2024 CRA CONFERENCE

Deep Learning Hardware: Past, Present, and Future

Session Chair

Divesh Srivastava
AT&T

2024 CRA CONFERENCE

Bill Dally

Chief Scientist

NVIDIA

Deep Learning Hardware: Past, Present, and Future

CRA Snowbird Conference

July 23, 2024

Bill Dally

Chief Scientist and SVP of Research, NVIDIA Corporation

Adjunct Professor of CS and EE, Stanford

Q: What would Bill Dally say about deep learning hardware

Chat GPT: Bill Dally, a renowned computer scientist and the Chief Scientist at NVIDIA, has frequently spoken about deep learning hardware. Here are some key points he might emphasize:

1.Specialized Hardware: Dally often discusses the importance of specialized hardware for deep learning. He emphasizes that traditional CPUs are not well-suited for the massive parallel processing required by deep learning algorithms. Instead, GPUs (Graphics Processing Units) and other specialized processors like TPUs (Tensor Processing Units) are much more efficient.

2.Energy Efficiency: One of Dally's significant points is the need for energy-efficient hardware. Deep learning computations can be power-intensive, so designing hardware that maximizes performance per watt is crucial.

3.Memory Bandwidth: Dally would likely talk about the importance of memory bandwidth in deep learning hardware. The ability to quickly move large amounts of data between memory and processors is a critical factor in the performance of deep learning systems.

4.Scalability: Another key point is the scalability of hardware. Dally would emphasize the need for hardware that can scale to support larger models and datasets, which is essential for the continued advancement of deep learning.

5.Programmability: Dally might also highlight the importance of programmability in deep learning hardware. Having flexible and programmable hardware allows researchers and developers to experiment with new models and techniques more easily.

6.Co-design of Hardware and Algorithms: He often advocates for the co-design of hardware and algorithms. By designing algorithms with hardware capabilities in mind, and vice versa, significant performance gains can be achieved.

Dally's insights are grounded in his extensive experience in both academia and industry, making his perspectives highly influential in the field of deep learning hardware.

Gemini: Bill Dally on Deep Learning Hardware: A Focus on Efficiency and Flexibility

Bill Dally, a renowned expert in computer architecture and the Chief Scientist at NVIDIA, has consistently emphasized the importance of **efficiency** and **flexibility** in deep learning hardware.
Here are some key points he would likely highlight:

Efficiency is Paramount

- **Maximize compute density:** Dally would stress the need for hardware that can pack as much computational power into a given area as possible. This is crucial for energy efficiency and performance.
- **Minimize data movement:** He would point out that moving data around consumes significant energy and time. Hardware should be designed to reduce data transfer overhead.
- **Leverage specialized hardware:** While general-purpose processors are versatile, specialized hardware like GPUs and TPUs offer superior performance for specific workloads like deep learning.

Flexibility is Essential

- **Programmability:** Dally would advocate for hardware that can be easily programmed to handle different deep learning models and algorithms. This adaptability is crucial for staying ahead of rapidly evolving research.
- **Hybrid architectures:** He would likely discuss the benefits of combining specialized hardware with general-purpose processors to create hybrid systems that offer both high performance and flexibility.

Specific Examples of Dally's Insights

- **GPU architecture:** Dally has been instrumental in developing NVIDIA's GPU architecture, which has become the de facto standard for deep learning acceleration. He would emphasize the importance of parallel processing,memory bandwidth, and efficient data movement in GPU design.
- **Efficient data centers:** Dally has also explored ways to optimize data center infrastructure for deep learning workloads. This includes efficient power delivery, cooling systems, and network connectivity.

In essence, Bill Dally's perspective on deep learning hardware is centered on creating systems that deliver exceptional performance while maintaining the ability to adapt to new challenges and opportunities.

3. **Beam Theory:**

   The maximum bending moment M_{max} for a center-loaded simply supported beam is:

$$M_{max} = \frac{PL}{4}$$

4. **Maximum Stress:**

   The maximum bending stress σ_{max} is:

$$\sigma_{max} = \frac{M_{max} \cdot c}{I}$$

   where $c = \frac{h}{2}$.

5. **Load Capacity:**

   Set σ_{max} equal to the yield strength:

$$\sigma_y = \frac{M_{max} \cdot c}{I}$$

   Solving for P:

$$P = \frac{4 \cdot \sigma_y \cdot I}{L \cdot c}$$

   Since $c = \frac{h}{2}$:

$$P = \frac{4 \cdot \sigma_y \cdot I}{L \cdot \frac{h}{2}} = \frac{8 \cdot \sigma_y \cdot I}{L \cdot h}$$

Example Calculation:

Let's assume typical dimensions for a 6-inch aluminum I-beam:

- Height $h = 6$ inches
- Flange width $b = 3$ inches
- Flange thickness $t_f = 0.5$ inches
- Web thickness $t_w = 0.25$ inches
- Span $L = 10$ feet (120 inches)

Using standard tables for I-beams or calculating:

- Moment of inertia I (approximate value for example): 9 in^4

Yield strength of aluminum (6061-T6):

- $\sigma_y = 35,000$ psi

Plug these values into the formula:

$$P = \frac{8 \cdot 35,000 \cdot 9}{120 \cdot 6}$$

$$P = \frac{2,520,000}{720}$$

$$P \approx 3,500 \text{ pounds}$$

THE WALL STREET JOURNAL.
English Edition ▾ | Print Edition | Video | Audio | Latest Headlines | More ▾

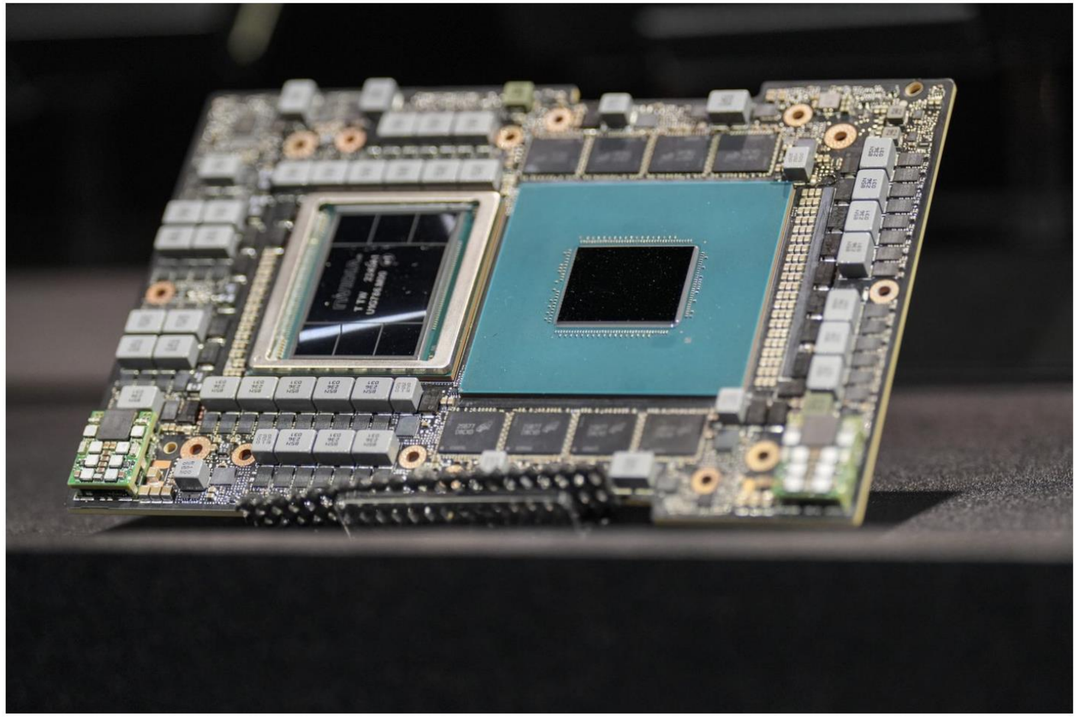
CIO JOURNAL

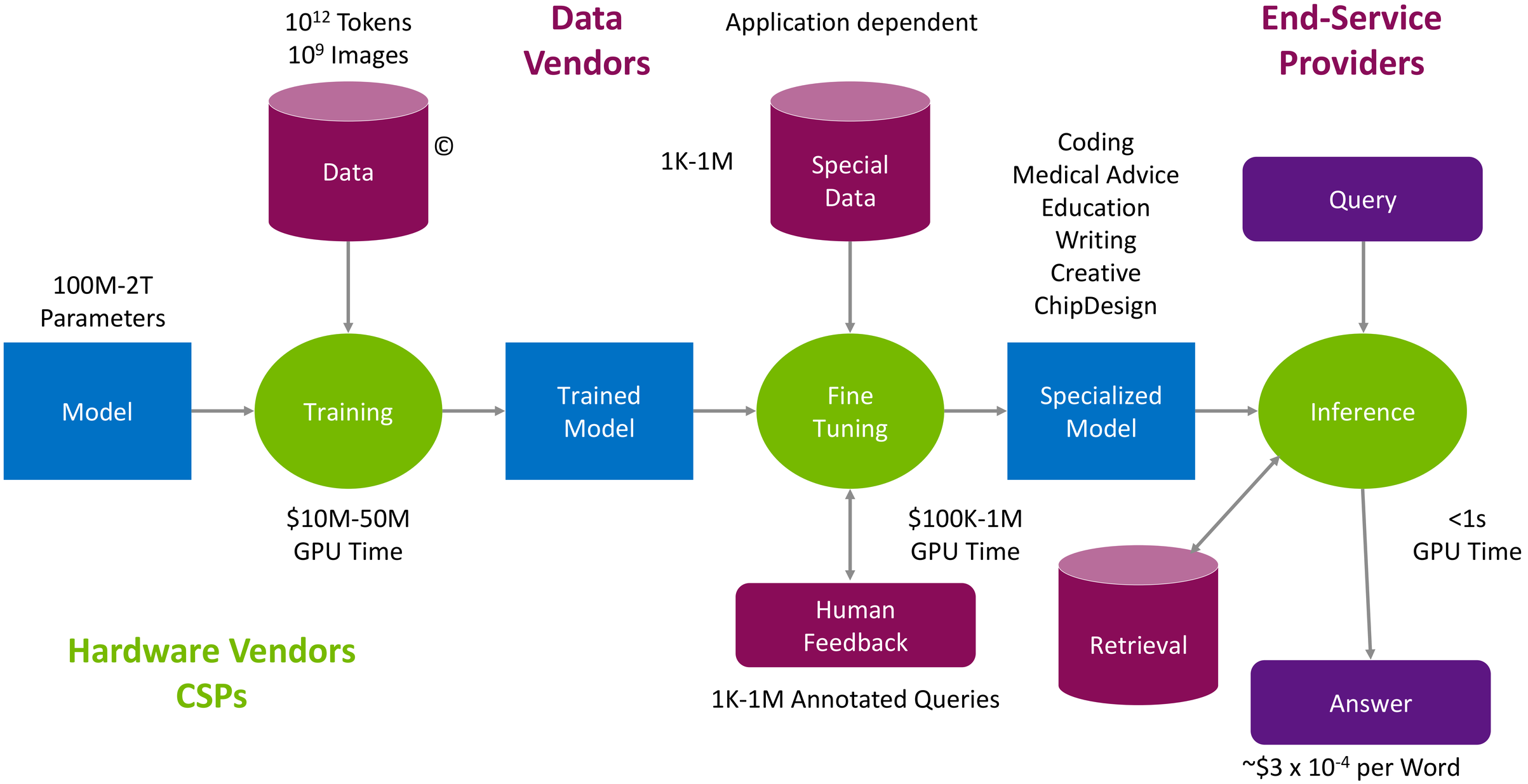
Pro Take: Going Beyond Moore's Law; Semiconductor Innovation Continues, But It Is Tougher

'It's getting harder, but we still have good ideas,' says Bill Dally, chief scientist at semiconductor powerhouse Nvidia, known for its AI chips

