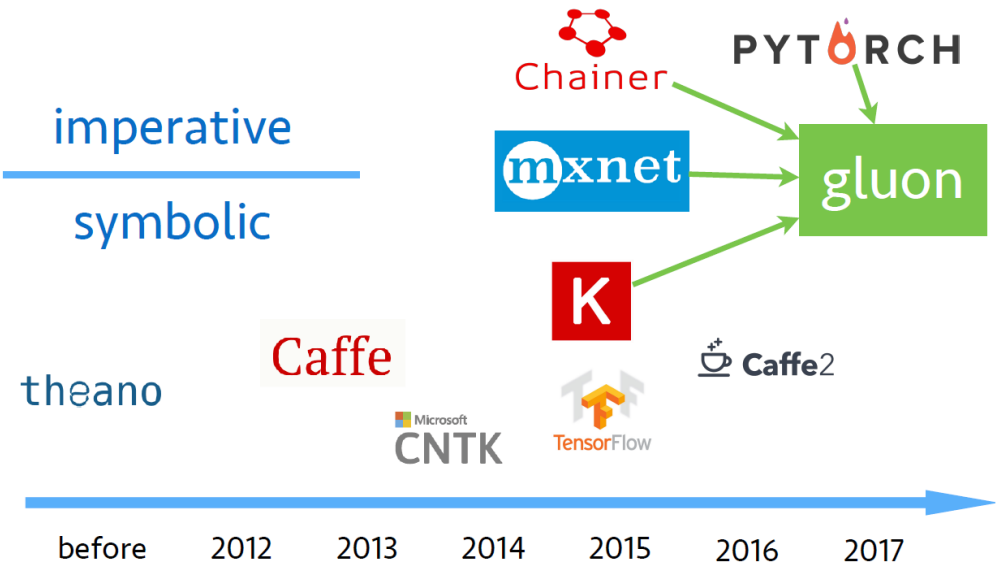
DECSAI
Departamento de Ciencias de la Computación e I.A.
Universidad de Granada

Deep Learning – Herramientas

Fernando Berzal, berzal@acm.org

Herramientas



https://mli.github.io/cvpr17/

Herramientas

- Caffe (University of California, Berkeley)
  http://caffe.berkeleyvision.org/
- Theano (University of Montreal)
  http://deeplearning.net/software/theano/
- Torch (New York University), e.g. Facebook
  http://torch.ch/
- TensorFlow (Google)
  https://www.tensorflow.org/
- CNTK: Computational Network Toolkit (Microsoft)
  http://www.cntk.ai/
- DL4J: Deep Learning for Java (Skymind)
  http://deeplearning4j.org/

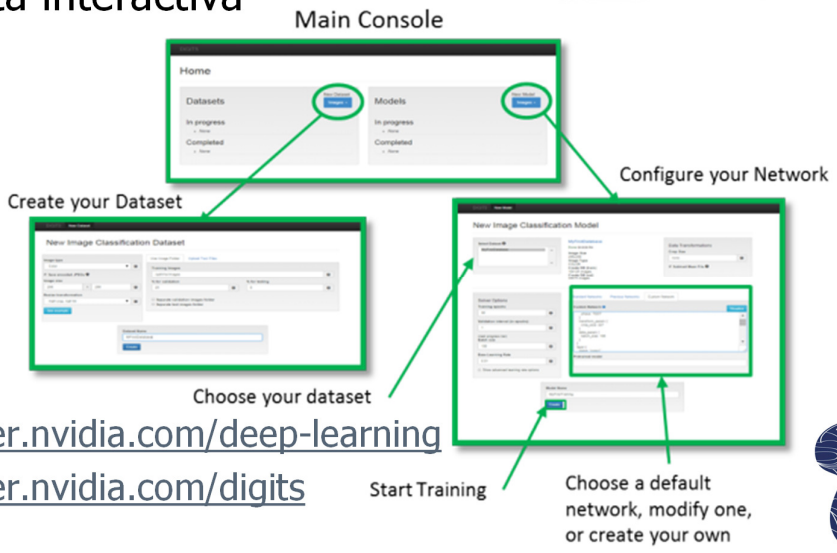
http://deeplearning.net/software_links/
https://en.wikipedia.org/wiki/Deep_learning#Commercial_activities

Herramientas

NVIDIA DIGITS:
Deep Learning GPU Training System

GPUs [Graphics Processing Units]

- Herramienta interactiva



https://developer.nvidia.com/deep-learning
https://developer.nvidia.com/digits

DECSAI
Departamento de Ciencias de la Computación e I.A.
Universidad de Granada

Deep Learning – Herramientas HW

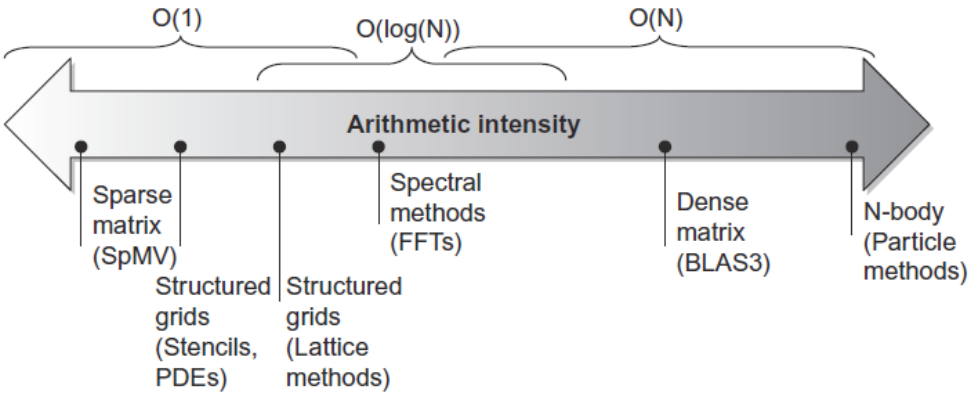
Fernando Berzal, berzal@acm.org

Hardware

Evaluación del rendimiento

Intensidad aritmética

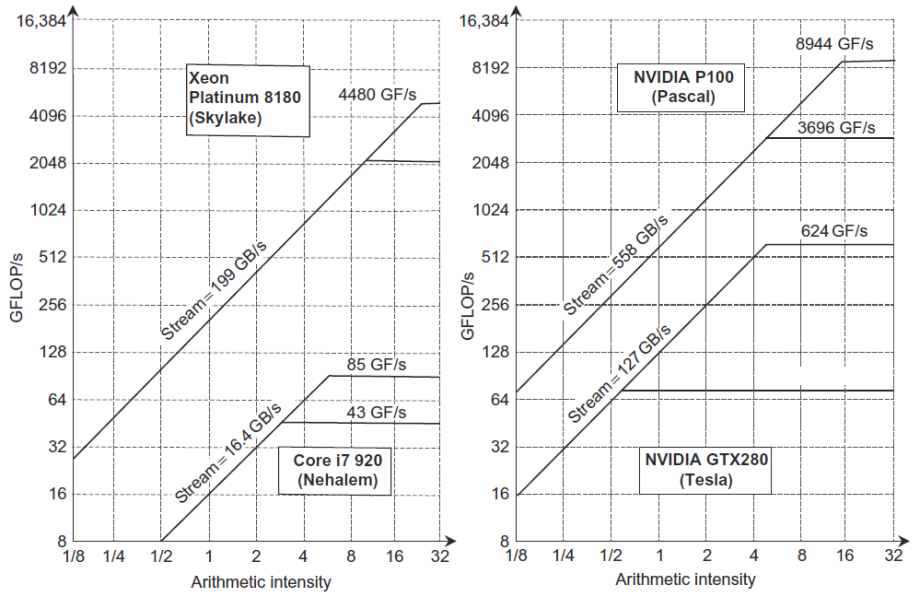
Operaciones en coma flotante (FLOPS)
/ Bytes a los que se accede en memoria principal



Hardware

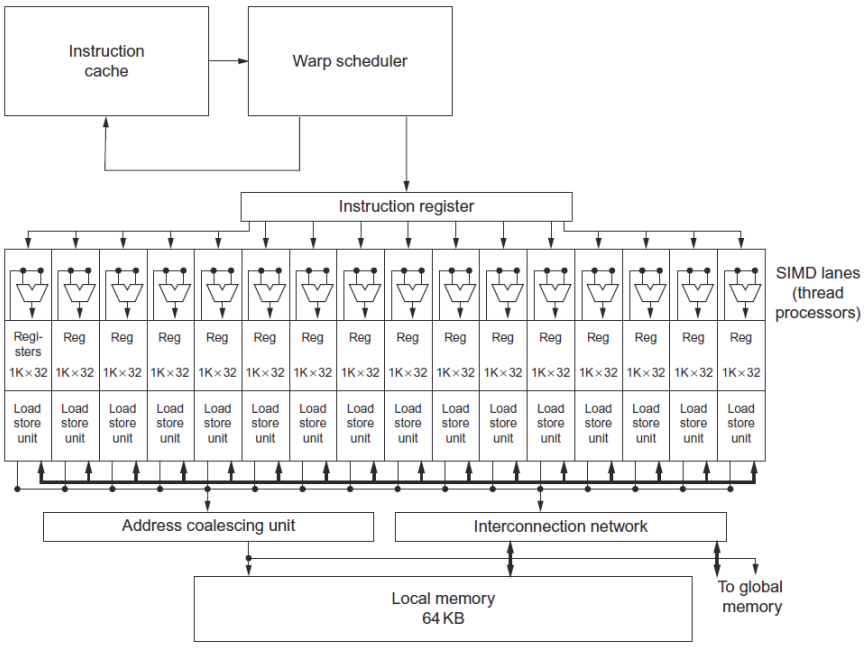
Evaluación del rendimiento

Roofline model: CPU vs. GPU



Hardware

GPU (procesador SIMD): Data-level parallelism



Hardware

NVIDIA Pascal Multithreaded SIMD Processor

Hardware

Pascal P100 GPU

Hardware

GPUs para sistemas empotrados (vehículos autónomos) y para servidores

	NVIDIA Tegra 2	NVIDIA Tesla P100
Market	Automotive, Embedded, Console, Tablet	Desktop, server
System processor	Six-Core ARM (2 Denver2 +4 A57)	Not applicable
System interface	Not applicable	PCI Express × 16 Gen 3
System interface bandwidth	Not applicable	16 GB/s (each direction), 32 GB/s (total)
Clock rate	1.5 GHz	1.4 GHz
SIMD multiprocessors	2	56
SIMD Lanes/SIMD multiprocessor	128	64
Memory interface	128-bit LP-DDR4	4096-bit HBM2
Memory bandwidth	50 GB/s	732 GB/s
Memory capacity	up to 16 GB	up to 16 GB
Transistors	7 billion	15.3 billion
Process	TSMC 16 nm FinFET	TSMC 16 nm FinFET
Die area	147 mm^2	645 mm^2
Power	20 W	300 W

Hardware

CUDA [Compute Unified Device Architecture]

Dialecto de C, similar a OpenCL

- Functions for the GPU device (`__device__` or `__global__`) vs. functions for the host (`__host__`)
- CUDA variables declared with `__device__` are allocated to the GPU memory, which is accessible by all multithreaded SIMD processors.

Hardware

CUDA [Compute Unified Device Architecture]

Dialecto de C, similar a OpenCL

Extended function call syntax for the function name that runs on the GPU is

```
function < <<dimGrid, dimBlock>> > (params)
```

where dimGrid and dimBlock specify the dimensions of the code (in Thread Blocks) and the dimensions of a block (in threads).

Hardware

CUDA [Compute Unified Device Architecture]

Dialecto de C, similar a OpenCL

```
function < <<dimGrid, dimBlock>> > (params)
```

- Identifier for blocks (`blockIdx`) and identifier for each thread in a block (`threadIdx`)
- Keyword for the number of threads per block (`blockDim`), which comes from the `dimBlock` parameter in the GPU function call.

