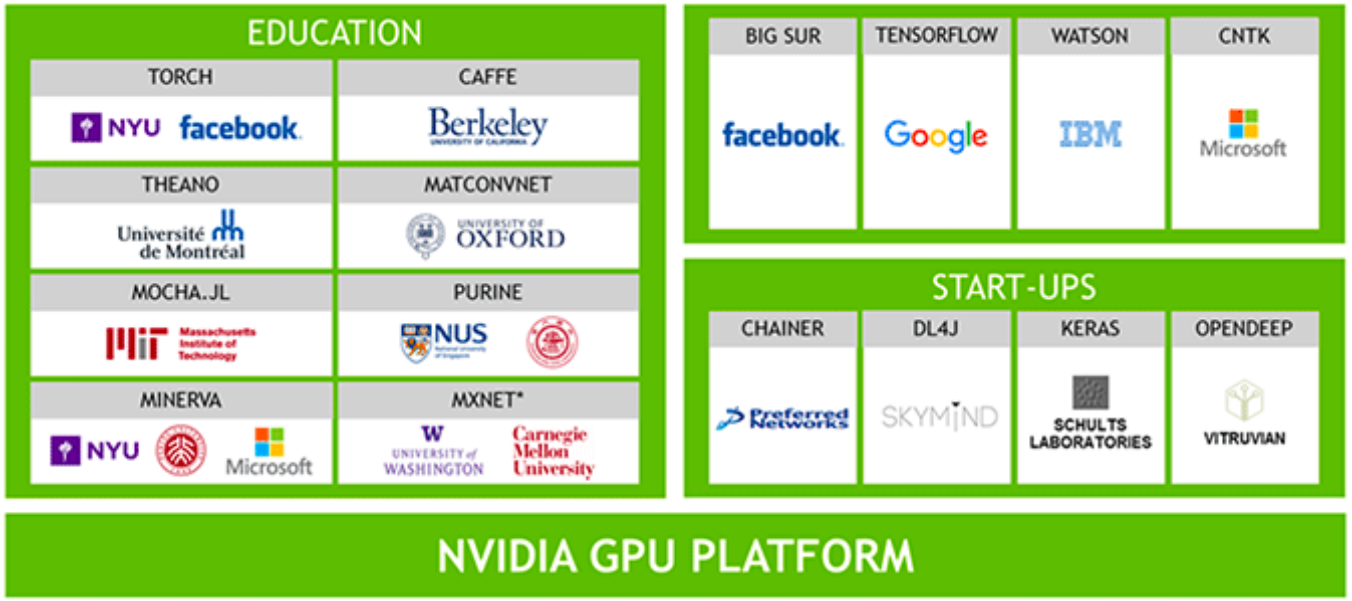
Top AI Frameworks and Tools



NVIDIA AI Enterprise—a comprehensive AI suite

Why GPU ?

- 深度学习每个计算任务都独立于其他计算，任何计算都不依赖于其他计算结果，可以采用高度并行的方式进行计算。
- GPU 相比 CPU 拥有更多独立大吞吐量计算通道，较少控制单元使其不会受到计算以外更多任务干扰，拥有比CPU更纯粹计算环境。



	神经网络	GPUs
天然并行	√	√
矩阵计算	√	√
浮点计算	√	√

Talk Overview

1. AI 计算体系
   - 深度学习计算模式
   - 计算体系与矩阵运算
2. **AI 芯片基础**
   - 通用处理器 CPU
   - 从数据看 CPU 计算
   - 通用图形处理器 GPU
   - AI专用处理器 NPU/TPU
   - 计算体系架构的黄金10年

1. **硬件基础**
   - GPU 工作原理
   - GPU AI 编程本质
2. **英伟达 GPU 架构**
   - 从 Fermi 到 Hopper 架构
   - Tensor Code 和 NVLink 详解
3. **GPU 图形处理流水线**
   - 图形流水线基础
   - GPU 逻辑模块划分
   - 图形处理算法到硬件

引用

1. https://www.youtube.com/watch?v=3jHi8E5C-18
2. https://www.youtube.com/watch?v=-P28LKWTzrI
3. https://www.youtube.com/watch?v=3110o0DYJXg
4. https://infohub.delltechnologies.com/l/white-paper-virtualizing-gpus-for-ai-with-vmware-and-nvidia-based-on-dell-infrastructure-1/nvidia-237
5. https://www.researchgate.net/figure/Architecture-of-compute-intensive-and-data-intensive-processing-paradigm_fig2_334952391