By *Steven Rosenbush* Follow

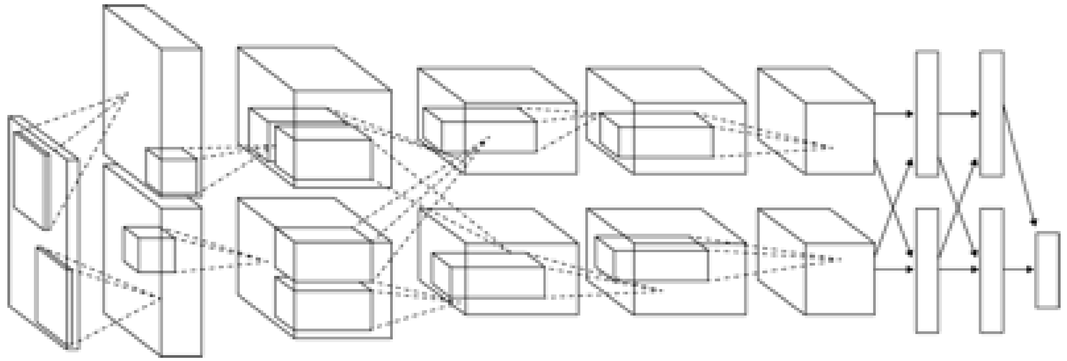
Nov. 8, 2023 7:00 am ET

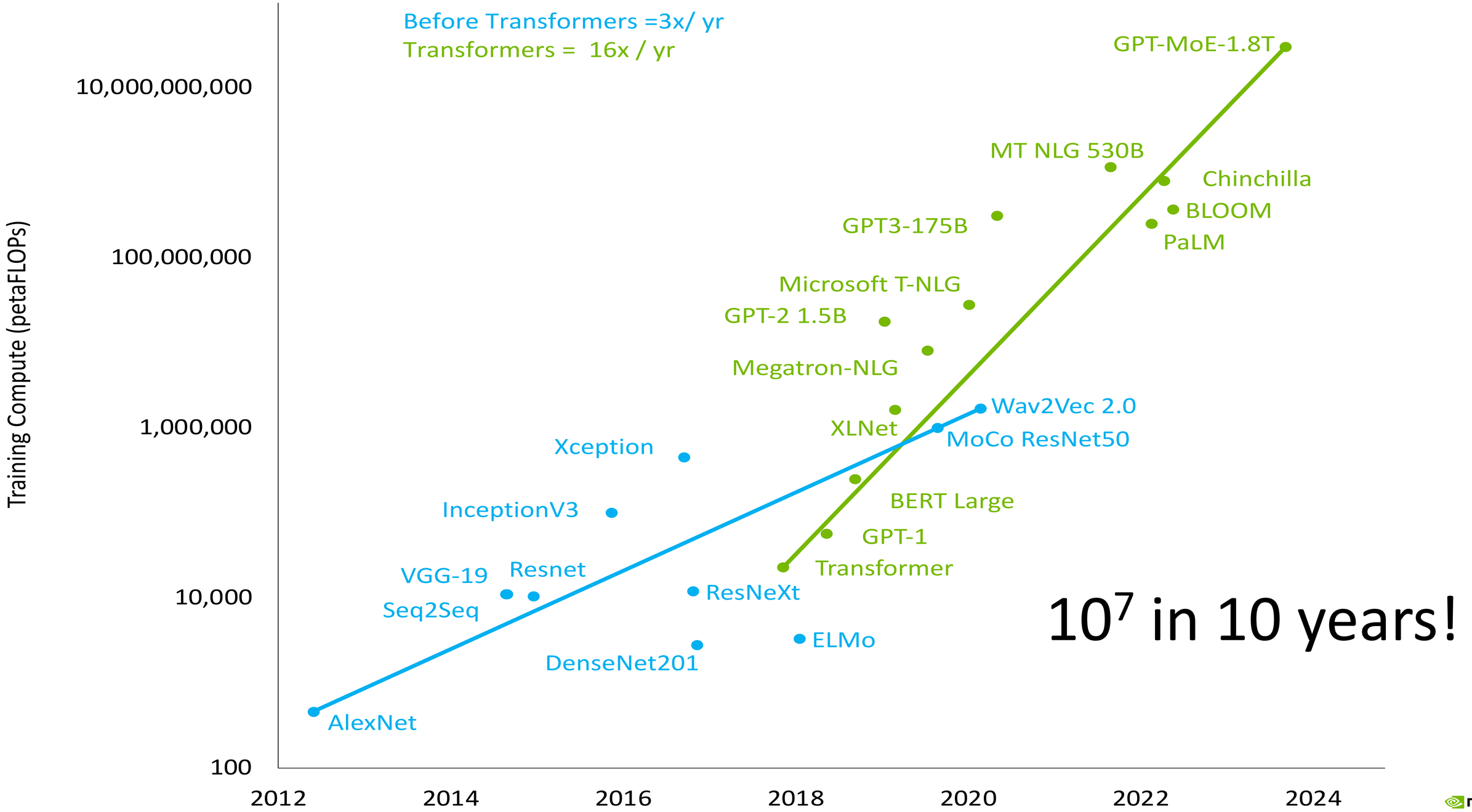



10^{12} Tokens
10^{9} Images
Data Vendors
Application dependent
End-Service Providers
Data
©
1K-1M
Special Data
Coding
Medical Advice
Education
Writing
Creative
ChipDesign
Query
100M-2T Parameters
Model
Training
Trained Model
Fine Tuning
Specialized Model
Inference
$10M-50M GPU Time
$100K-1M GPU Time
<1s GPU Time
Human Feedback
Retrieval
Answer
1K-1M Annotated Queries
~$3 x 10^{-4} per Word
Hardware Vendors CSPs

Motivation

Deep Learning was Enabled by Hardware




Before Transformers =3x/ yr
Transformers = 16x / yr
Training Compute (petaFLOPs)
10,000,000,000
100,000,000
1,000,000
10,000
100
2012
2014
2016
2018
2020
2022
2024
GPT-MoE-1.8T
MT NLG 530B
Chinchilla
BLOOM
PaLM
GPT3-175B
Microsoft T-NLG
GPT-2 1.5B
Megatron-NLG
XLNet
Wav2Vec 2.0
MoCo ResNet50
Xception
BERT Large
InceptionV3
GPT-1
Transformer
VGG-19
Resnet
ResNeXt
Seq2Seq
ELMo
DenseNet201
AlexNet
10^7 in 10 years!

Some History

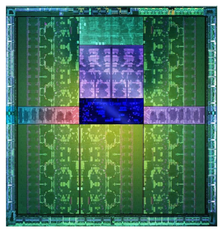
Single-Chip Inference Performance - 1000X in 10 years

B200
20,000

Int 8 TOPS

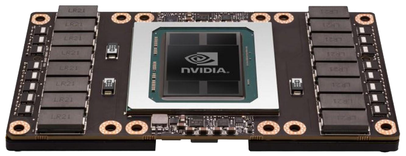
4500.00
4000.00
3500.00
3000.00
2500.00
2000.00
1500.00
1000.00
500.00
0.00

4/1/12
8/14/13
12/27/14
5/10/16
9/22/17
2/4/19
6/18/20
10/31/21
3/15/23



FP32
FMA

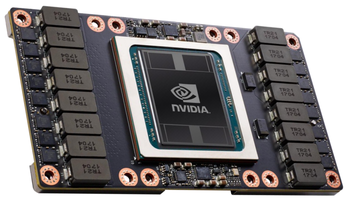
K20X
3.94



FP16
HDP4

M40
6.84

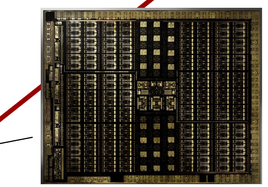
P100
21.20



FP16
HMMA

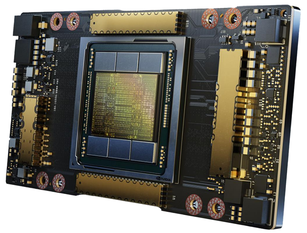
V100
125.00

Q8000
261.00

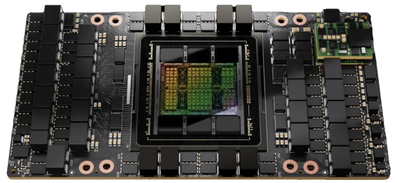


INT8
IMMA

A100
1248.00



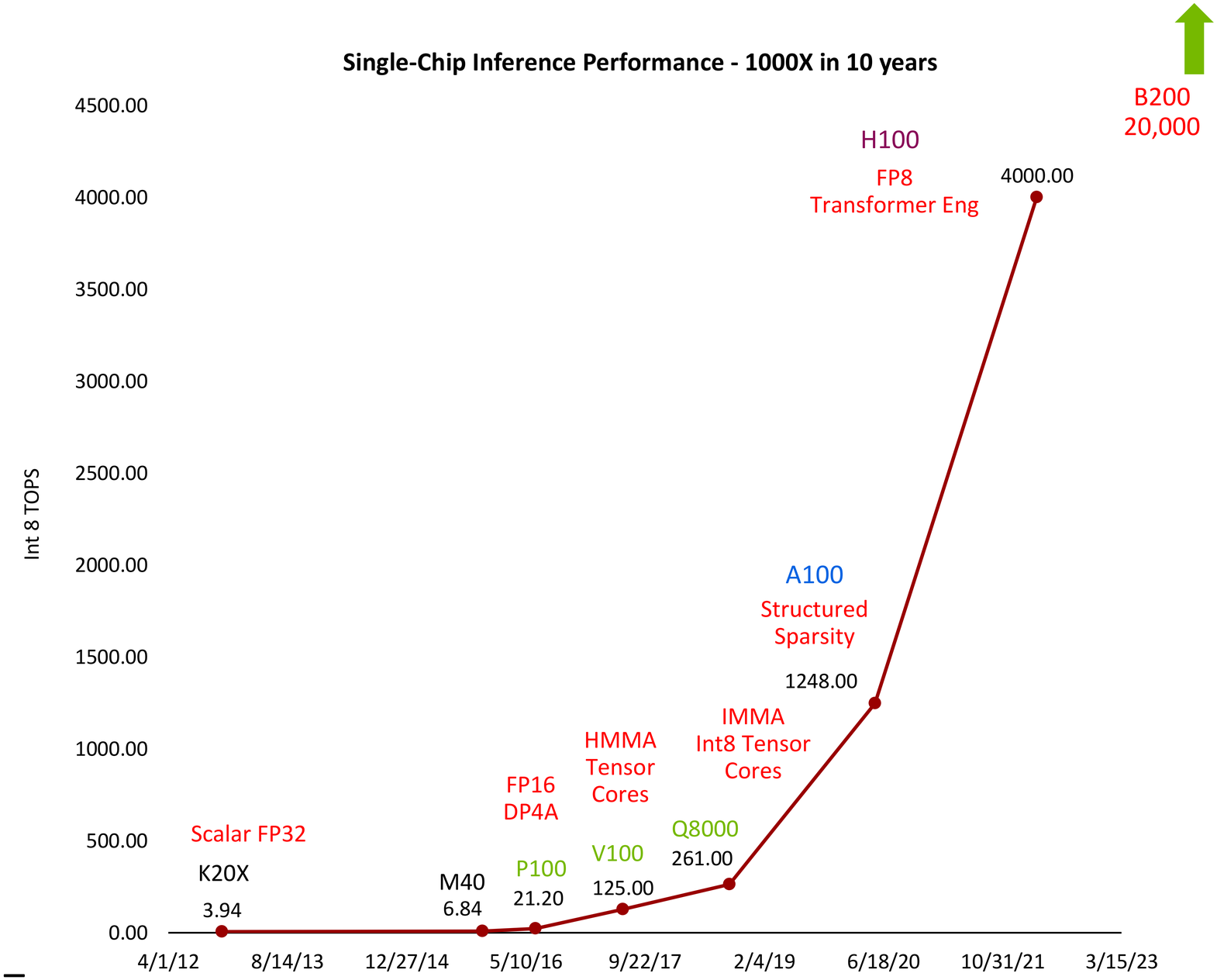
Sparsity



H100
4000.00

Gains from

- Number Representation
  - FP32, FP16, Int8, FP4
  - (TF32, BF16)
  - ~16x, 32x
- Complex Instructions
  - DP4, HMMA, IMMA
  - ~12.5x
- Process
  - 28nm, 16nm, 7nm, 5nm, 4nm
  - ~2.5x, 3x
- Sparsity ~2x
- Die Size 2x
- Model efficiency has also improved – overall gain > 1000x



Specialized Instructions Amortize Overhead

Operation	Energy**	Overhead*
HFMA	1.5pJ	2000%
HDP4A	6.0pJ	500%
HMMA	110pJ	22%
IMMA	160pJ	16%

*Overhead is instruction fetch, decode, and operand fetch – 30pJ
**Energy numbers from 45nm process

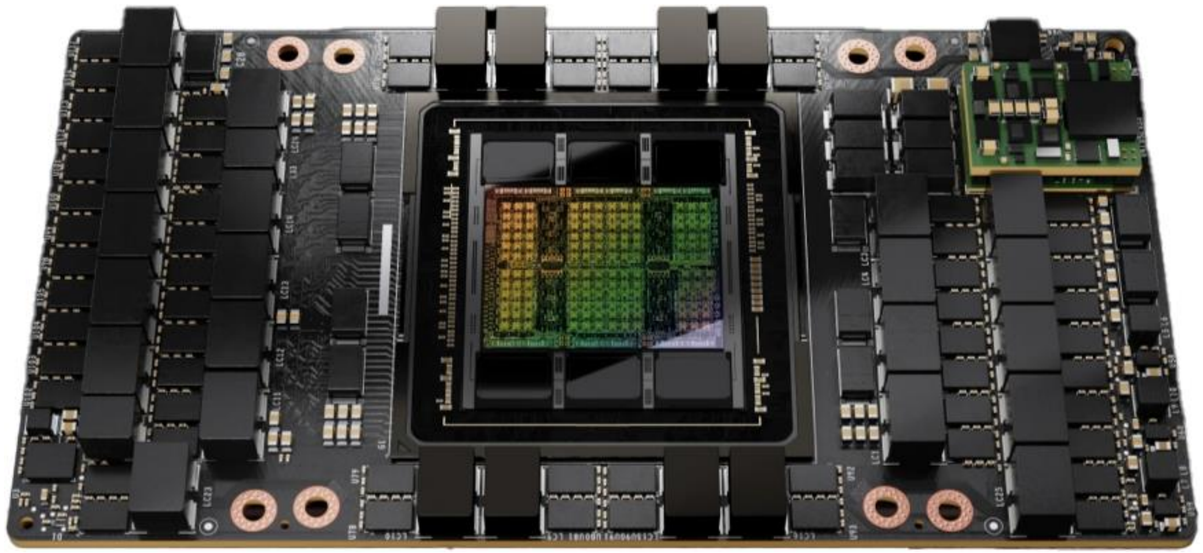
Hopper H100

1 PFLOPS (TF32)
1 / 2 PLFLOPS (FP16 or BF16) (dense/sparse)
2 / 4 PLFOPS (FP8 or Int8) (dense/sparse)

3.4TB/s (HBM3) 94GB
18 NVLINK ports
400Gb/s each 900GB/s total
700W

Transformer Engine
Dynamic Programming Instructions

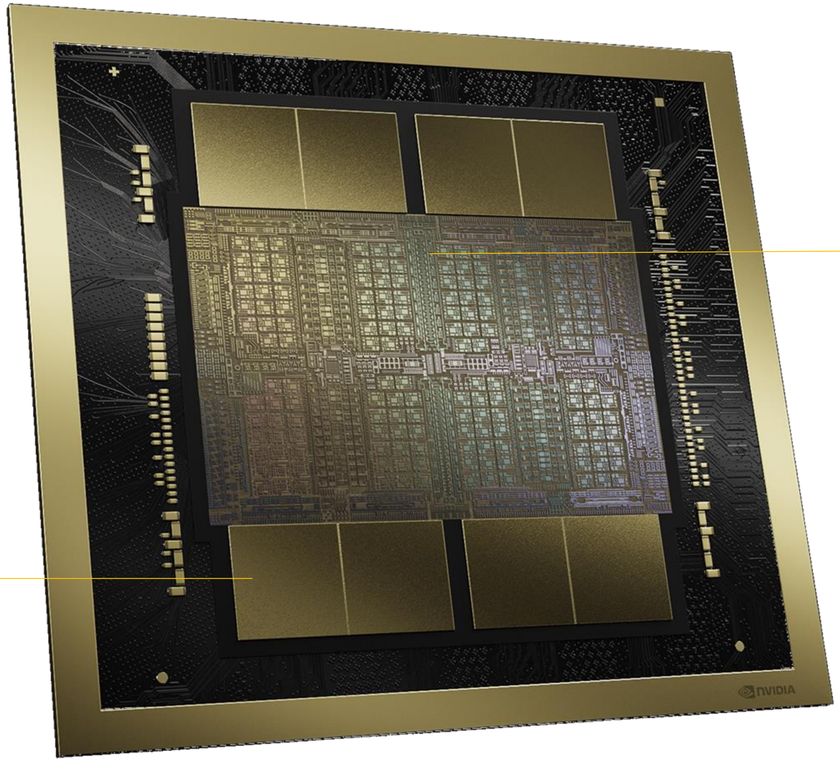
9 TOPS/W (Int8/FP8)