Hardware

CUDA [Compute Unified Device Architecture]

Ejemplo: DAXPY [Double-precision aX+Y] @ CPU

```
// Invoke DAXPY
daxpy(n, a, x, y);

// DAXPY in C
void daxpy (int n, double a, double *x, double *y)
{
  for (int i=0; i<n; i++)
      y[i] = a*x[i] + y[i];
}
```

Hardware

CUDA [Compute Unified Device Architecture]

Ejemplo: DAXPY [Double-precision aX+Y] @ GPU

```
// Invoke DAXPY with 256 threads per Thread Block
__host__ int nblocks = (n+ 255) / 256;
daxpy<<<nblocks, 256>>>(n, 2.0, x, y);

// DAXPY in CUDA (n threads, one per vector element)
__global__
void daxpy(int n, double a, double *x, double *y)
{
  int i = blockIdx.x*blockDim.x + threadIdx.x;
  if (i < n) y[i] = a*x[i] + y[i];
}
```

Hardware

PTX [Parallel Thread Execution]

NVDIA GPU Instruction Set Architecture [ISA]

`opcode.type d, a, b, c;`

- Operandos (`a, b, c`): registros o constantes.
- Destino (`d`): registro o dirección de memoria (`st`).

Control flow instructions are functions `call` and `return`, thread `exit`, `branch`, and `barrier` synchronization for threads within a Thread Block (`bar.sync`).

All instructions can be predicated by 1-bit predicate registers, which can be set by `setp` (set predicate) and before a branch instruction gives us conditional branches.

Hardware

PTX [Parallel Thread Execution]

Ejemplo: DAXPY
[Double-precision aX+Y]

```
for (i=0; i<n; i++)
    Y[i] = a*X[i] + Y[i];
```

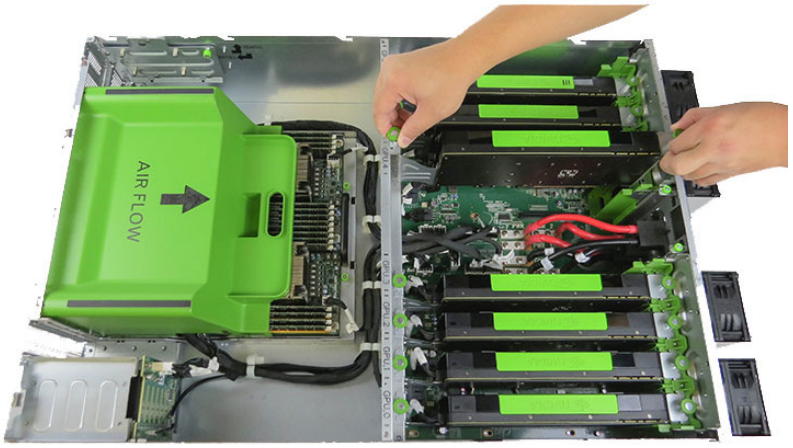
```
shl.u32 R8, blockIdx, 8 ; Thread Block ID * Block size (256=2^8)
add.u32 R8, R8, threadIdx ; R8 = i = my CUDA Thread ID
shl.u32 R8, R8, 3 ; byte offset
ld.global.f64 RD0, [X+R8] ; RD0 = X[i]
ld.global.f64 RD2, [Y+R8] ; RD2 = Y[i]
mul.f64 RD0, RD0, RD4 ; Product in RD0 = RD0 * RD4 (scalar a)
add.f64 RD0, RD0, RD2 ; Sum in RD0 = RD0 + RD2 (Y[i])
st.global.f64 [Y+R8], RD0 ; Y[i] = sum (X[i]*a + Y[i])
```

The CUDA programming model assigns one CUDA Thread to each loop iteration and offers a unique identifier number to each Thread Block (`blockIdx`) and one to each CUDA Thread within a block (`threadIdx`).

Hardware

Facebook Big Sur

Servidor con 8 GPUs NVIDIA Tesla para deep learning

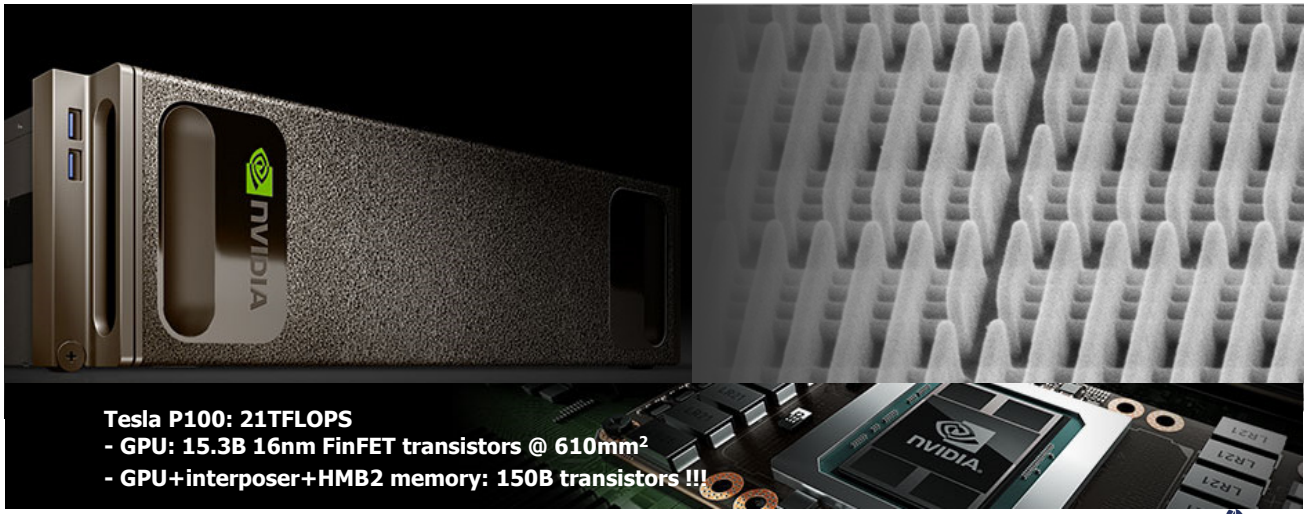


Facebook Engineering Blog, December 2015
https://code.facebook.com/posts/1687861518126048/facebook-to-open-source-ai-hardware-design/

Hardware

NVIDIA DGX-1 deep learning supercomputer $ 129 000

8x GPUs NVIDIA Tesla P100, 28672 CUDA cores

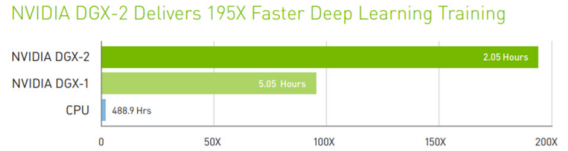


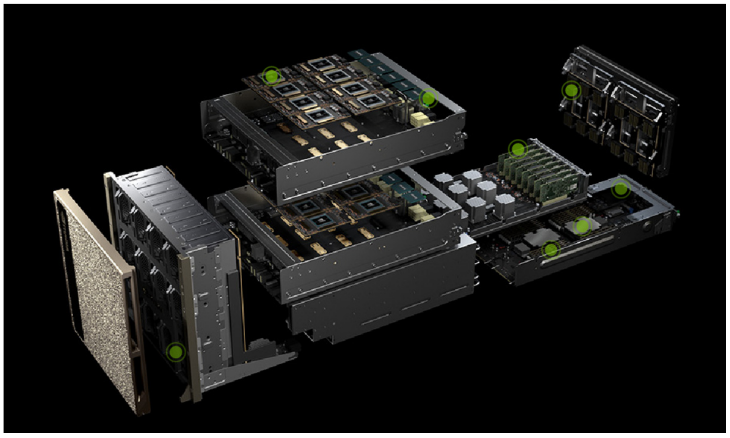
DGX-1: 8xP100, 512 GB DDR4, 4x1.92TB SSD, 170 TFLOPS, 60kg, 3200W
A $2 Billion Chip to Accelerate Artificial Intelligence, MIT Technology Review, April 2016
https://www.technologyreview.com/s/601195/a-2-billion-chip-to-accelerate-artificial-intelligence/
http://www.nvidia.com/object/deep-learning-system.html

Hardware

NVIDIA DGX-2 deep learning supercomputer $ 399 000

16x GPUs NVIDIA Tesla V100, 81920 CUDA cores, 2 petaFLOPS





DGX-2: 16x V100, 96MB SRAM, 512 GB DDR, 30TB SSD, 2 petaFLOPS, 10kW, 163kg

https://www.nvidia.com/en-us/data-center/dgx-2/

Hardware

NVIDIA DGX A100 $ 199 000

8x NVIDIA Ampere A100 Tensor Core GPUs, 5 petaFLOPS

A100 accelerator (May 2020): TSMC 7nm N7, 54.2B transistors, 6912 CUDA cores, 400W

DGX A100: 8x A100, 320GB GPU memory, 1TB system, 15TB SSD, 5 petaFLOPS, 6.5kW, 123kg

https://www.nvidia.com/en-us/data-center/dgx-a100/

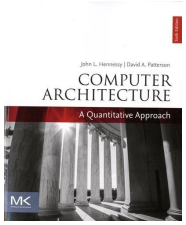
Hardware especializado

DSAs [Domain-Specific Architectures]

- **ASICs** [Application-Specific Integrated Circuits]
  - Mayor coste NRE [Non recurrent engineeering]
  - Mayor rendimiento
- Circuitos reconfigurables,
  p.ej. **FPGAs** [Field-Programmable Gate Arrays]
  - Menor coste NRE
  - Menor rendimiento

Hardware especializado

DSAs [Domain-Specific Architectures]



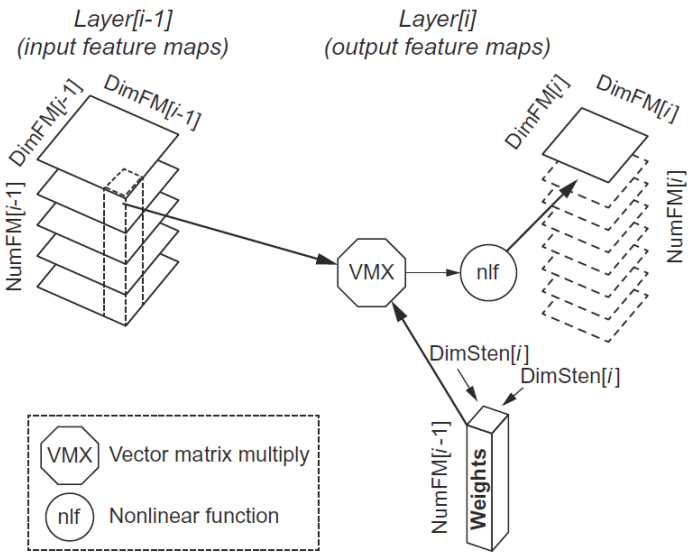
Diseño energéticamente eficiente

- Memorias dedicadas [scratchpad] gestionadas por el programador en lugar de memorias caché.
- Recursos ahorrados en la microarquitectura dedicados a más unidades funcionales o más memoria.
- Paralelismo ajustado al dominio de aplicación (SIMD).
- Reducción de la precisión (8-, 16-bit), para aprovechar mejor el ancho de banda de memoria.
- DSL [domain-specific language] para portar código a la DSA, p.ej. Halide (visión) o TensorFlow (DNNs).