4PF Sparse FP8, 900GB/s, 700W

Blackwell B200

The Two Largest Dies Possible—Unified as One GPU



Fast Memory
192GB HBM3e

2 reticle-limited dies operate as One Unified CUDA GPU

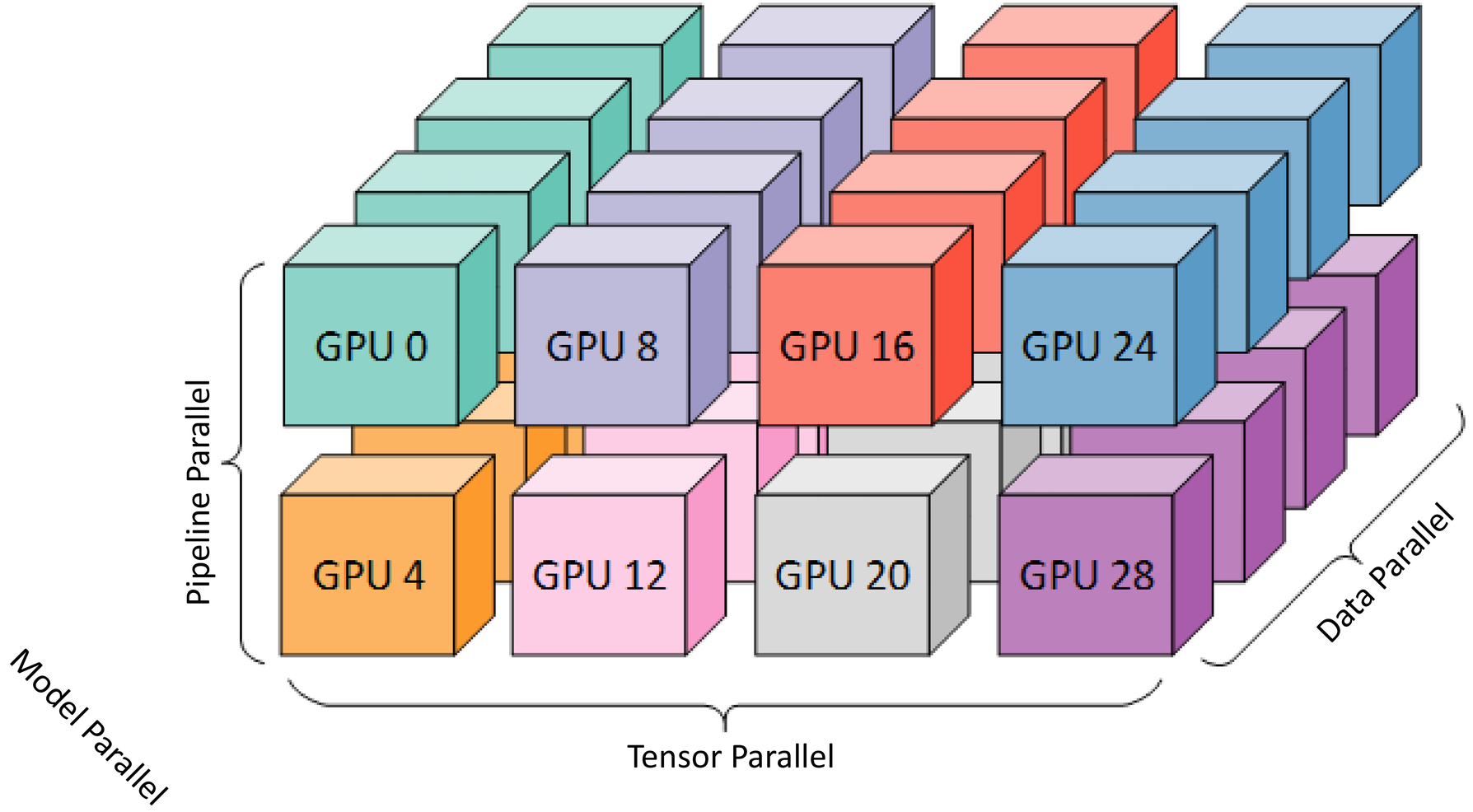
NV-HBI 10TB/s High Bandwidth Interface

Full performance. No compromises

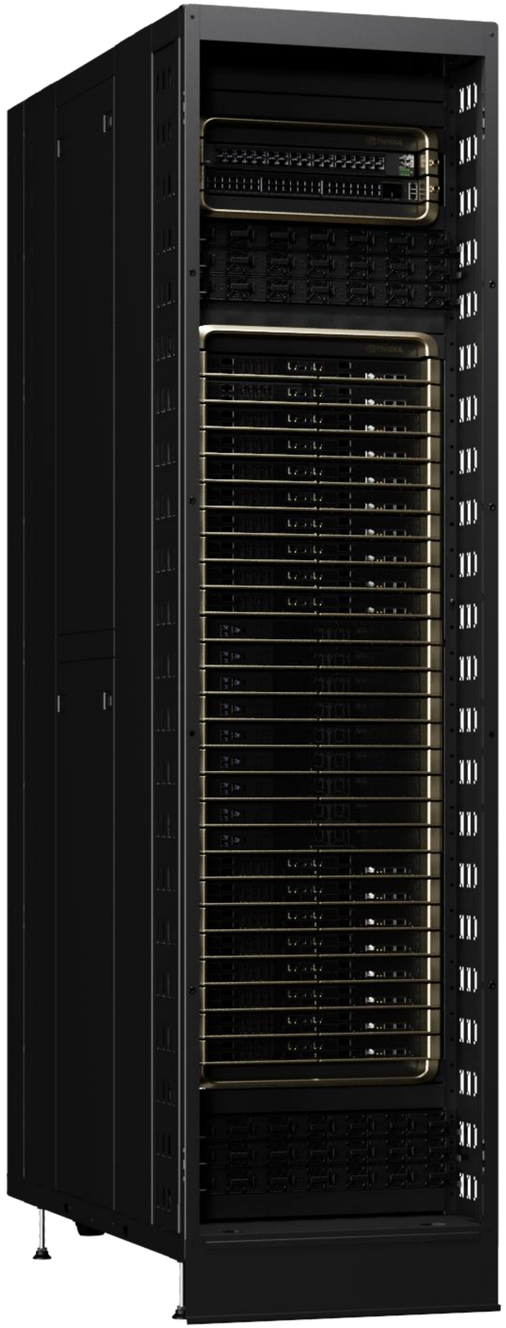
4X Training | 30X Inference | 25X Energy Efficiency & TCO

10 PetaFLOPS FP8 | 20 PetaFLOPS FP4
192GB HBM3e | 8 TB/sec HBM Bandwidth | 1.8TB/s NVLink

3D Parallelism



It takes 20 GPUs to hold one copy of GPT4 model parameters



GB200 NVL72

Delivers New Unit of Compute

GB200 NVL72	36 GRACE CPUs
	72 BLACKWELL GPUs
	Fully Connected NVLink Switch Rack

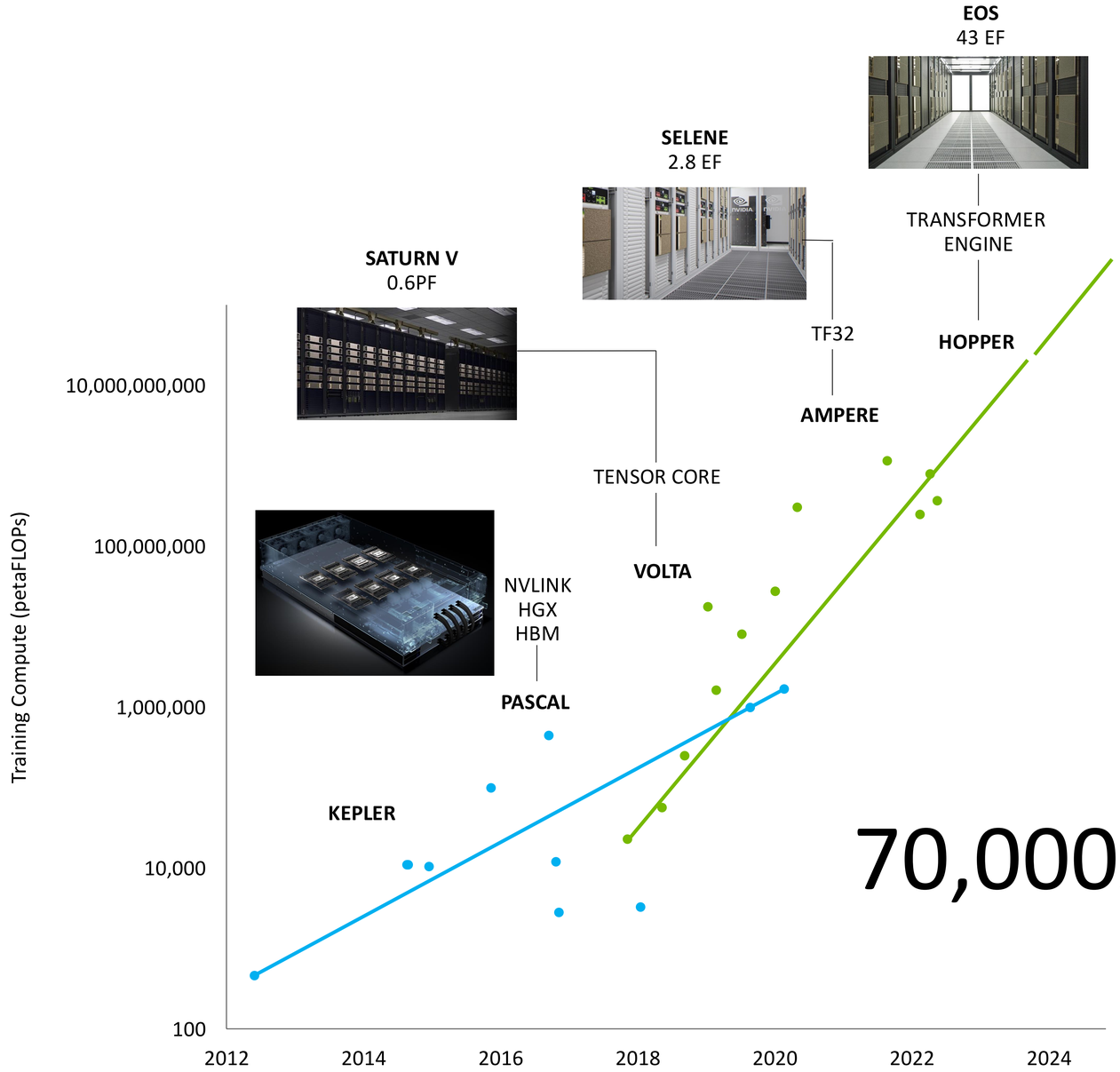
Training	720 PFLOPs
Inference	1.4 EFLOPs
NVL Model Size	27T params
Multi-Node All-to-All	130 TB/s
Multi-Node All-Reduce	260 TB/s

Scale-up – NVLink and NVSwitch – to 256 GPUs

Scale-out – IB to 10,000s of GPUs

Collectives Double Effective Network Bandwidth (AllReduce)