Hardware especializado

DSAs [Domain-Specific Architectures]

Ejemplo: CNN



Intensidad aritmética:

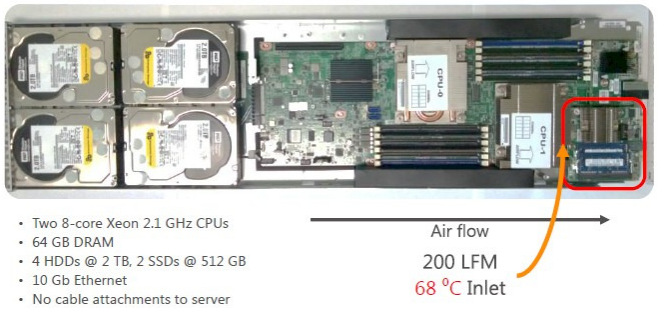
operations/weight = 2 x DimFM[i]²

Hardware especializado

Microsoft Research Catapult

FPGAs [Field Programmable Gate Arrays]

- Menor consumo de energía: 25W
- Menor ancho de banda: 11GB/s
  (datos en la memoria DDR3 de la propia FPGA para evitar PCIe)



CPU-FPGA minimalista:
FPGA Altera Stratix V @ Open CloudServer

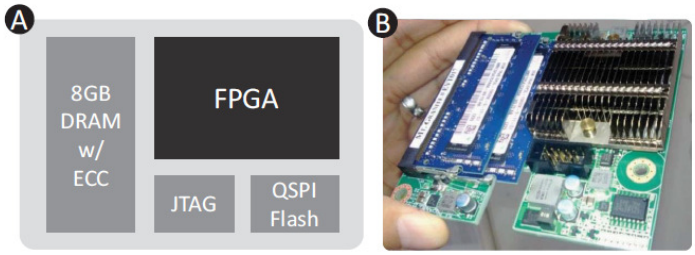
Toward Accelerating Deep Learning at Scale Using Specialized Logic
HOTCHIPS'2015: A Symposium on High Performance Chips, August 2015

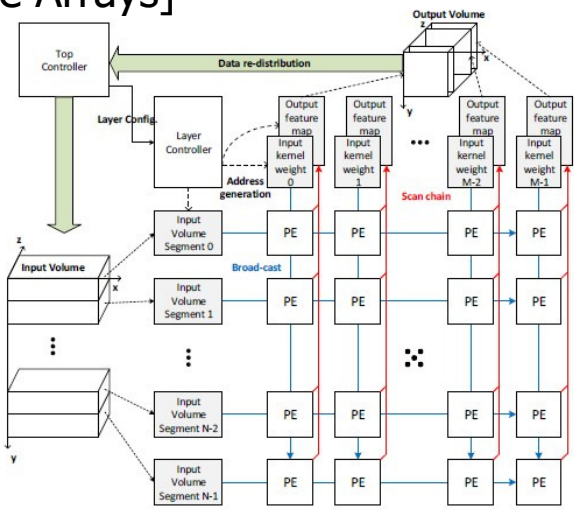
Hardware especializado

Microsoft Catapult v1

FPGAs [Field Programmable Gate Arrays]

- 3926 18-bit ALUs
- 2D array of PEs
- Red 20Gbps entre 48 FPGAs (topología de toro 6x8)

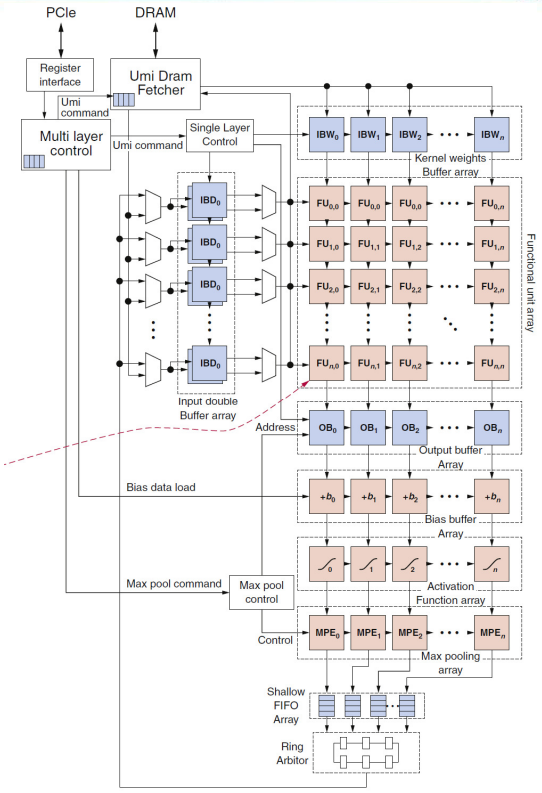




Hardware especializado

Microsoft Catapult v1

CNN Accelerator

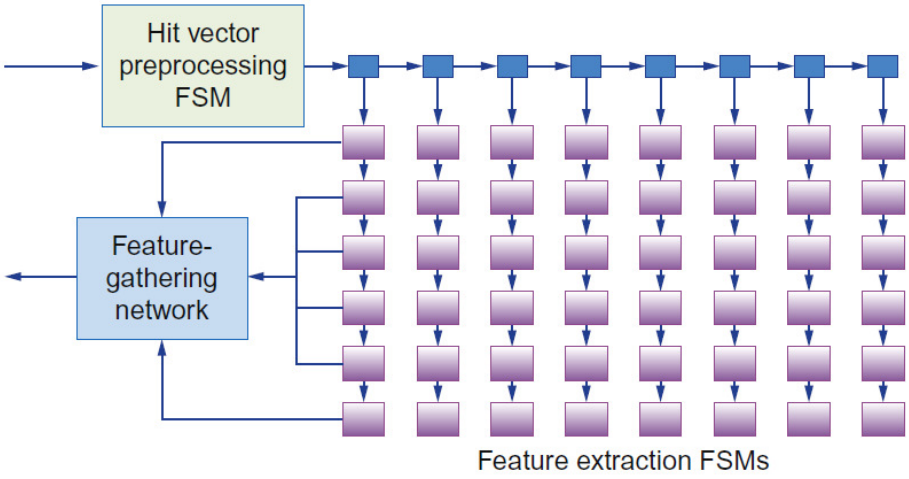


FU [Functional Unit]
= ALU + Registers

Hardware especializado

Microsoft Catapult v1

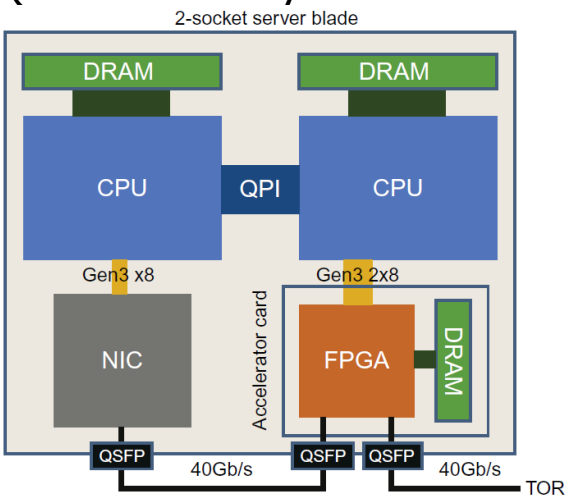
Search engine ranking @ Bing

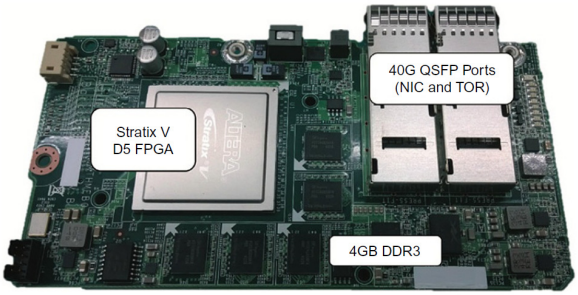


Hardware especializado

Microsoft Catapult v2

Cambio en la posición de la FPGA en el centro de datos (entre la CPU y el NIC de cada servidor, 32W)





Usos: CNN, Bing & Azure Networking

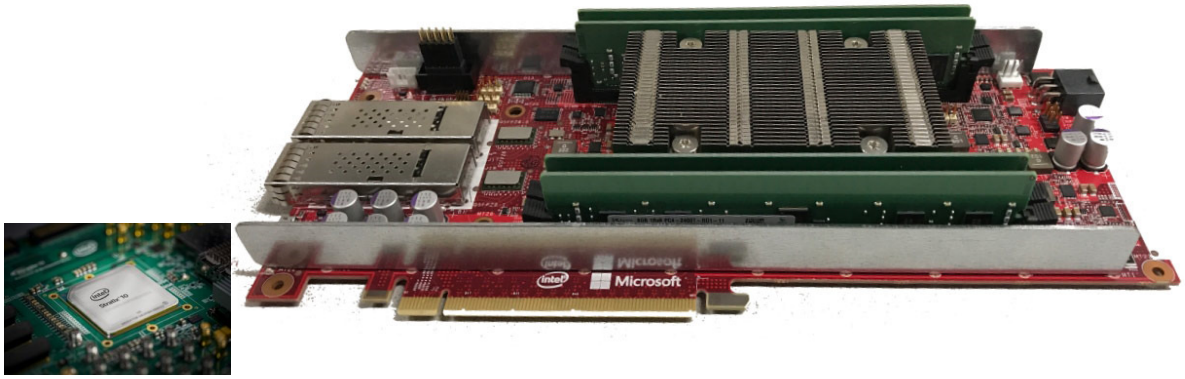
A cloud-scale acceleration architecture. MICRO Conference, 2016

Hardware especializado

Microsoft Research Project BrainWave

FPGAs [Field Programmable Gate Arrays]

- DNNs as "hardware microservices"



FPGA Intel Stratix 10 (e.g. GRU @ 39.5TFLOPS)

Accelerating Persistent Neural Networks at Datacenter Scale

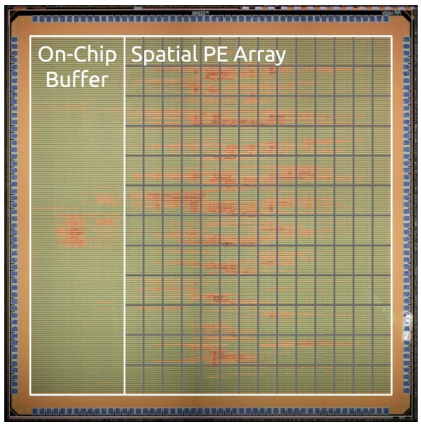
HOTCHIPS'2017 & NIPS'2017

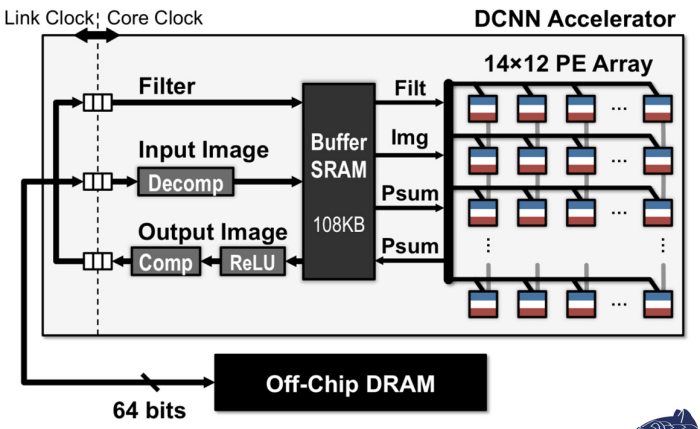
Hardware especializado

MIT Eyeriss

168 PE [Processing Elements], 0.3W (<10% mobile GPU)

http://www.mit.edu/~sze/eyeriss.html





A Deep Learning AI Chip for Your Phone, IEEE Spectrum, February 2016

http://spectrum.ieee.org/tech-talk/semiconductors/processors/a-deep-learning-ai-chip-for-your-phone

Hardware especializado

TPU [Tensor Processing Unit]