System Scaling

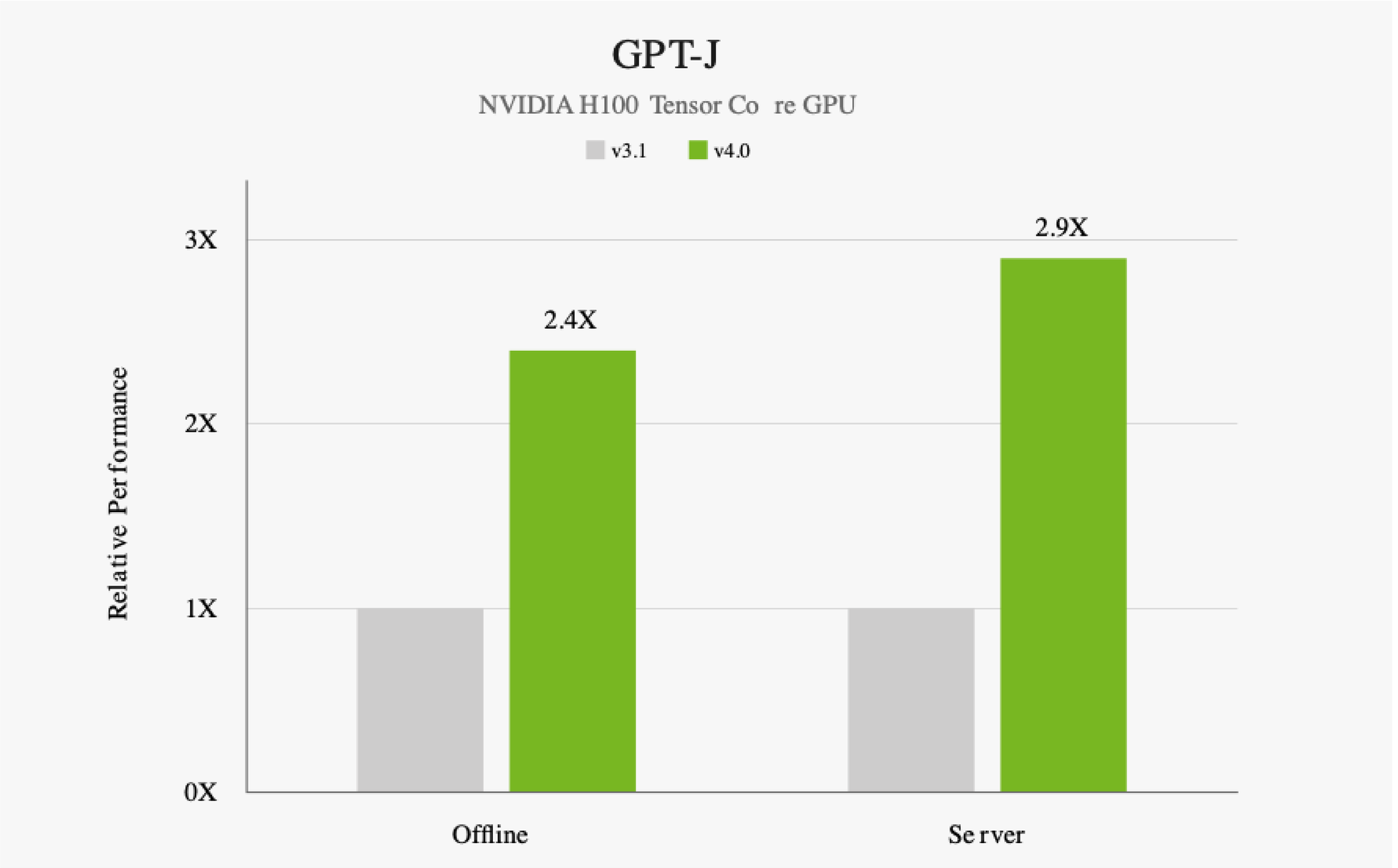


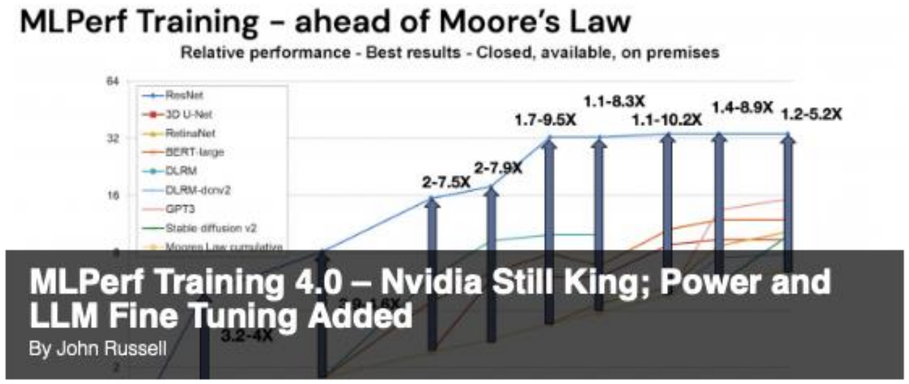
70,000x in 5 years

Software


MODULUS
CLARA MONAI
RIVA
MAXINE
NEMO
MERLIN
AVATAR
DRIVE
ISAAC
METROPOLIS
HOLOSCAN
APPLICATION FRAMEWORKS
PLATFORM
NVIDIA HPC
NVIDIA AI
NVIDIA OMNIVERSE
SYSTEM SOFTWARE
RTX
CUDA-X
PHYSX
UCF
DOCA
MAG
BASE CMD
FLEET CMD
AERIAL
HARDWARE
RTX
DGX
HGX
EGX
OVX
SUPER POD
AGX
GPU
CPU
DPU
NIC
SWITCH
SOC

2.4-2.9x From Software Improvements





June 12, 2024

There are really two stories packaged in the most recent MLPerf Training 4.0 results, released today. The first, of course, is the results. Nvidia (currently king of accelerated computing) wins again, sweeping all nine "events" (workflows) as it were. Its lead remains formidable. Story number two,

March 28, 2024

HPC wire
Since 1987 - Covering the Fastest Computers in the World and the People Who Run Them
Home
Topics
Sectors
Exascale
Specials
Resource Library

NVIDIA H100 GPUs Set Standard for Generative AI in Debut MLPerf Benchmark

June 28, 2023

June 28, 2023 — Leading users and industry-standard benchmarks agree: NVIDIA H100 Tensor Core GPUs deliver the best AI performance, especially on the large language models (LLMs) powering generative AI.

H100 GPUs set new records on all eight tests in the latest MLPerf training benchmarks released this week, excelling on a new MLPerf test for generative AI. That excellence is delivered both per-accelerator and at-scale in massive servers.

MLPerf

Home
Topics
Sectors
Exascale
Specials
Resource Library
Podcast

September 13, 2023

MLCommons this week issued the results of its latest MLPerf Inference (v3.1) benchmark exercise. Nvidia was again the top performing accelerator, but Intel (Xeon CPU) and Habana (Gaudi1 and 2) performed well. Google provided a peak at its new TPU (v5e) performance. ML Commons also

HPC wire
Since 1987 - Covering the Fastest Computers in the World and the People Who Run Them
Home
Topics
Sectors
Exascale
Specials

New MLPerf Training and HPC Benchmark Results Showcase 49X Performance Gains in 5 Years

November 8, 2023

SAN FRANCISCO, Nov. 8, 2023 — Today, MLCommons announced new results from two industry-standard MLPerf benchmark suites:

- The **MLPerf Training v3.1 suite**, which measures the

April 5, 2023

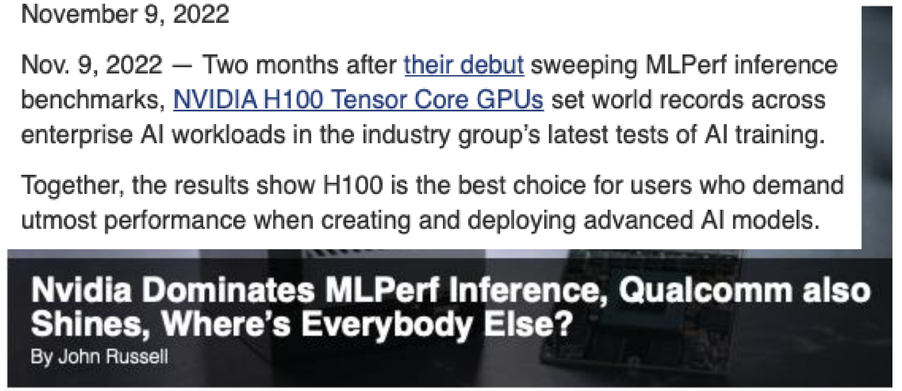
HPC wire
Since 1987 - Covering the Fastest Computers in the World and the People Who Run Them
Home
Topics
Sectors
Technologies
Sectors
AI/ML/DL
Exascale

Nvidia Hopper, Ampere GPUs Sweep MLPerf Benchmarks in AI Training

November 9, 2022

Nov. 9, 2022 — Two months after their debut sweeping MLPerf inference benchmarks, NVIDIA H100 Tensor Core GPUs set world records across enterprise AI workloads in the industry group's latest tests of AI training.

Together, the results show H100 is the best choice for users who demand utmost performance when creating and deploying advanced AI models.



April 6, 2022

Future Directions

Future Directions

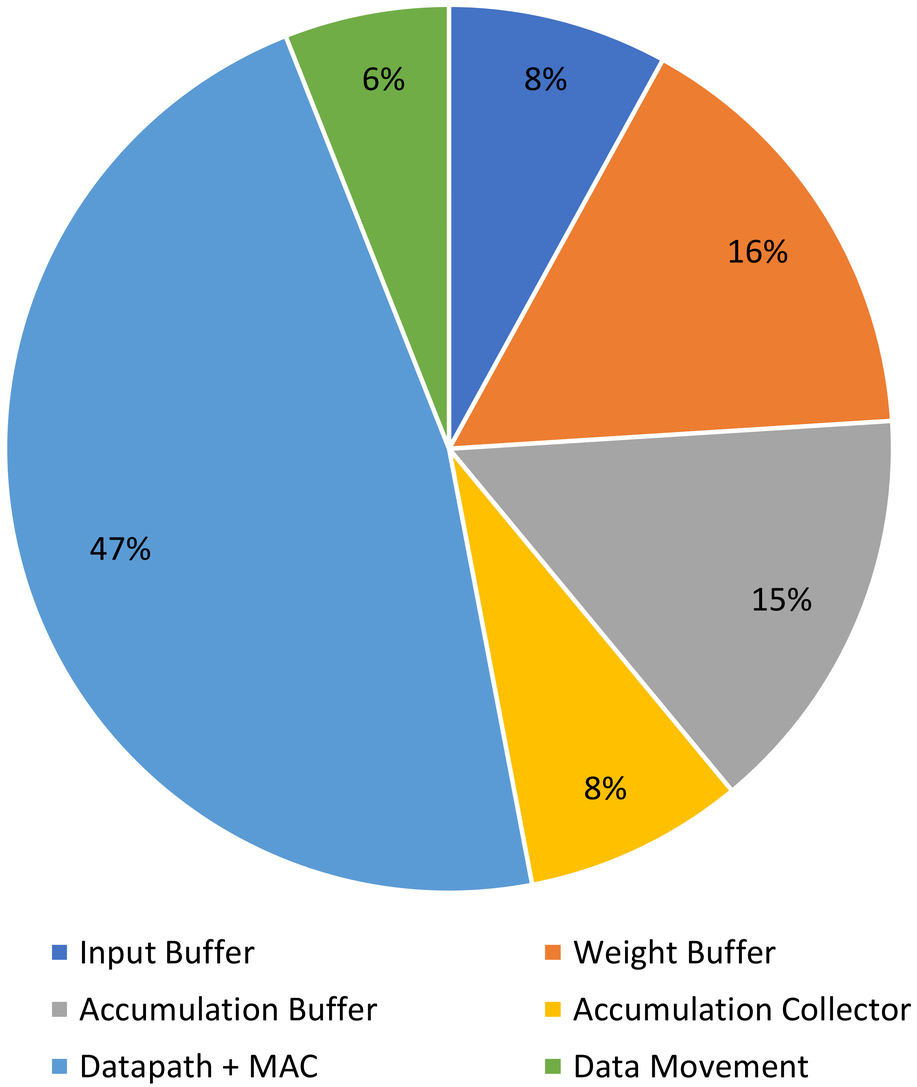
Number representation
- Log numbers
- Vector scaling (VS-Quant)
- Optimal Clipping
- Much cheaper math
- Smaller numbers

Sparsity
- Activations
- Lower density (vs 2:4 in A100/H100)

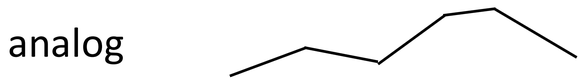
Better tiling
- Lower memory energy

Circuits
- Memory
- Communication
- 3D memory

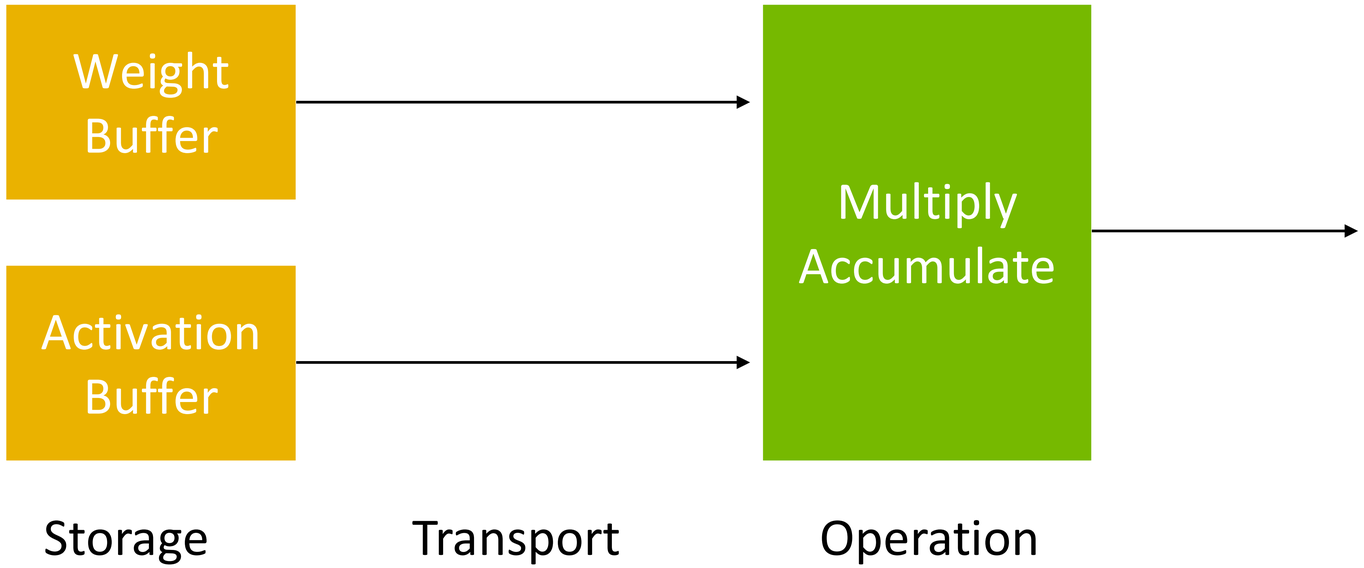
Process
- Capacitance scaling

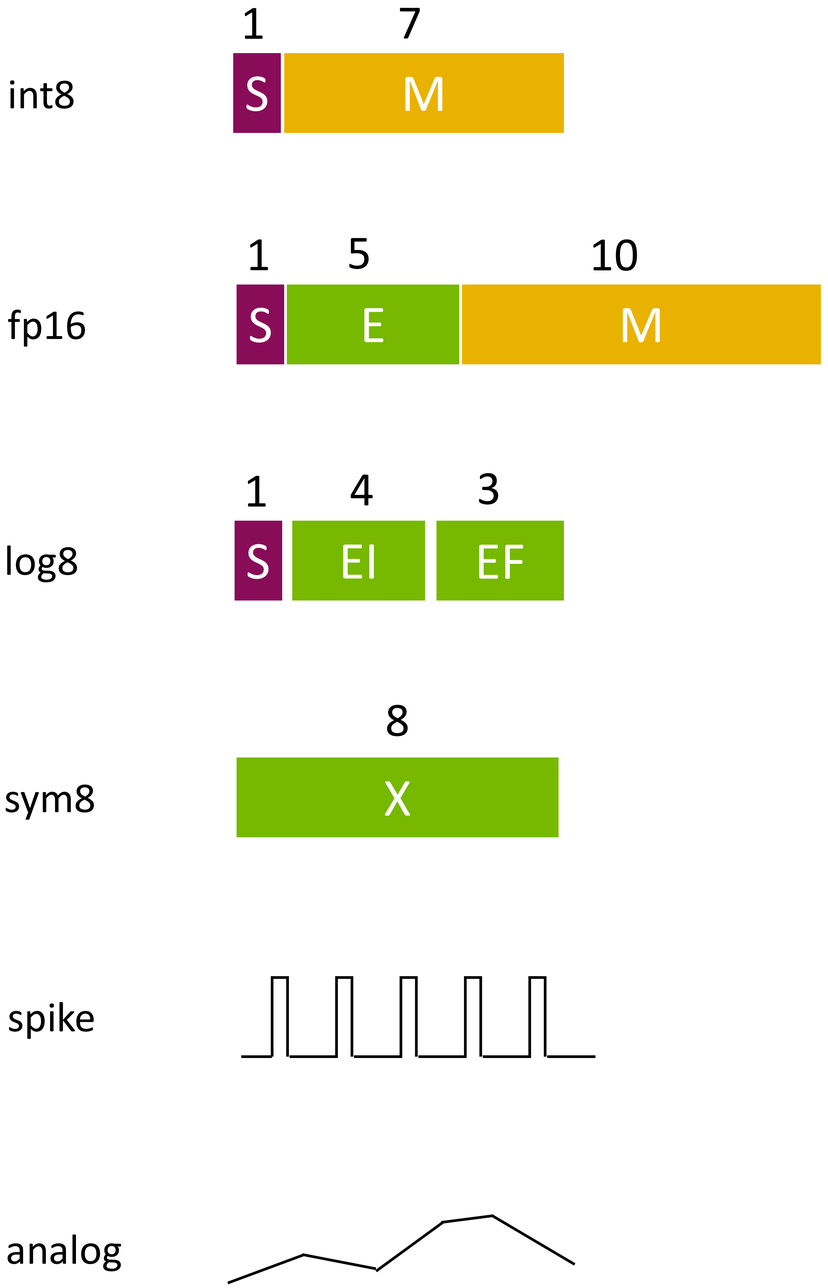


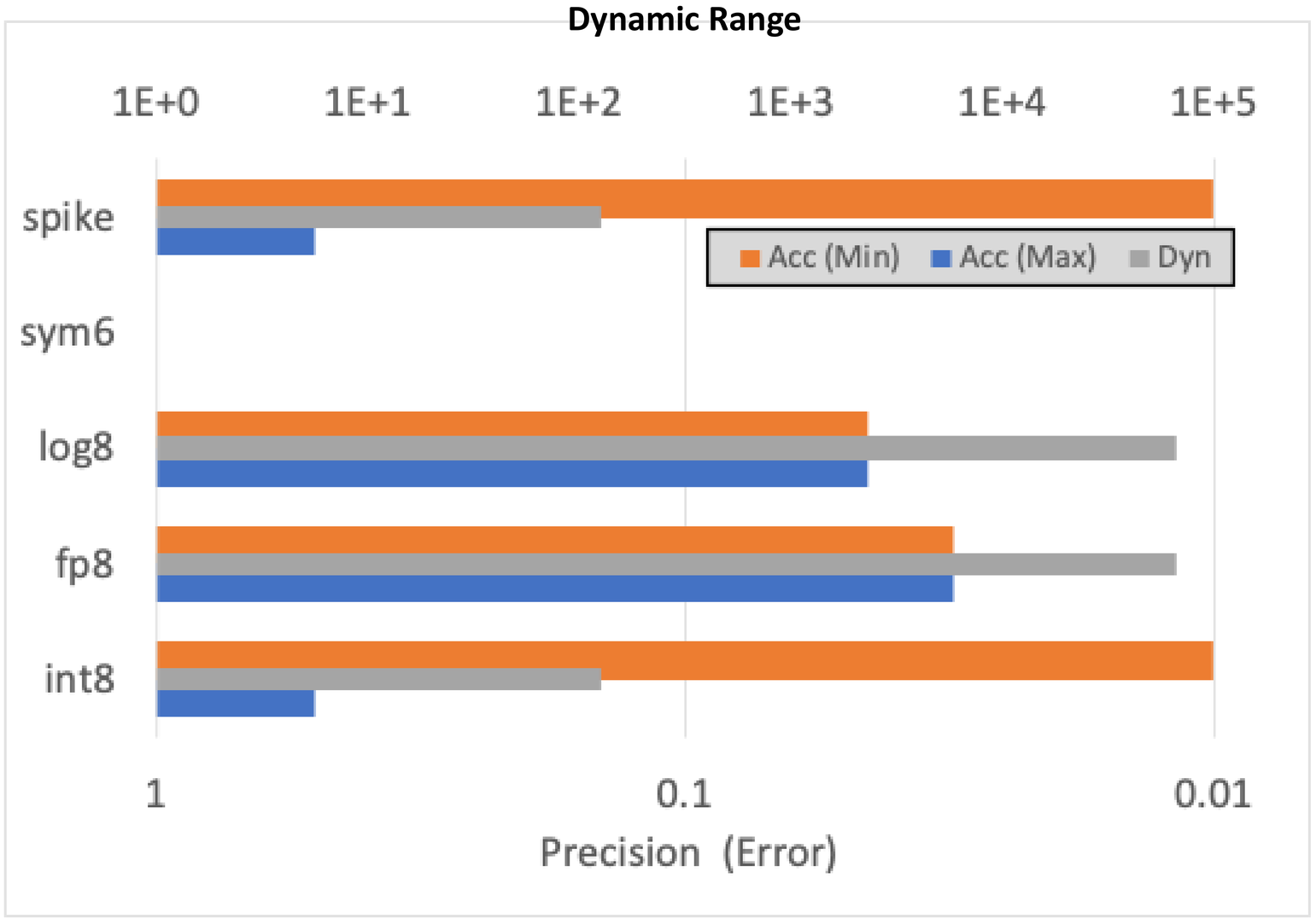
Number Representation



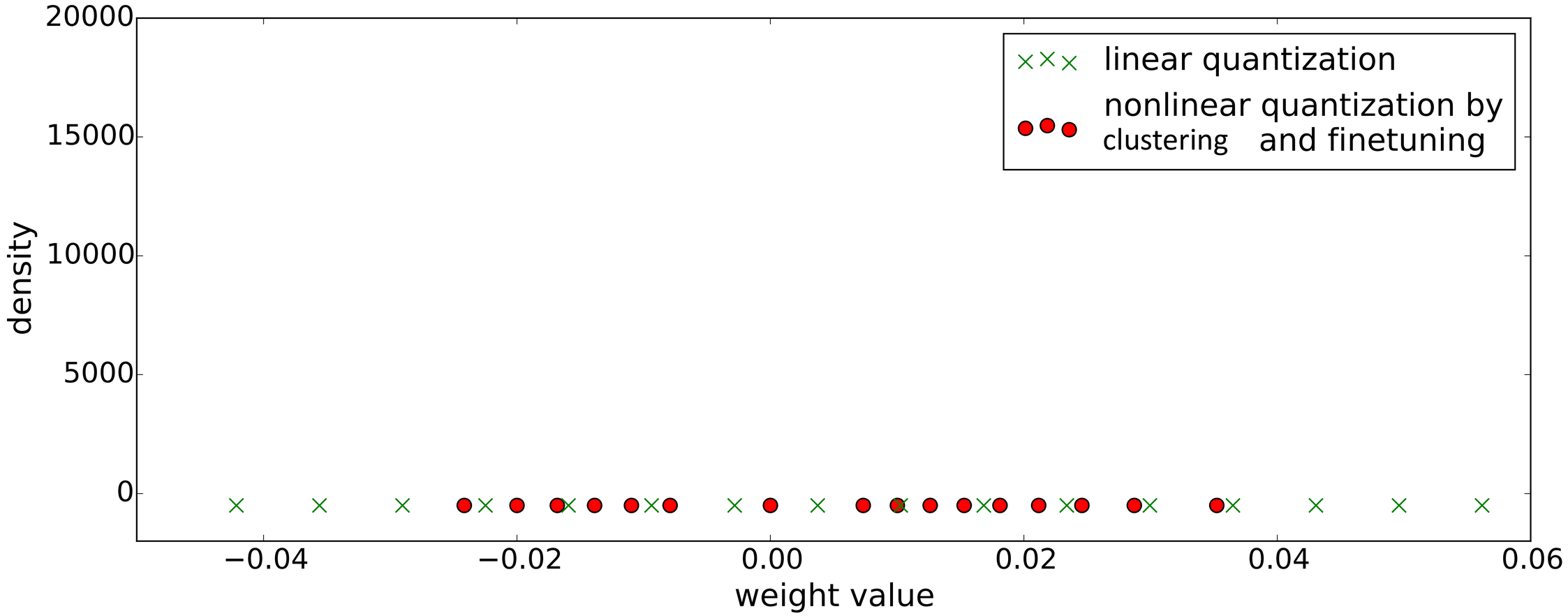
- Attributes:
  - Cost
    - Operation energy
    - Movement energy
  - Accuracy
    - Dynamic range
    - Precision (error)



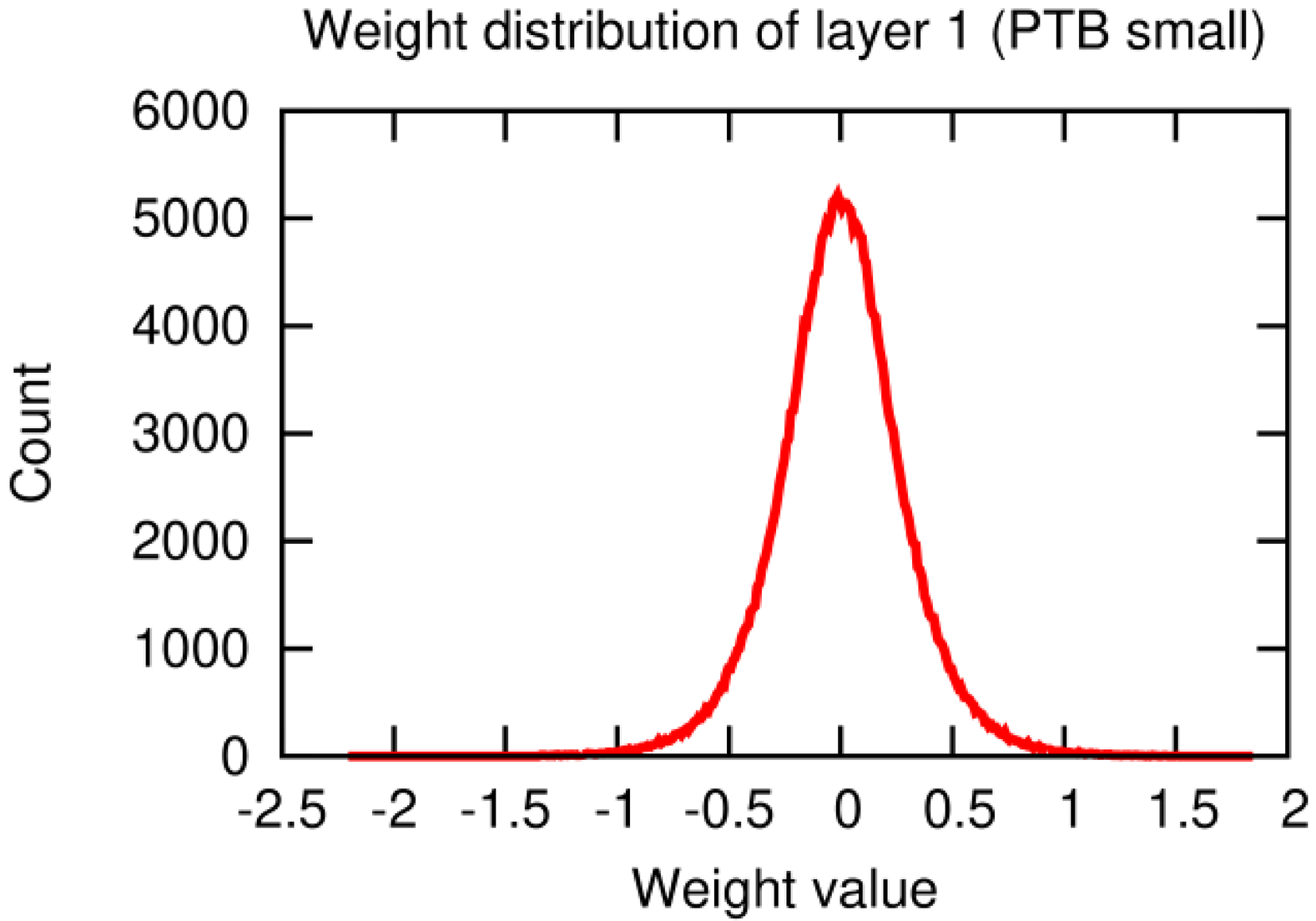

1
7
int8
S
M
1
5
10
fp16
S
E
M
1
4
3
log8
S
EI
EF
8
sym8
X
spike
analog


Dynamic Range
1E+0
1E+1
1E+2
1E+3
1E+4
1E+5
spike
sym6
log8
fp8
int8
Acc (Min)
Acc (Max)
Dyn
1
0.1
0.01
Precision (Error)