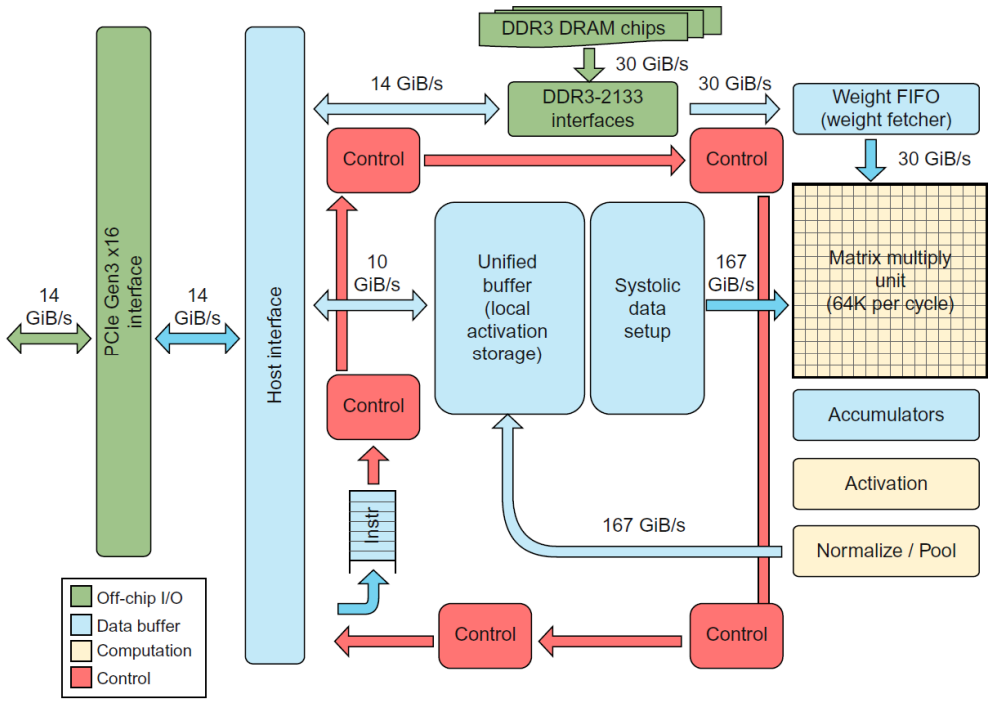
Google, 2015-

Características

- 65,536 (256x256) 8-bit ALU Matrix Multiply Unit
- Large software-managed on-chip memory
- Single-threaded, deterministic execution model
- Coprocessor on the PCIe I/O bus
- The host server sends instructions over the PCIe bus directly to the TPU for it to execute, rather than having the TPU fetch the instructions (closer to FPUs than to GPUs).

Hardware especializado

TPU [Tensor Processing Unit]



Hardware especializado

TPU [Tensor Processing Unit]

CISC ISA (due to slow PCIe bus)

- `Read_Host_Memory`
  (CPU Host Memory -> Unified Buffer)
- `Read_Weights`
  (Weight Memory -> Weight Buffer)
- `MatrixMultiply / Convolve`
  (Unified Buffer -> Accumulators)
- `Activate`
  (Accumulators -> Outout Buffer)
- `Write_Host_Memory`
  (Unified Buffer -> CPU Host Memory)

Clock cycles per instruction (CPI) ~ 10-20

Hardware especializado

TPU Microarchitecture

- Goal:

  Keep the Matrix Multiply Unit busy.

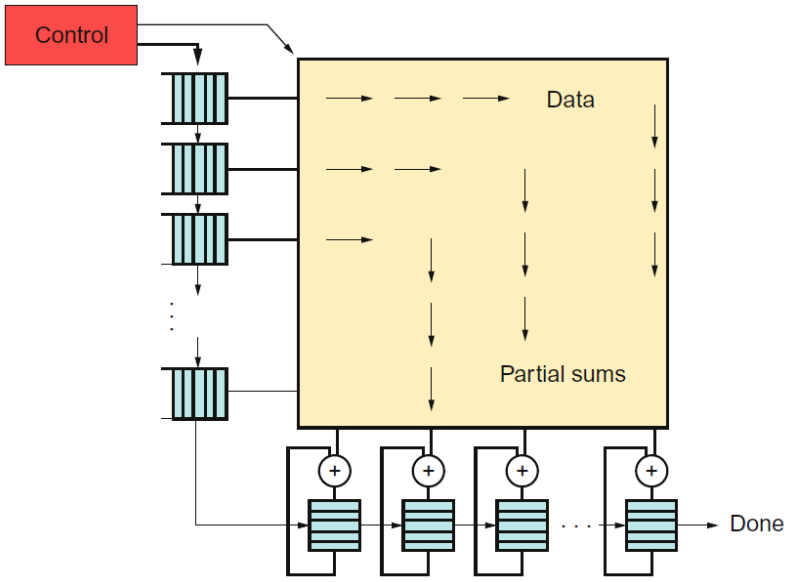
- Plan:

  Hide the execution of the other instructions by overlapping their execution with the MatrixMultiply instruction

Hardware especializado

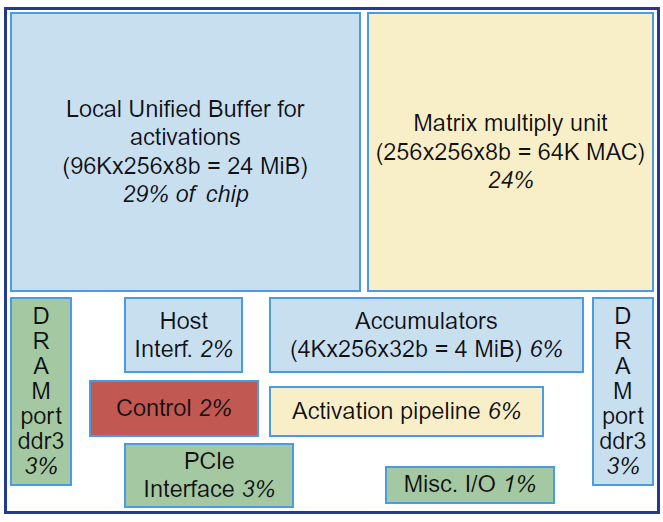
TPU Microarchitecture

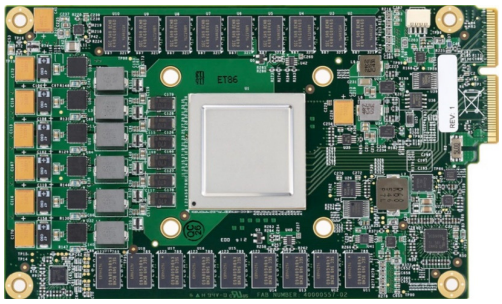
The Matrix Multiply Unit uses systolic execution to save energy by reducing reads and writes of the Unified Buffer



Hardware especializado

TPU Implementation





TPU die floor plan

Hardware especializado

TPU Programming

- The portion of the application run on the TPU is typically written using TensorFlow and is compiled into an API that can run on GPUs or TPUs.

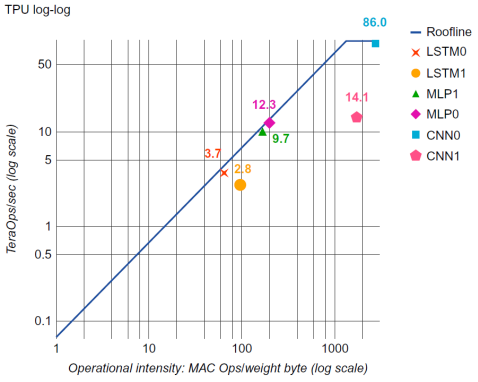
TPU Software

- Lightweight kernel driver:
  Memory management and interrupts.
- User space driver:
  Sets up and controls TPU execution, reformats data into TPU order, and translates API calls into TPU instructions and turns them into an application binary

Hardware especializado

TPU Applications

		DNN layers							
Name	LOC	FC	Conv	Element	Pool	Total	Weights	TPU Ops/Weight	% deployed TPUs 2016
MLP0	100	5				5	20M	200	61%
MLP1	1000	4				4	5M	168	
LSTM0	1000	24		34		58	52M	64	29%
LSTM1	1500	37		19		56	34M	96	
CNN0	1000		16			16	8M	2888	5%
CNN1	1000	4	72		13	89	100M	1750	



6 applications, 95% workload @ Google

- MLP: RankBrain
- LSTM: Google Neural Translator
- CNN: DeepMind AlphaGo

Hardware especializado

TPU Terminology

Accumulators	—	The 4096 256 × 32-bit registers (4 MiB) that collect the output of the MMU and are input to the Activation Unit
Activation unit	—	Performs the nonlinear functions (ReLU, sigmoid, hyperbolic tangent, max pool, and average pool). Its input comes from the Accumulators and its output goes to the Unified Buffer
Matrix multiply unit	MMU	A systolic array of 256 × 256 8-bit arithmetic units that perform multiply-add. Its inputs are the Weight Memory and the Unified Buffer, and its output is the Accumulators
Systolic array	—	An array of processing units that in lockstep input data from upstream neighbors, compute partial results, and pass some inputs and results to downstream neighbors
Unified buffer	UB	A 24 MiB on-chip memory that holds the activations. It was sized to try to avoid spilling activations to DRAM when running a DNN
Weight memory	—	An 8 MiB external DRAM chip containing the weights for the MMU. Weights are transferred to a *Weight FIFO* before entering the MMU

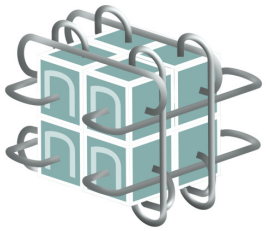
Hardware especializado

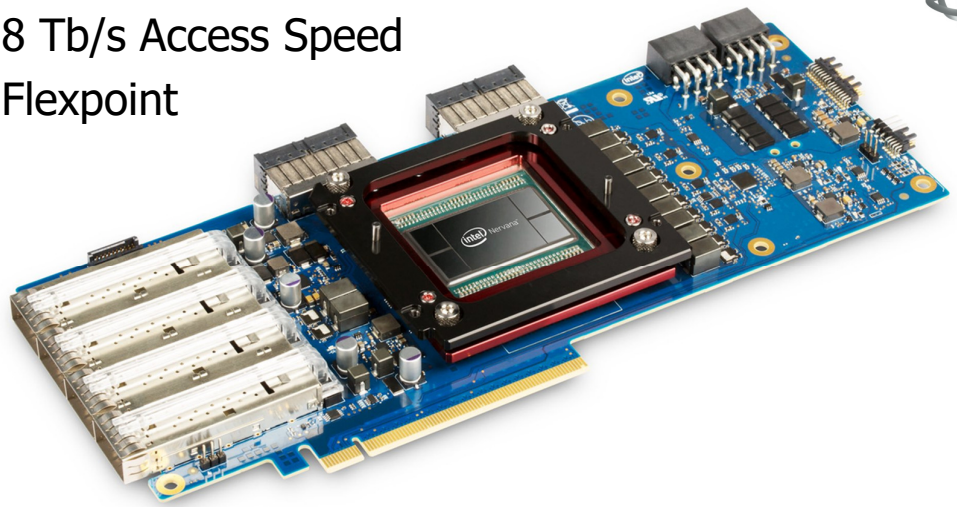
CPU Intel i7	GPU NVIDIA	FPGA Catapult	ASIC TPU
SIMD extensions (MMX, SSE, AVX)	Streaming multiprocessors	3926 18-bit PEs (reconfigurable)	256x256 Matrix Multiply Unit
1D SIMD		2D SIMD	
Multithreading		Pipelined systolic array	
Out-of-order execution	Multiprocessing	Programmable controller	Simple execution of instructions
32 & 64-bit FP	32 & 64-bit FP	18-bit integers	8-bit integers
x86 ISA (C, Java, Python…)	**PTX** (CUDA, OpenCL)	**RTL** (Verilog, VHDL)	**Reduced CISC** ISA (API via DSLs: Halide, TensorFlow)

Hardware especializado

Intel Nervana Neural Network Processor (NNP)

"Lake Crest"
32 GB HBM2,
1 TB/s Bandwidth
8 Tb/s Access Speed
Flexpoint

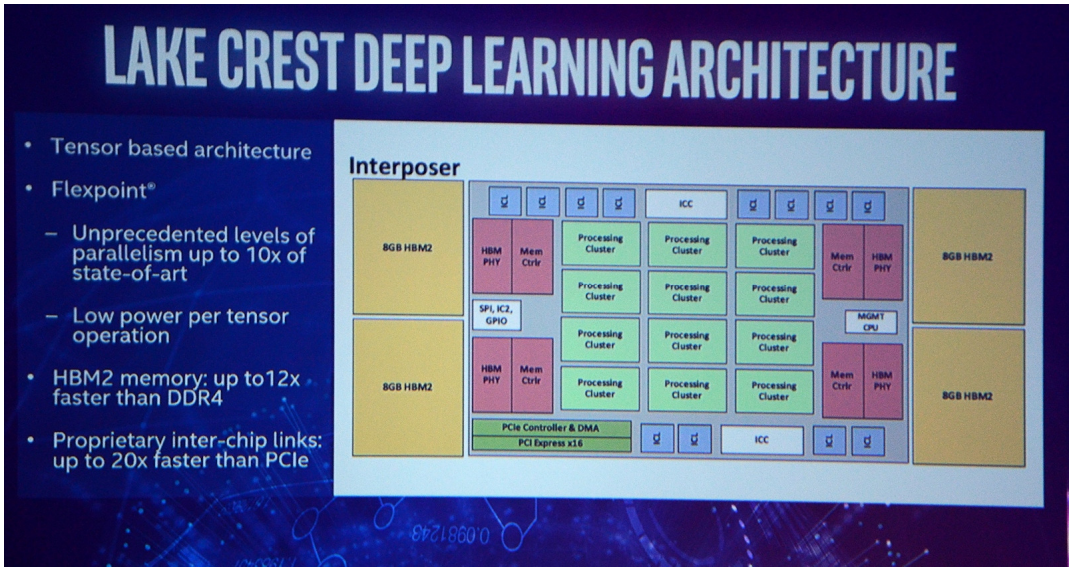




Hardware especializado

Intel Nervana Neural Network Processor (NNP)

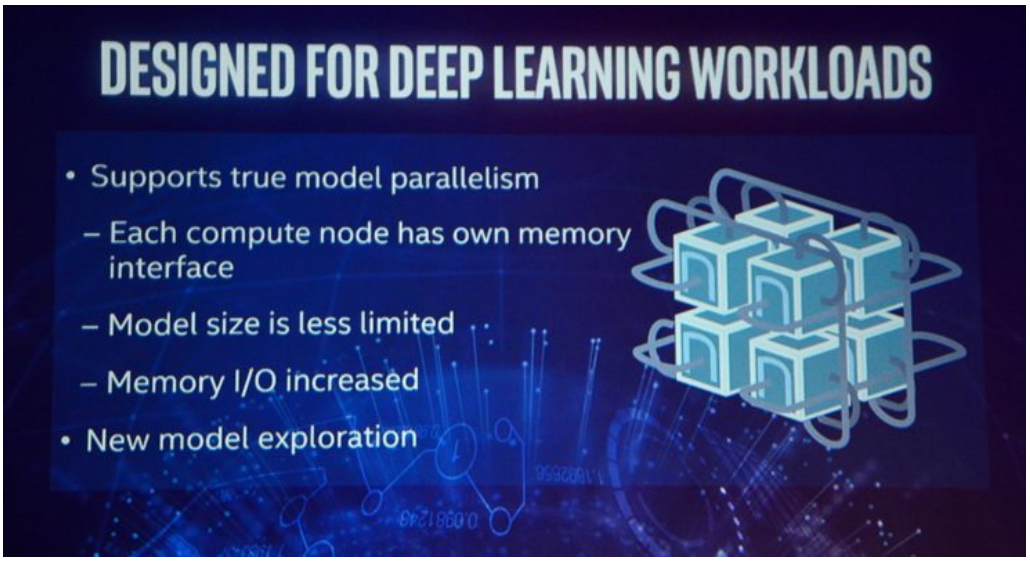
"Lake Crest"



Hardware especializado

Intel Nervana Neural Network Processor (NNP)

"Lake Crest"

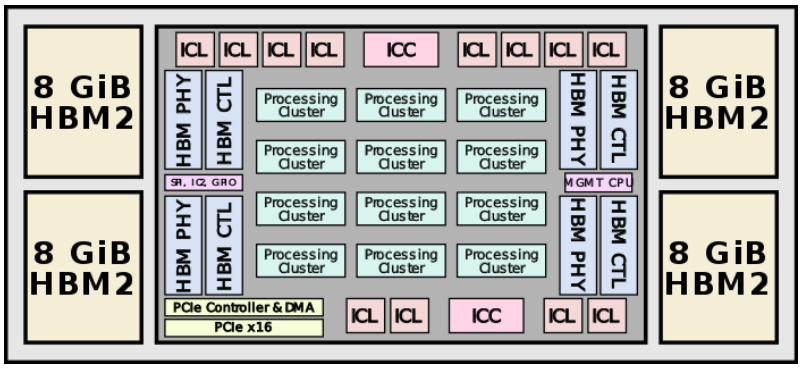


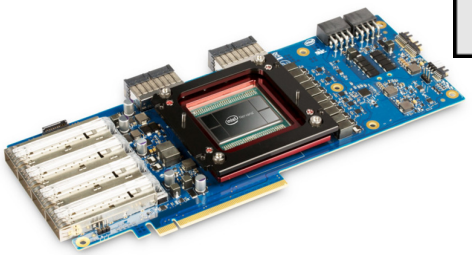
Hardware especializado

Intel Nervana Neural Network Processor (NNP)

"Lake Crest", 1st generation NNP, 2016

nervana SYSTEMS



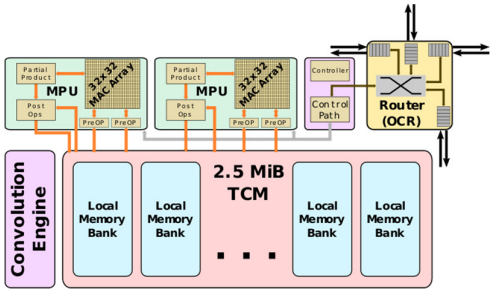


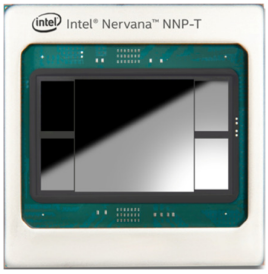
Lake Crest Accelerator PCIe card

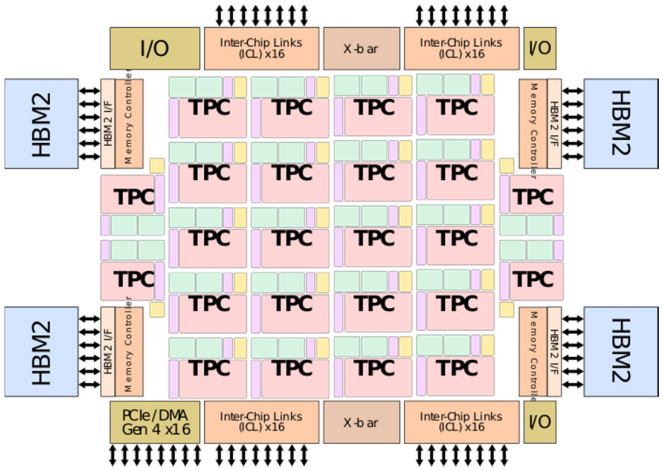
Hardware especializado

Intel Nervana Neural Network Processor (NNP)

"Spring Crest", 2nd generation NNP, 2019







NNP T-1400, 375W, 108TFLOPS

Hardware especializado

Intel Nervana Neural Network Processor (NNP)

Febrero 2020:

"Intel Axes Nervana Just Two Months After Launch"

Habana HLS-1 (8x HL-205)

habana GOYA™

Hardware especializado

Intel + Habana Labs

$2bn

News Release

December 16, 2019

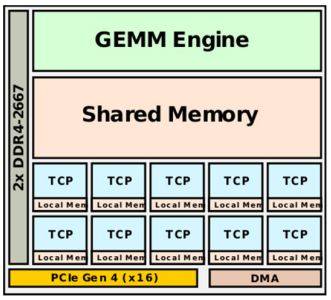
Contact Intel PR

Intel Acquires Artificial Intelligence Chipmaker Habana Labs

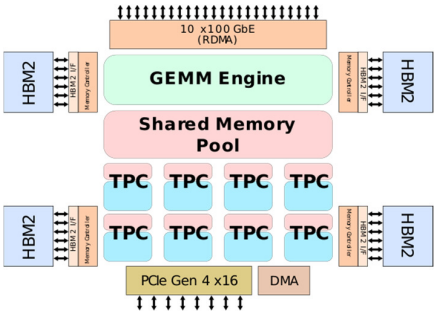
Combination Advances Intel's AI Strategy, Strengthens Portfolio of AI Accelerators for the Data Center

SANTA CLARA Calif., Dec. 16, 2019 – Intel Corporation today announced that it has acquired Habana Labs, an Israel-based developer of programmable deep learning accelerators for the data center for approximately $2 billion. The combination strengthens Intel's artificial intelligence (AI) portfolio and accelerates its efforts in the nascent, fast-growing AI silicon market, which Intel expects to be greater than $25 billion by 2024[1].

habana GOYA



habana GAUDI



Hardware especializado

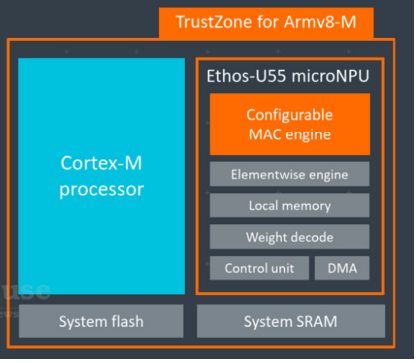
ARM Ethos microNPU

Cortex M55 CPU + Ethos U55 NPU (2020)

arm ETHOS

Ethos-U55: The First microNPU for Cortex-M

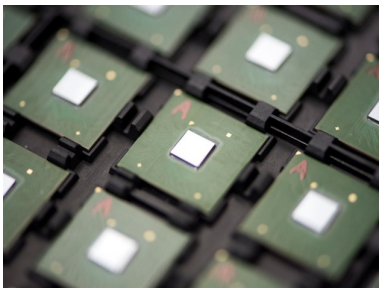
- ✓ Highest efficiency and small memory footprint
- ✓ 32, 64, 128, or 256 unit multiply-accumulate (MAC) engine
- ✓ Weight decoder and DMA for on-the-fly weight decompression
- ✓ Tooling available for offline optimization
- ✓ Works with a range of Cortex-M processors:
  - Cortex-M55
  - Cortex-M7
  - Cortex-M33
  - Cortex-M4



Hardware especializado

Implementaciones analógicas

… para reducir el consumo energético



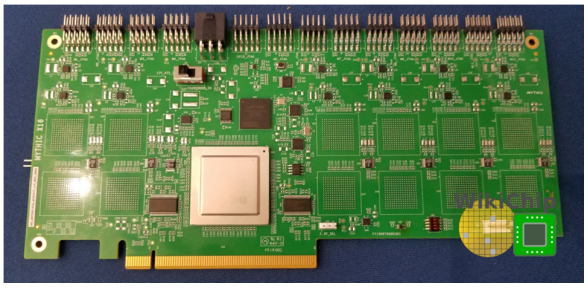
Trillion operations per watt (TOPS/W)