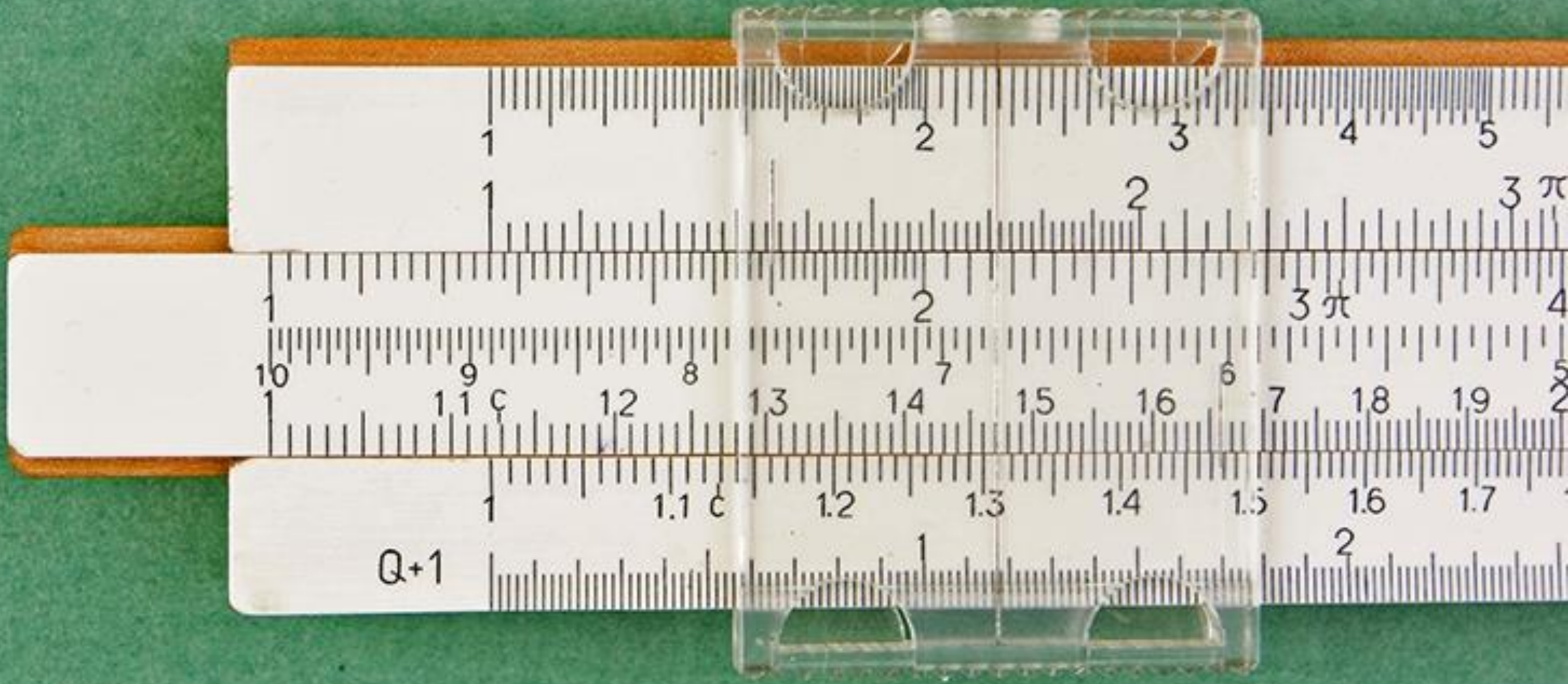
Symbol Representation (Codebook)



Han et al. Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding, arXiv 2015


Weight distribution of layer 1 (PTB small)
Count
6000
5000
4000
3000
2000
1000
0
-2.5
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
Weight value

Log Representation


1 2 3 4 5
1 2 3 π
1 2 3 π 4
10 9 8 7 6 5
1 1.1 C 1.2 1.3 1.4 1.5 1.6 1.7 2
1 1.1 C 1.2 1.3 1.4 1.5 1.6 1.7
Q+1
1 2

- Dynamic Range 10^5
- WC Accuracy 4%

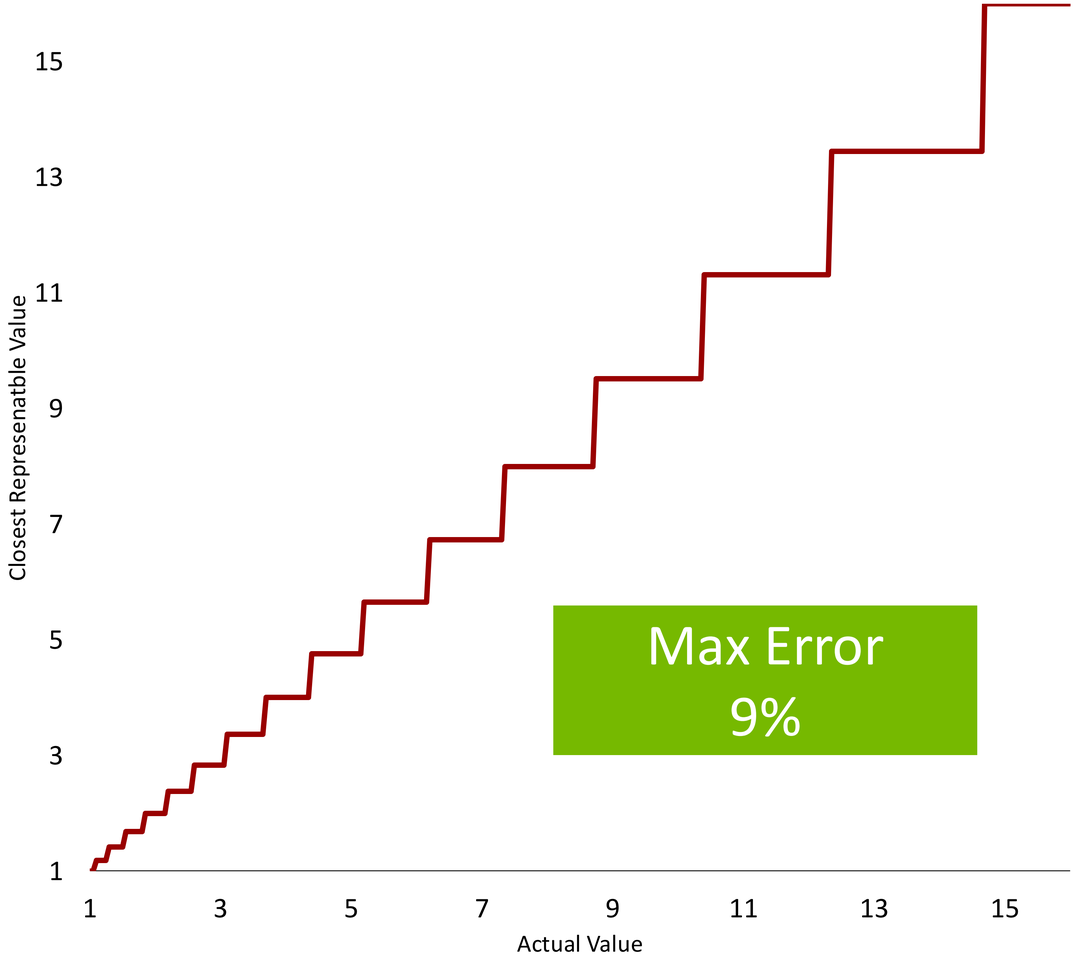
- Vs Int8 – DR 10^2
- WC Accuracy 33%

- Vs FP8 (E4M3) – DR 10^5
- WC Accuracy 6%

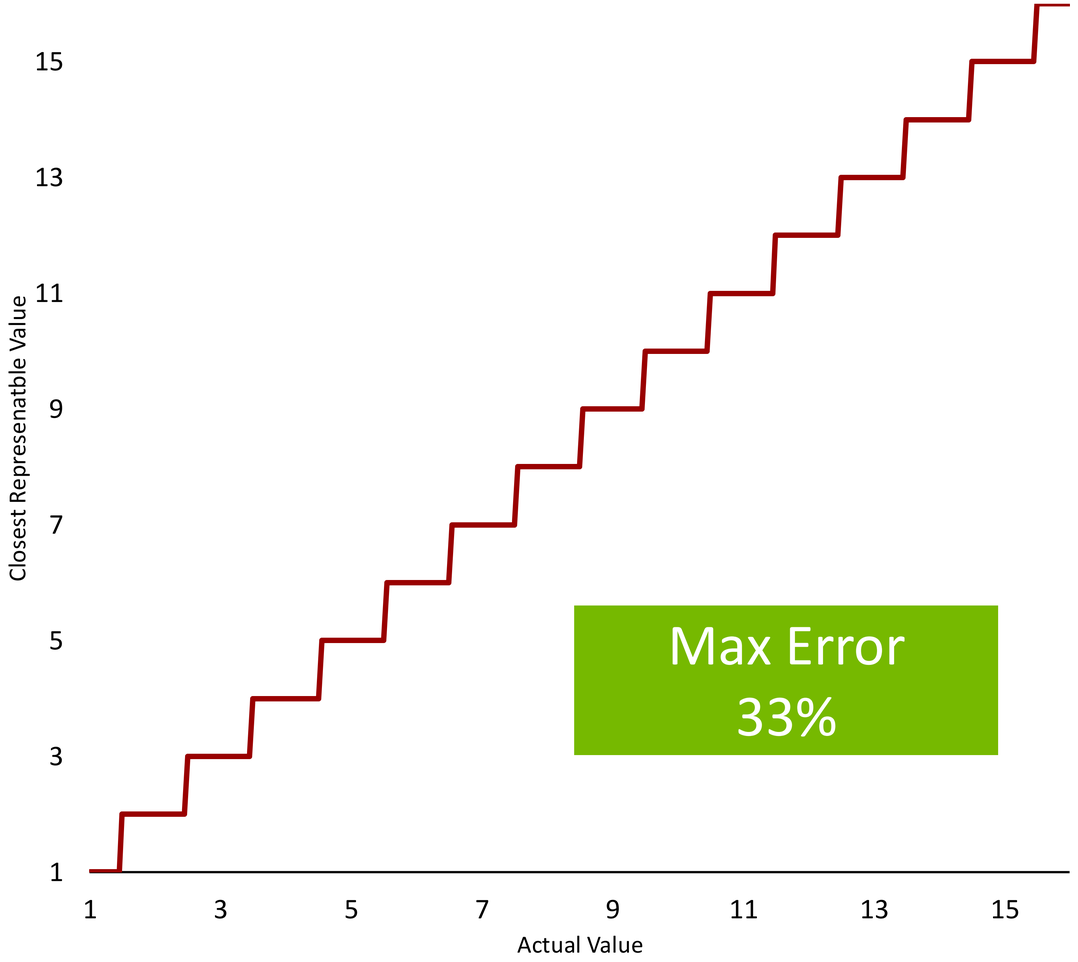
$$v = -1^{s}2^{ei.ef}$$

Log4.3

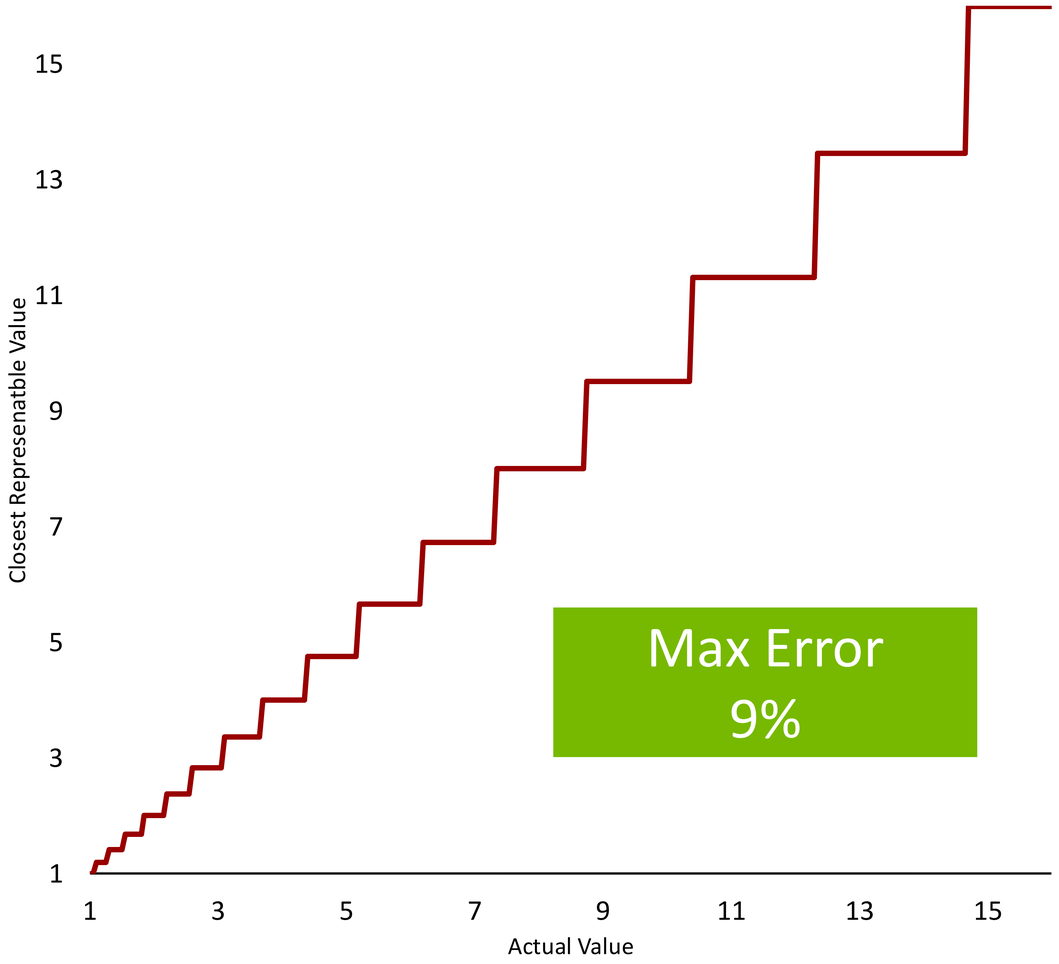
4-bit Log Representation (L2.2)