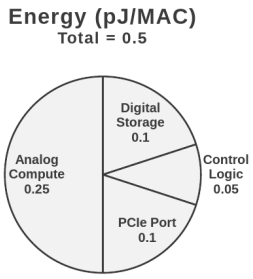
- 0.4 TOPS/W: NVIDIA Volta V100 GPU
- 4 TOPS/W: **Mythic AI** analog in-memory computing, (analog flash array + programmable digital circuit) https://www.mythic-ai.com/
- 20 TOPS/W: **Syntiant** Neural Decision Processor (NDP), (entire analog network) https://www.syntiant.com/

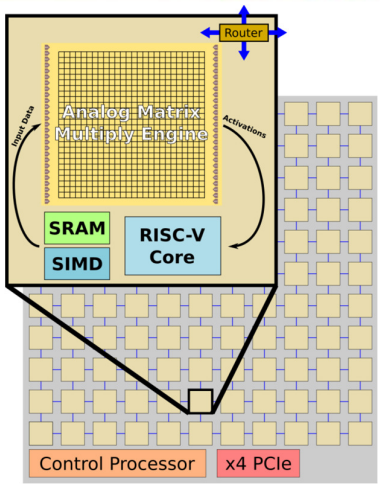
Two Startups Use Processing in Flash Memory for AI at the Edge, IEEE Spectrum, August 2018
https://spectrum.ieee.org/tech-talk/computing/embedded-systems/two-startups-use-processing-in-flash-memory-for-ai-at-the-edge

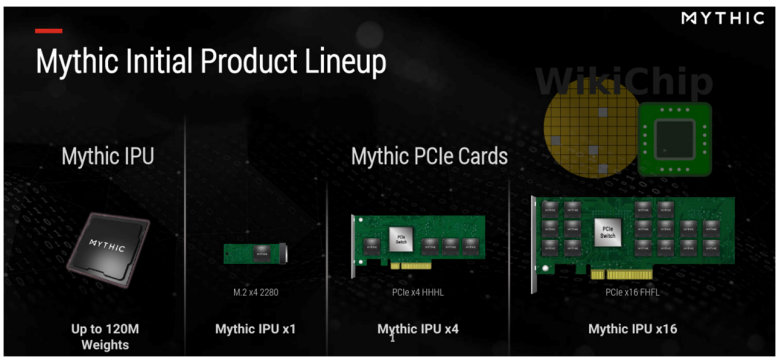
Hardware especializado

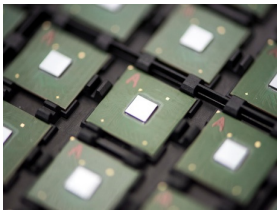
Mythic AI IPU







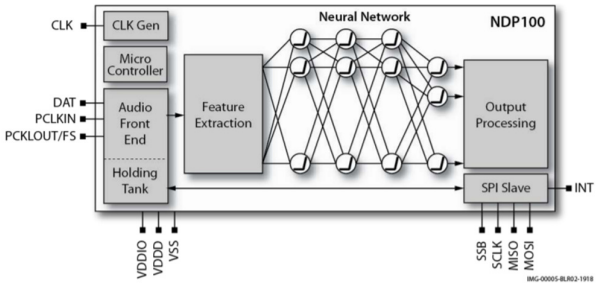


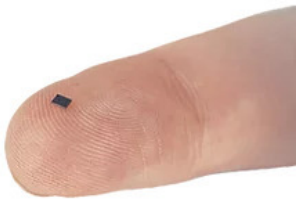


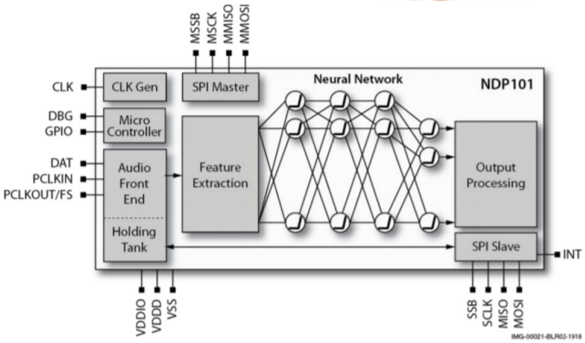
Hardware especializado

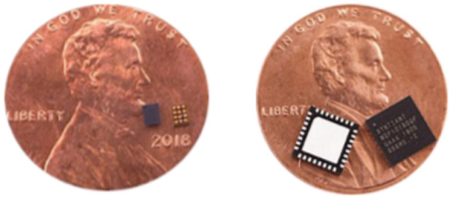
Syntiant NDP [Neural Decision Processor]

<140μW





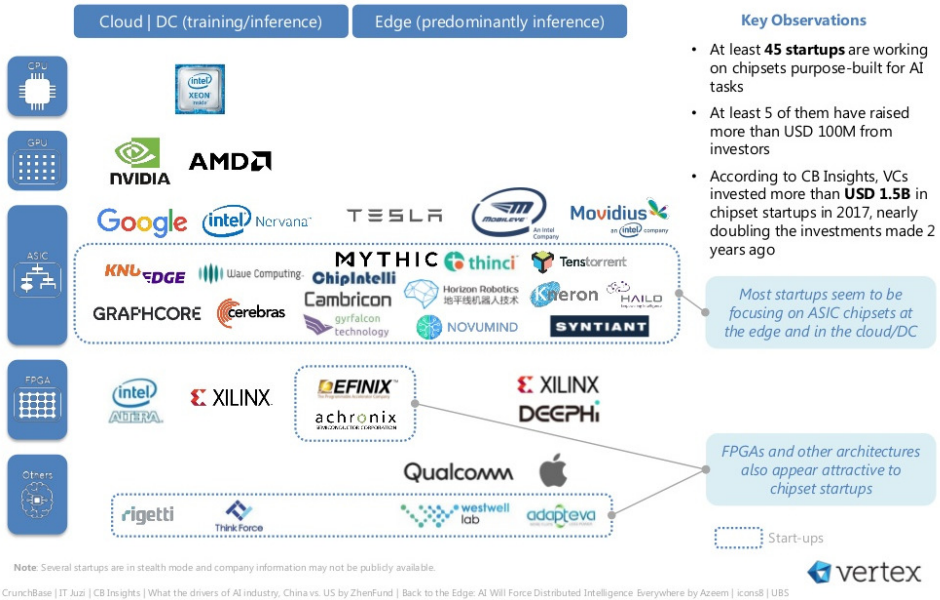




Syntiant NDP100 & NDP101

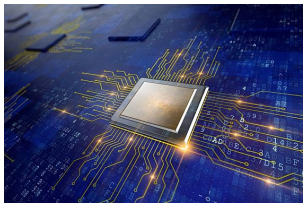
Hardware especializado

Docenas de empresas desarrollan chips para IA…



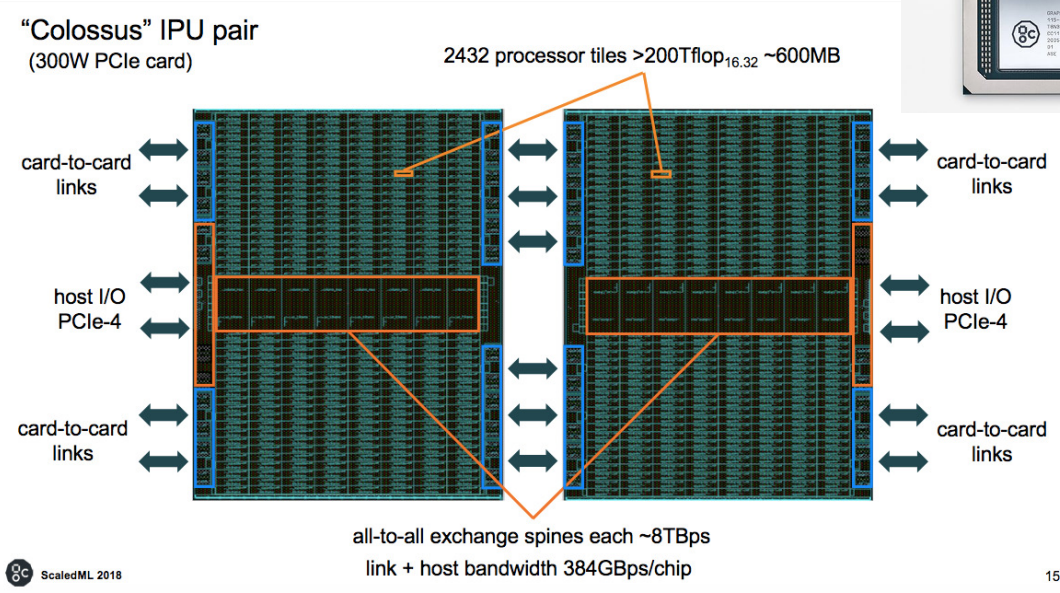
Key Observations

- At least **45 startups** are working on chipsets purpose-built for AI tasks
- At least 5 of them have raised more than USD 100M from investors
- According to CB Insights, VCs invested more than **USD 1.5B** in chipset startups in 2017, nearly doubling the investments made 2 years ago



Hardware especializado

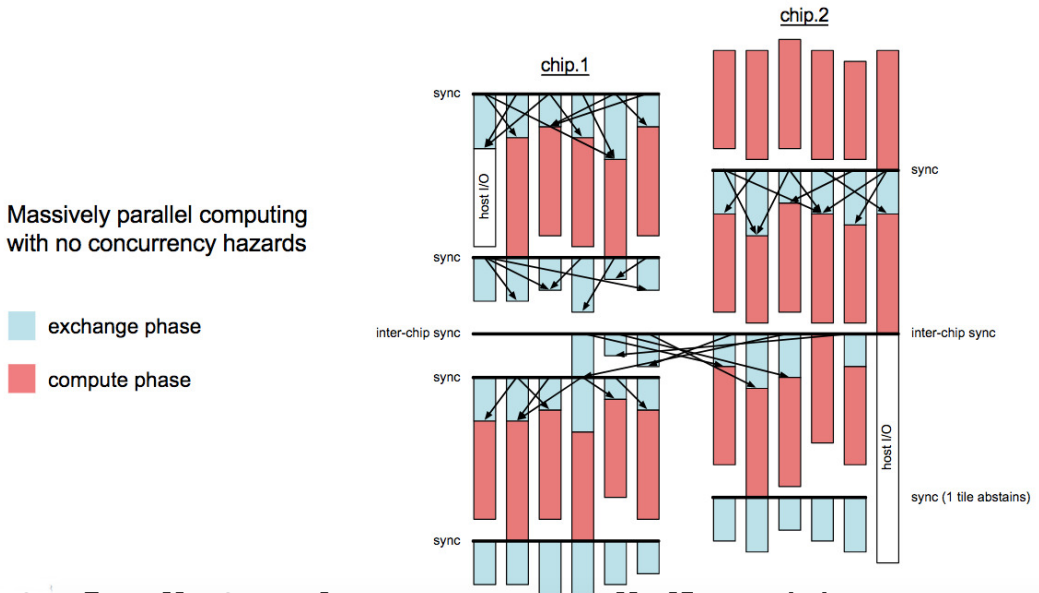
Graphcore Colossus IPU, UK



IPU [Intelligent Processing Unit] pair

Hardware especializado

Graphcore Colossus IPU, UK



BSP [Bulk Synchronous Parallel] model
https://en.wikipedia.org/wiki/Bulk_synchronous_parallel

Hardware especializado

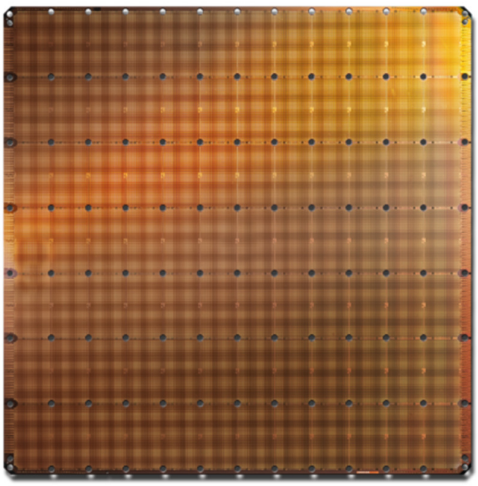
Graphcore Colossus IPU, UK

1 petaFLOP
IPU M2000 machine

Dell DSS8440

Hardware especializado

Cerebras WSE [Wafer-Scale Engine]
>400000 AI cores, 18GB SRAM, 17.5kW

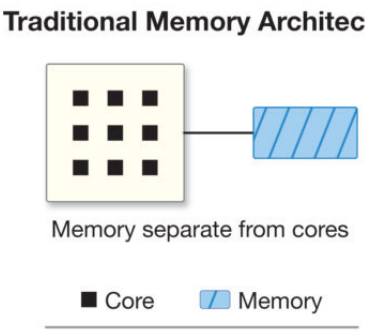


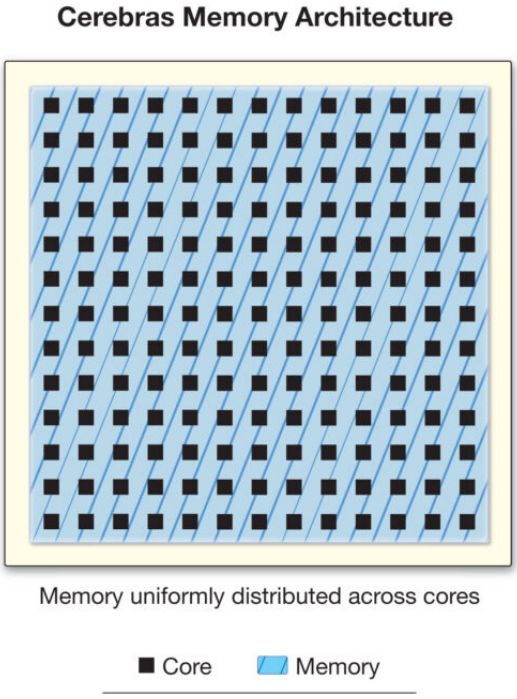
Cerebras WSE
1.2 Trillion transistors
46,225 mm² silicon

Largest GPU
21.1 Billion transistors
815 mm² silicon

Hardware especializado

Cerebras WSE [Wafer-Scale Engine]





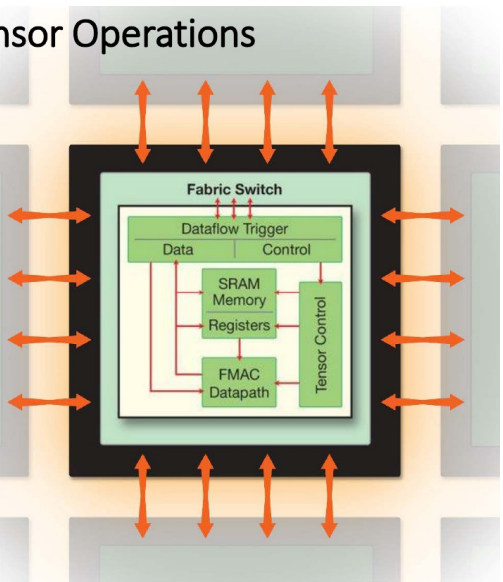
Hardware especializado

Cerebras WSE [Wafer-Scale Engine]

Flexible Cores Optimized for Tensor Operations

Key to supporting rapidly evolving NN architectures

- Fully programmable compute core
- Full array of general instructions with ML extensions
- Flexible **general ops** for control processing
  - e.g. arithmetic, logical, load/store, branch
- Optimized **tensor ops** for data processing
  - Tensors as first class operands
  - e.g. `fmac [z] = [z], [w], a`
    `3D 3D 2D scalar`



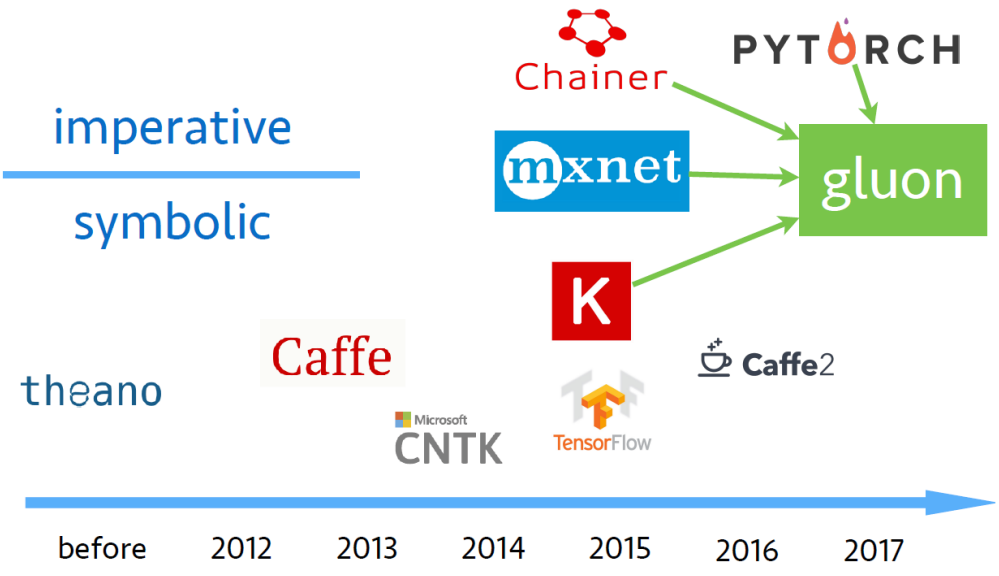
cerebras

DECSAI
Departamento de Ciencias de la Computación e I.A.
Universidad de Granada

Deep Learning – Herramientas SW

Fernando Berzal, berzal@acm.org

Herramientas



https://mli.github.io/cvpr17/

Herramientas

- Caffe (University of California, Berkeley)
  C++ / Python / Matlab
  http://caffe.berkeleyvision.org/
- Theano (University of Montreal)
  Python
  http://deeplearning.net/software/theano/
- Torch (New York University), e.g. Facebook
  C / Lua
  http://torch.ch/
- TensorFlow (Google)
  Python ++
  https://www.tensorflow.org/
- CNTK: Computational Network Toolkit (Microsoft)
  C++ / Python (Keras)
  http://www.cntk.ai/

Herramientas

- Keras (François Chollet)
  Python (backends: TensorFlow, Theano, CNTK, MXNet…)
  https://keras.io/
- MXNet (Apache), p.ej. Amazon
  Multilenguaje: C++ / Python / Julia / Matlab / JS / Go / R / Scala…
  https://mxnet.apache.org/
- Gluon API (Amazon & Microsoft, sobre MXNet)
  Python (backend: MXNet)
  https://github.com/gluon-api/gluon-api
- PyTorch (Facebook, C++ & Python)
  Python / C++ / Julia
  https://pytorch.org/
- Fast.ai (Jeremy Howard & Rachel Thomas)
  Python (backend: PyTorch)
  https://www.fast.ai/

Herramientas

- Chainer (Preferred Networks, Japan)
  Python
  https://chainer.org/
- DyNet (CMU)
  Python
  http://dynet.io/
- Matlab Deep Learning Toolbox (Matlab)
  Matlab
  https://www.mathworks.com/products/deep-learning.html
- PlaidML (Vertex.AI & Intel)
  Python / C++
  https://www.mathworks.com/products/deep-learning.html
- DL4J: Deep Learning for Java (Skymind)
  Java
  http://deeplearning4j.org/

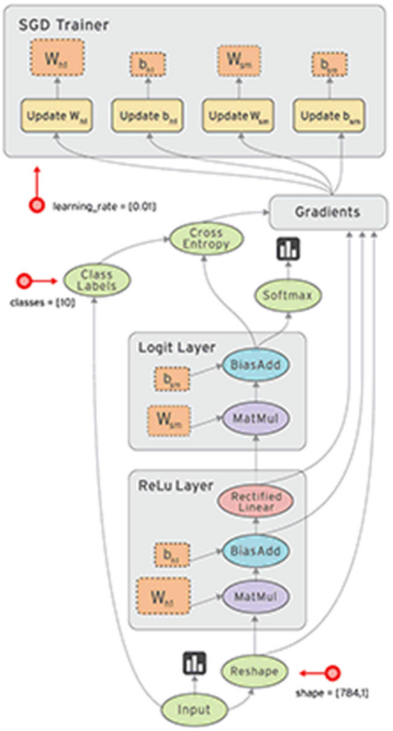
Herramientas

Google TensorFlow

https://www.tensorflow.org/

Licencia Apache 2.0

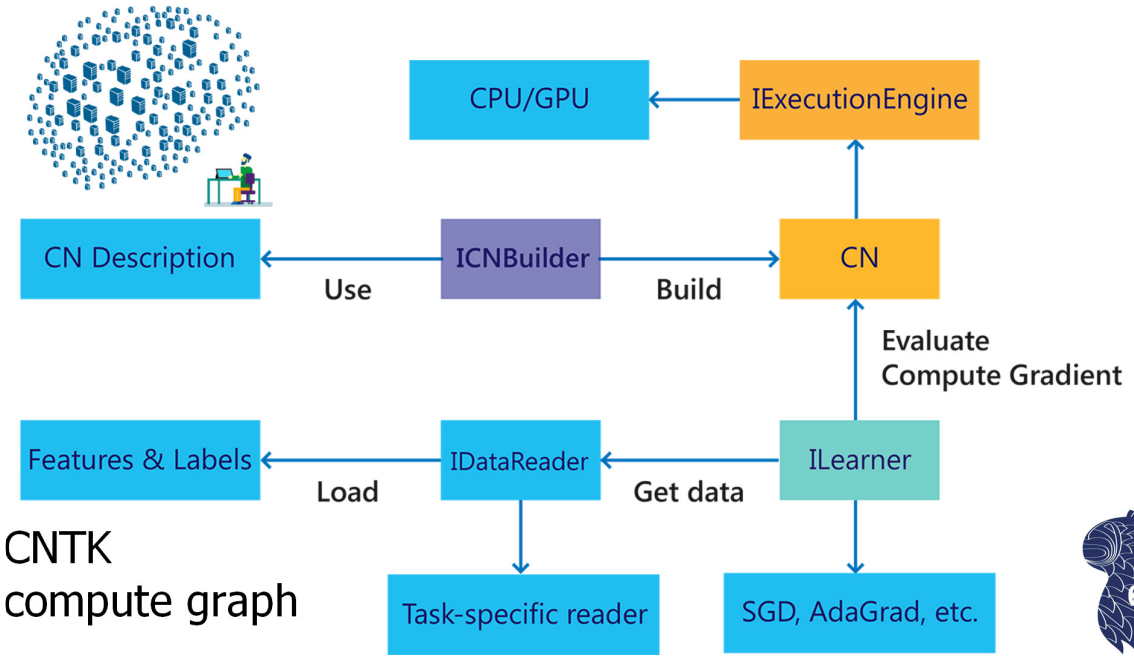
Data flow graph



Herramientas

Microsoft CNTK [Computational Network Toolkit]