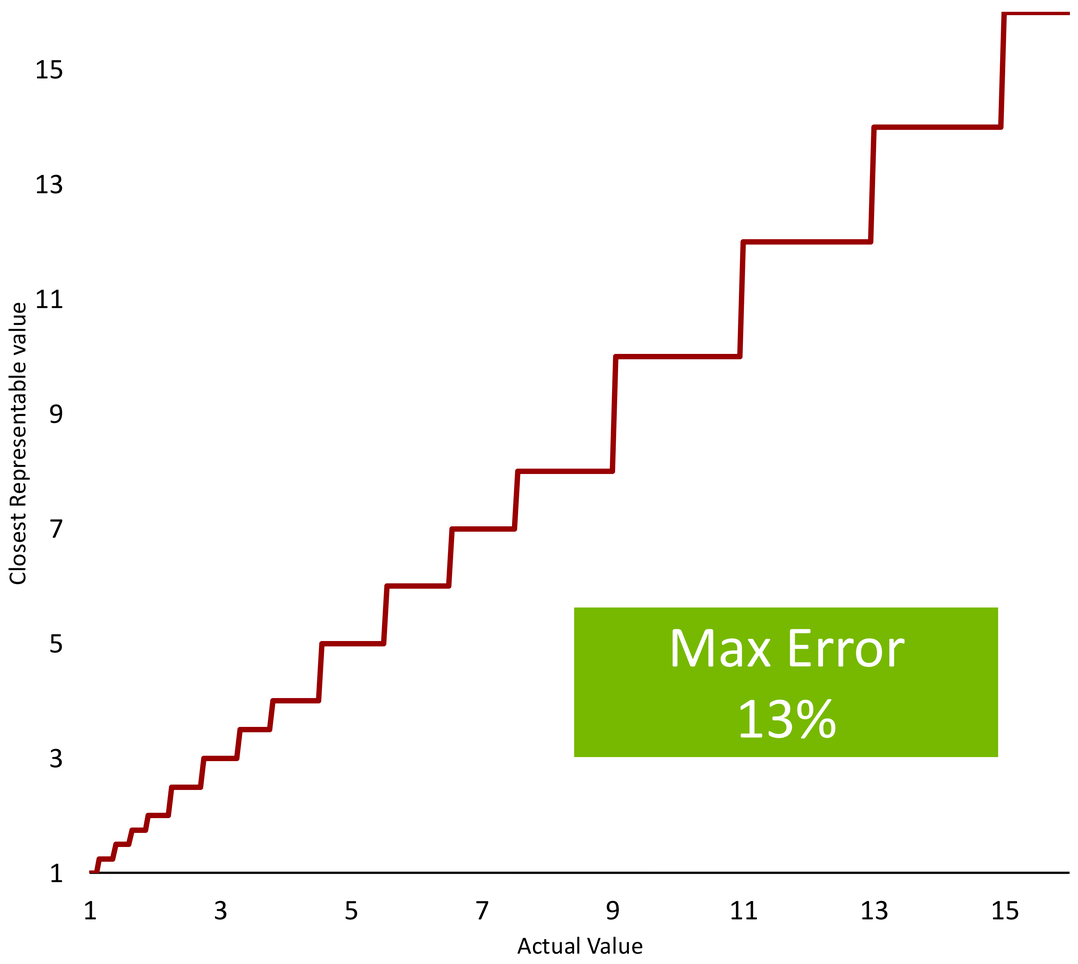
4-bit Integer Representation (Int4)



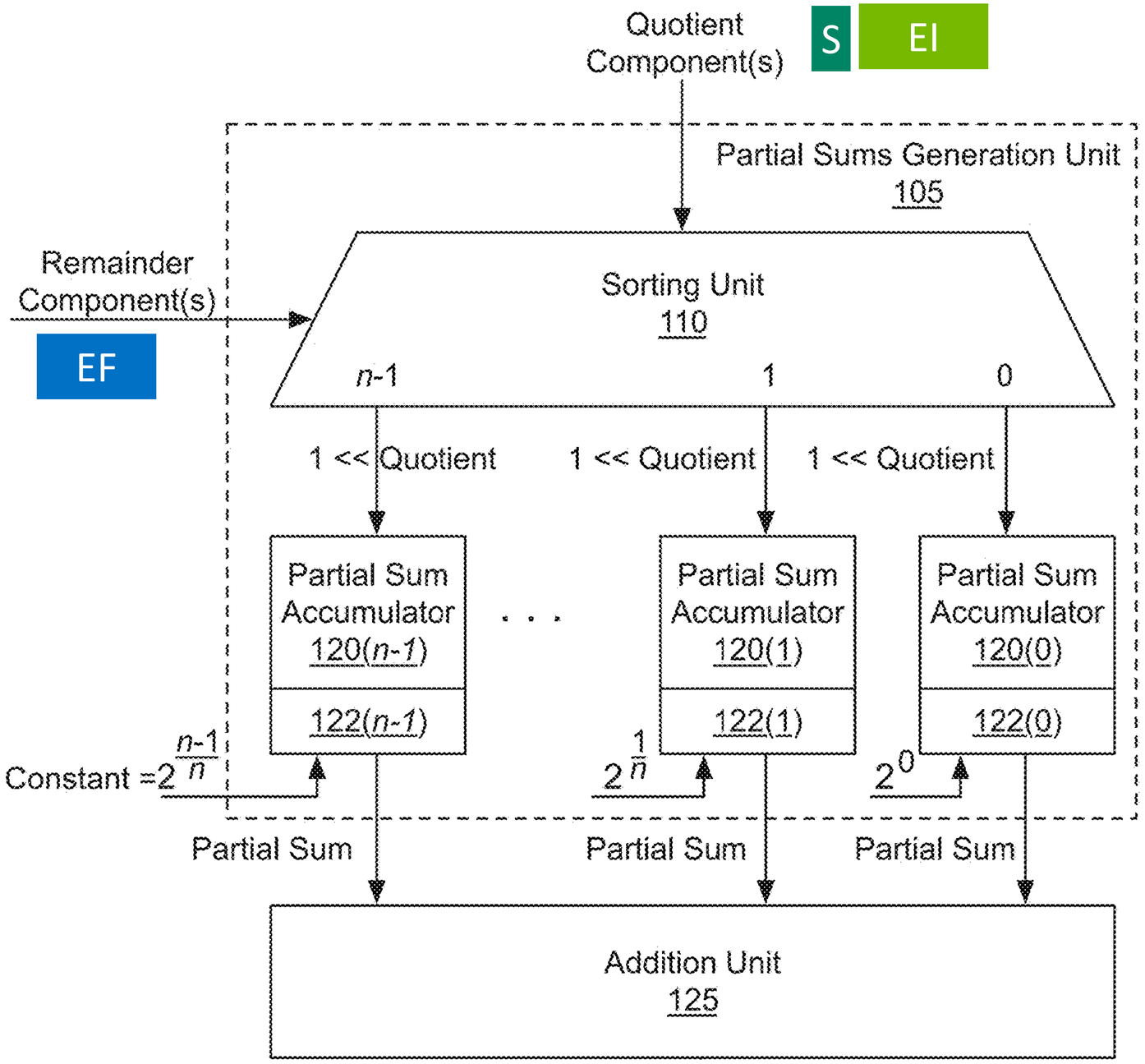
4-bit Log Representation (L2.2)



FP2.2



- Log Numbers
- Multiplies are cheap – just an add
- Adds are hard – convert to integer, add, convert back
  - Fractional part of log is a lookup
  - Integer part of log is a shift
- Can factor the lookup outside the summation
  - Only convert back after summation (and NLF)

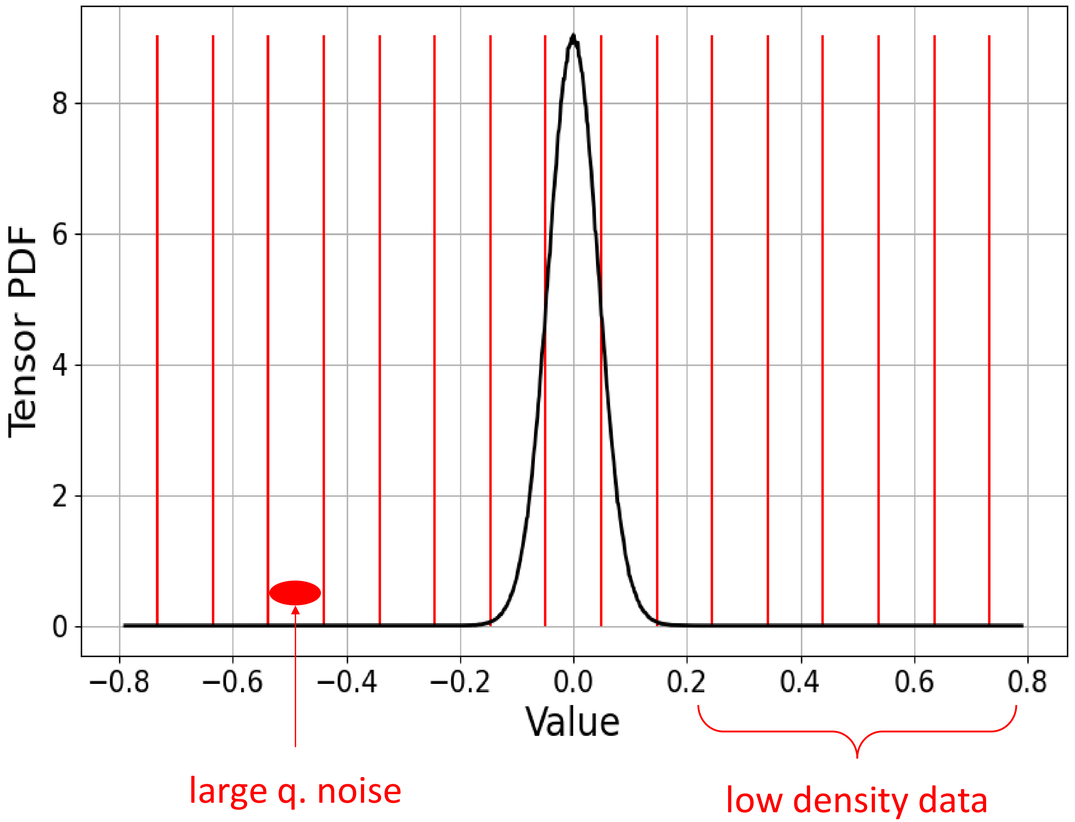

Quotient Component(s)
S
EI
Partial Sums Generation Unit
105
Remainder Component(s)
EF
Sorting Unit
110
n-1
1
0
1 << Quotient
1 << Quotient
1 << Quotient
Partial Sum Accumulator 120(n-1)
Partial Sum Accumulator 120(1)
Partial Sum Accumulator 120(0)
122(n-1)
122(1)
122(0)
Constant = 2^(n-1)/n
2^1/n
2^0
Partial Sum
Partial Sum
Partial Sum
Addition Unit
125