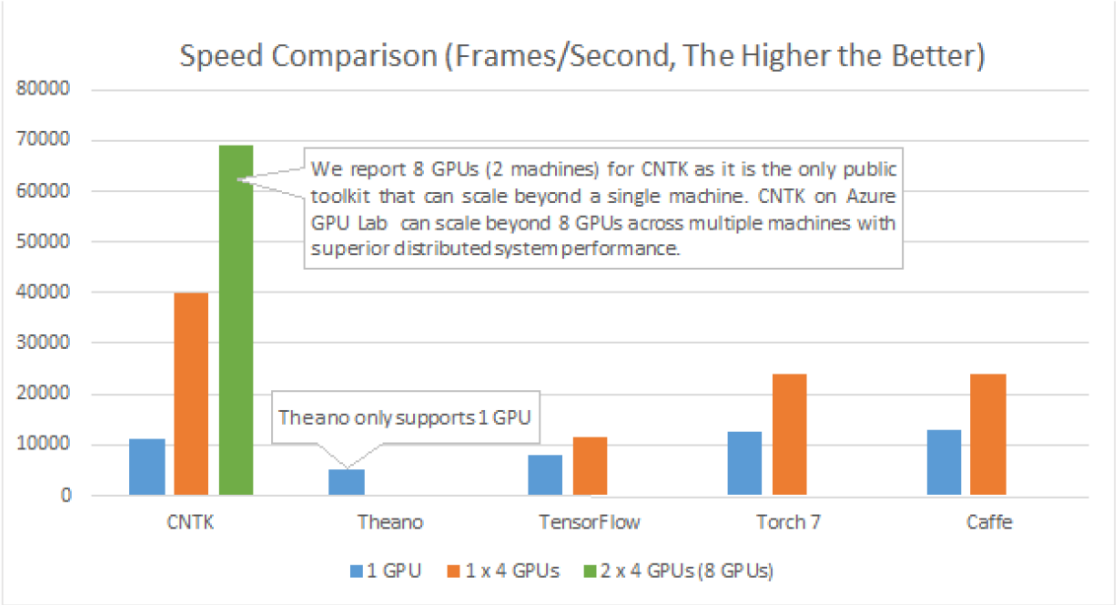
http://www.cntk.ai/



CNTK
compute graph

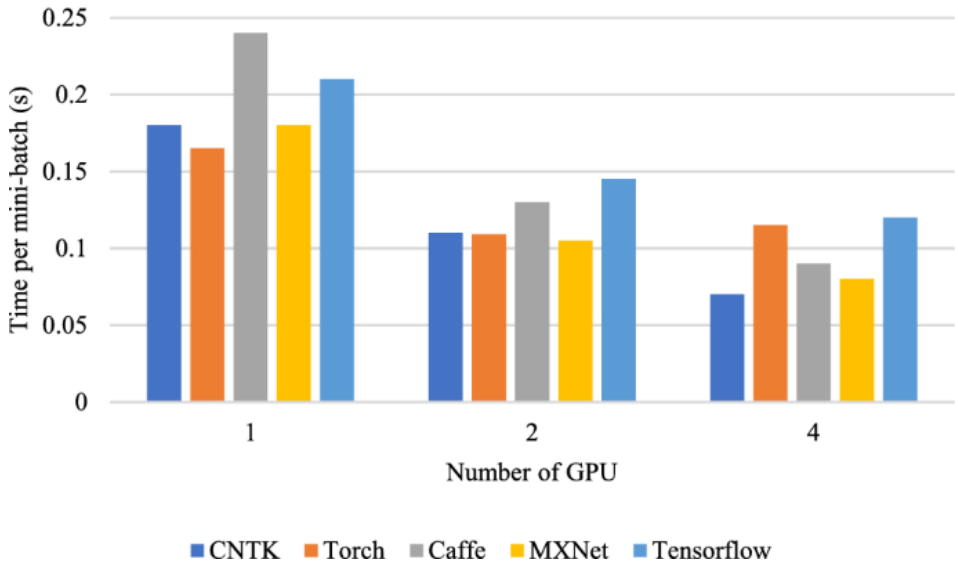
Herramientas

Rendimiento (enero de 2016)



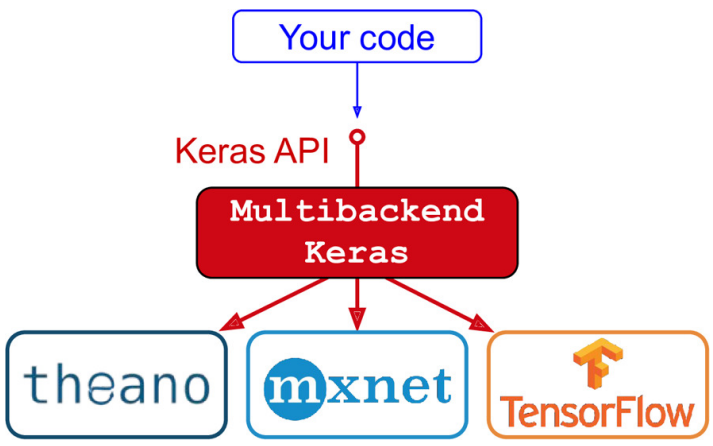
Herramientas

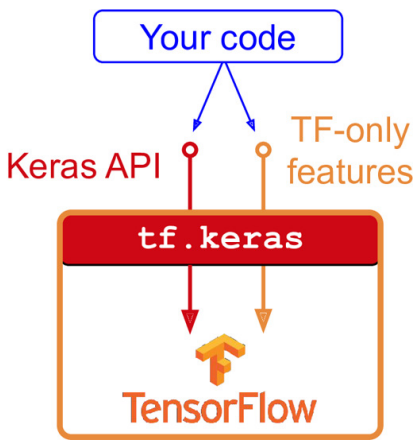
Rendimiento (febrero de 2019)



Herramientas

Keras
API multibackend vs. tf.keras

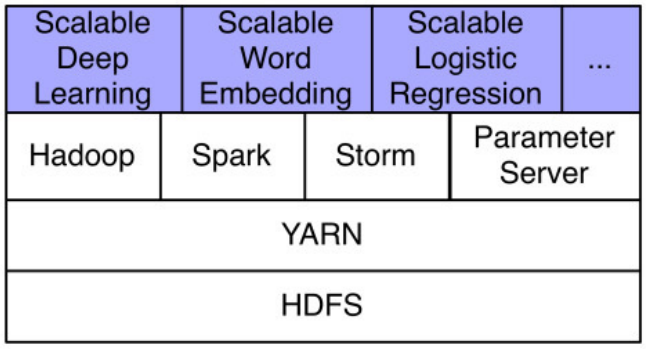


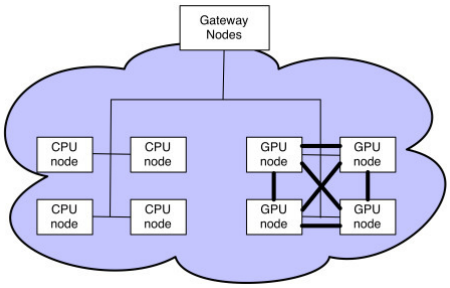


Herramientas

Yahoo Deep Learning Cluster

40 000 Hadoop nodes, 600 Petabytes, 100Gb InfiniBand





Hardware: NVIDIA Tesla GPUs

Software: Spark & Caffe

Inside Yahoo's Super-Sized Deep Learning Cluster, October 2015
http://www.datanami.com/2015/10/12/inside-yahoos-super-sized-deep-learning-cluster/

Herramientas

Servicios ofrecidos por empresas

- Ersatz Labs (cloud-based deep learning platform)
  http://www.ersatzlabs.com/
- Microsoft Azure Machine Learning
  https://azure.microsoft.com/en-us/services/machine-learning/
  CNTK Computational Network ToolKit
  https://github.com/Microsoft/CNTK
- Amazon Machine Learning
  https://aws.amazon.com/machine-learning/
  DSSTNE: Deep Scalable Sparse Tensor Network Engine
  https://github.com/amznlabs/amazon-dsstne
- IBM Watson
  https://www.ibm.com/smarterplanet/us/en/ibmwatson/
  IBM Deep Learning Framework (SystemML)
  http://systemml.apache.org/
  AlchemyAPI
  http://www.alchemyapi.com/

Herramientas

Más bibliotecas y frameworks:

- **PaddlePaddle**, de Baidu
  [PArallel Distributed Deep Learning] (C++ & Python)
  http://www.paddlepaddle.org/
- **VELES**, de Samsung
  (Python)
  https://velesnet.ml/ & https://github.com/samsung/veles
- **Brainstorm**, del IDSIA, Lugano, Suiza
  (Python)
  https://github.com/IDSIA/brainstorm
- **Marvin**, Princeton Computer Vision & Robotics Lab
  (C/C++)
  http://marvin.is/

Herramientas

Más bibliotecas y frameworks:

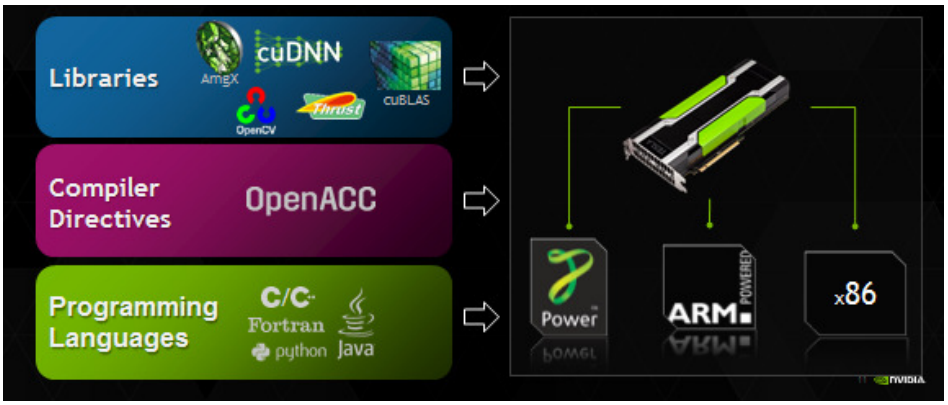
- **Neon**, de Nervana Systems ➔ Intel
  (Python)
  https://github.com/NervanaSystems/neon
- **Mocha**
  (Julia [http://julialang.org/], inspirado en Caffe [C++])
  https://github.com/pluskid/Mocha.jl
- **Flux**
  (Julia [http://julialang.org/])
  https://fluxml.ai/
- **Apache Singa**
  (C++ & Python)
  https://singa.incubator.apache.org/

Herramientas

Más bibliotecas y frameworks:

cuDNN

- **cuDNN**, de NVIDIA: GPU Accelerated Deep Learning (CUDA), usada por Caffe, Theano, TensorFlow & Torch
  https://developer.nvidia.com/cudnn

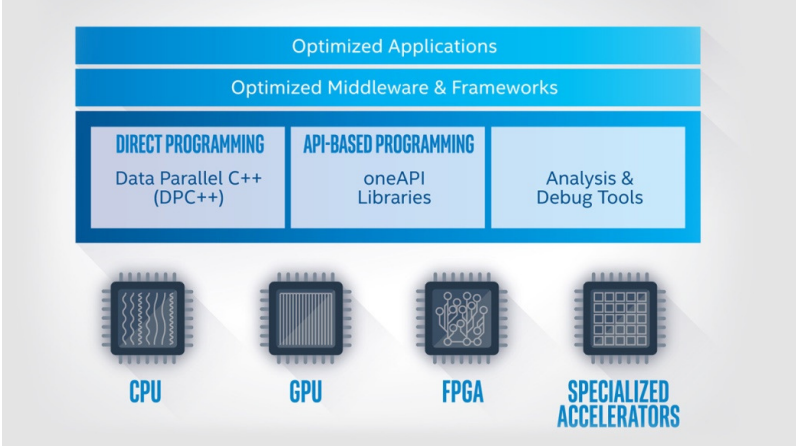


Herramientas

Más bibliotecas y frameworks:

oneAPI

- **oneAPI**, de Intel:
  Interfaz unificada para múltiples tipos de aceleradores
  https://software.intel.com/oneapi



Herramientas

Keras

fast.ai

TensorFlow

GLUON

Caffe

PyTorch

Caffe2

APACHE mxnet

Chainer

Microsoft CNTK

DL4J

∂y/net

theano