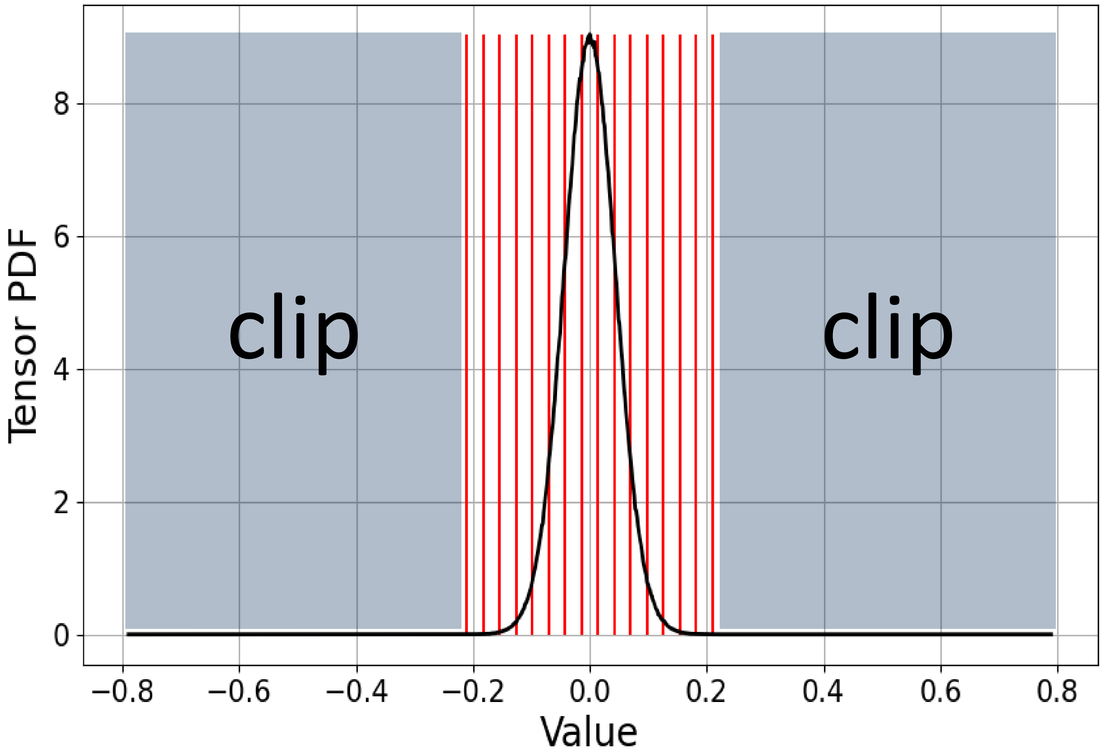

1
4
3
S
EI
EF

Optimum Clipping

Whatever number representation you use

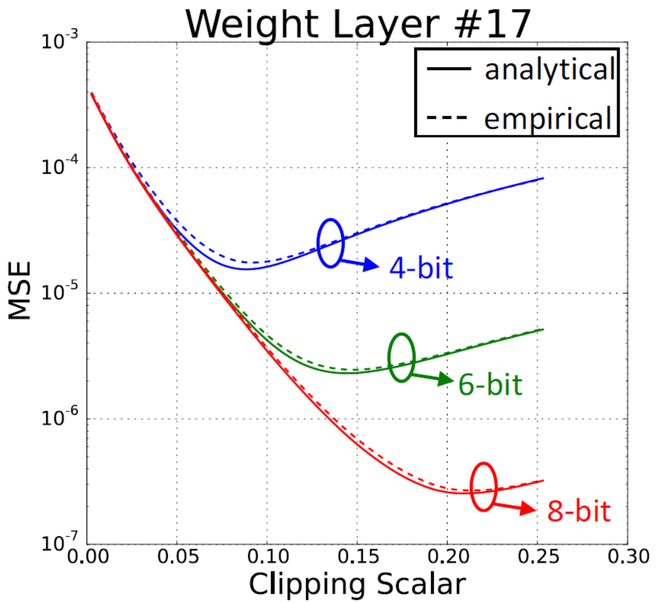
Pick the range optimally

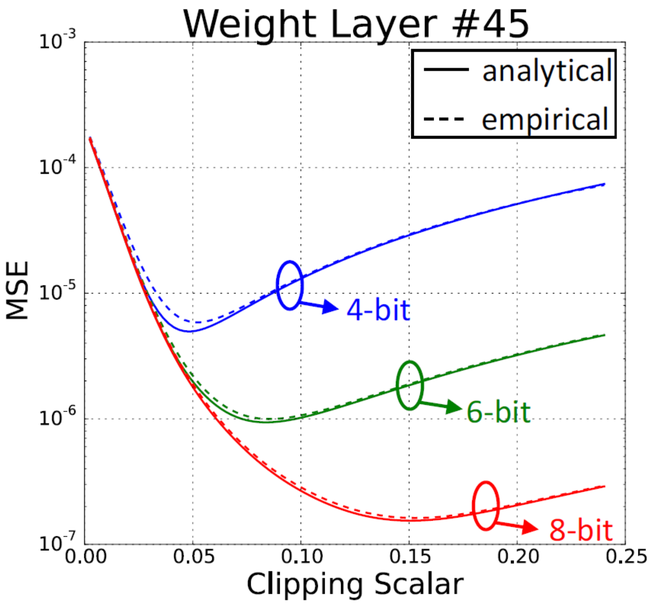

Tensor PDF
8
6
4
2
0
−0.8
−0.6
−0.4
−0.2
0.0
0.2
0.4
0.6
0.8
Value
large q. noise
low density data

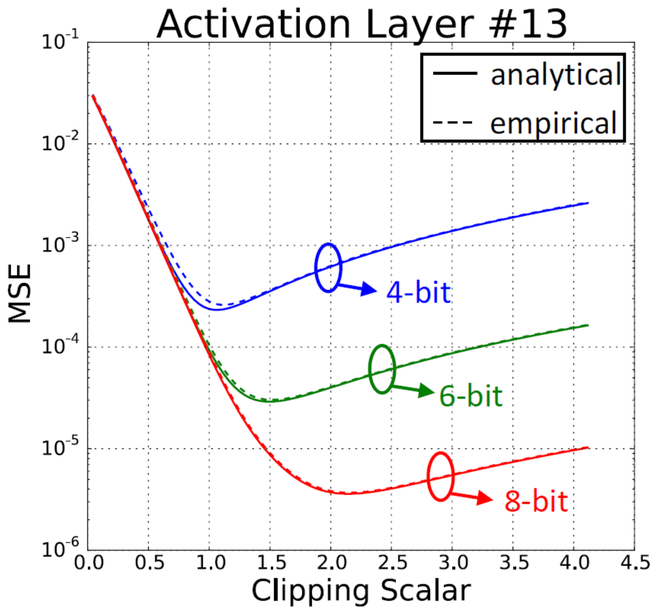

Tensor PDF
8
6
4
2
0
clip
clip
−0.8
−0.6
−0.4
−0.2
0.0
0.2
0.4
0.6
0.8
Value

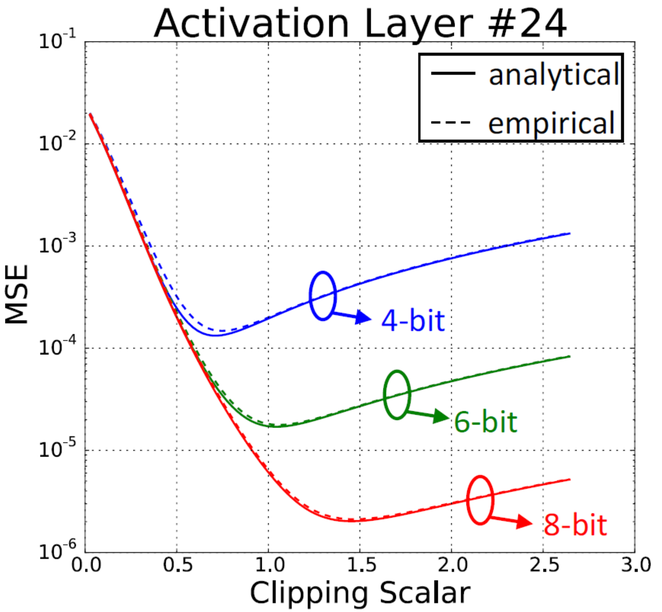
$$J = \frac{4^{-B}}{3} s^2 \int_0^s f_{|X|}(x)dx + \int_s^\infty (s-x)^2 f_{|X|}(x)dx$$

$$s_{n+1} = \frac{\boldsymbol{E}\left[|X| \cdot \mathbf{1}_{\{|X|>s_n\}}\right]}{\frac{4^{-B}}{3}\boldsymbol{E}\left[\mathbf{1}_{\{|X|<s_n\}}\right] + \boldsymbol{E}\left[\mathbf{1}_{\{|X|>s_n\}}\right]}$$









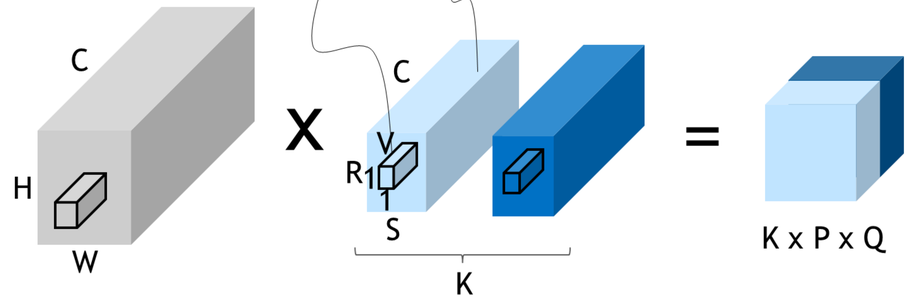
Vector Scaling

VS-Quant

Per-vector scaled quantization for low-precision inference

$$v \approx v_{int} \cdot \underbrace{s'_{int} \cdot s''}_{s'}$$

s'_{quant} s''

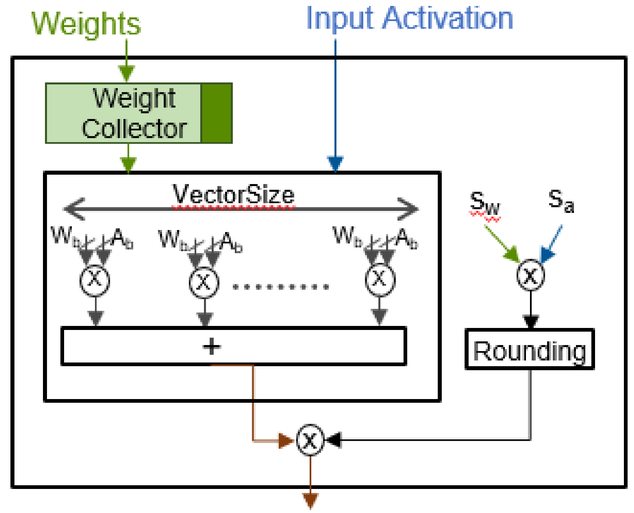


Input activation Weight Output activation

K x R x S x ceil(C/V) integer scale factors
+
K floating-point scale factors

Fine-grained scale factors per vector

$$y_q(j) = \left(\sum_{i=0}^{vecsize-1} w_q(i) a_q(i) \right) s_w(j) s_a(j)$$

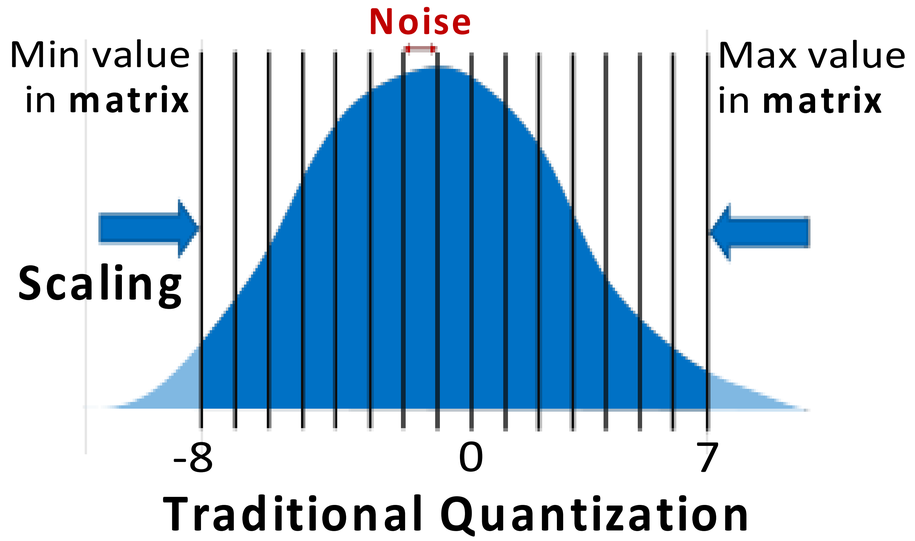


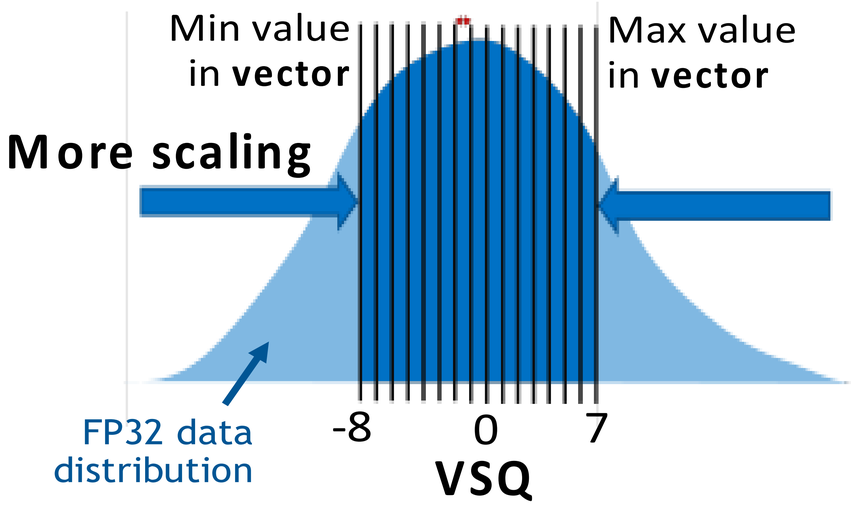
Modified vector MAC unit for VS-Quant

Works with either post-training quantization or quantization-aware retraining!

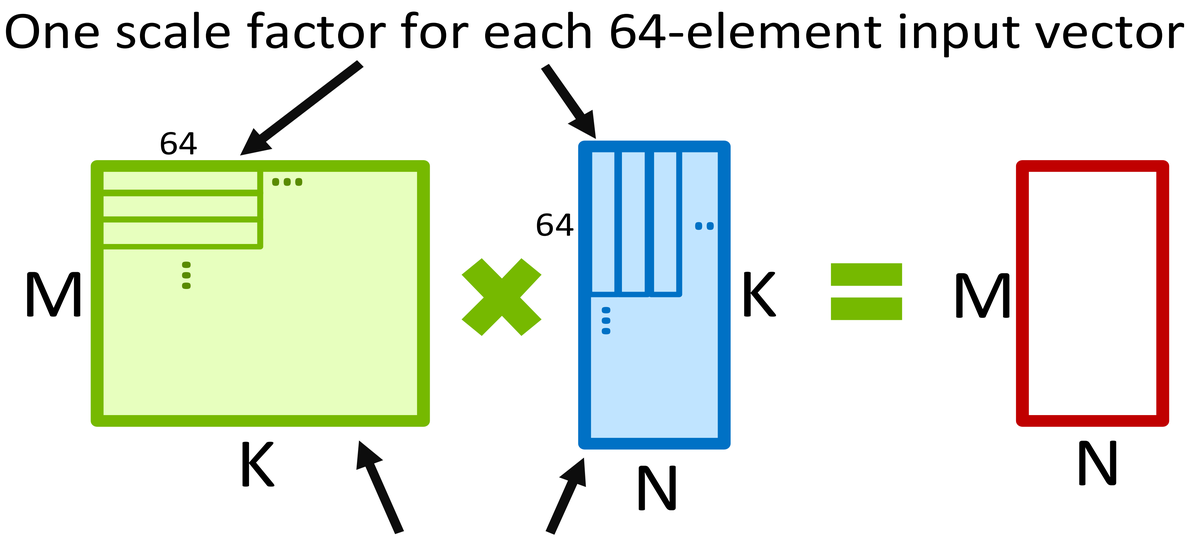
[Dai et al., MLSYS 2021]

INT4 Quantization





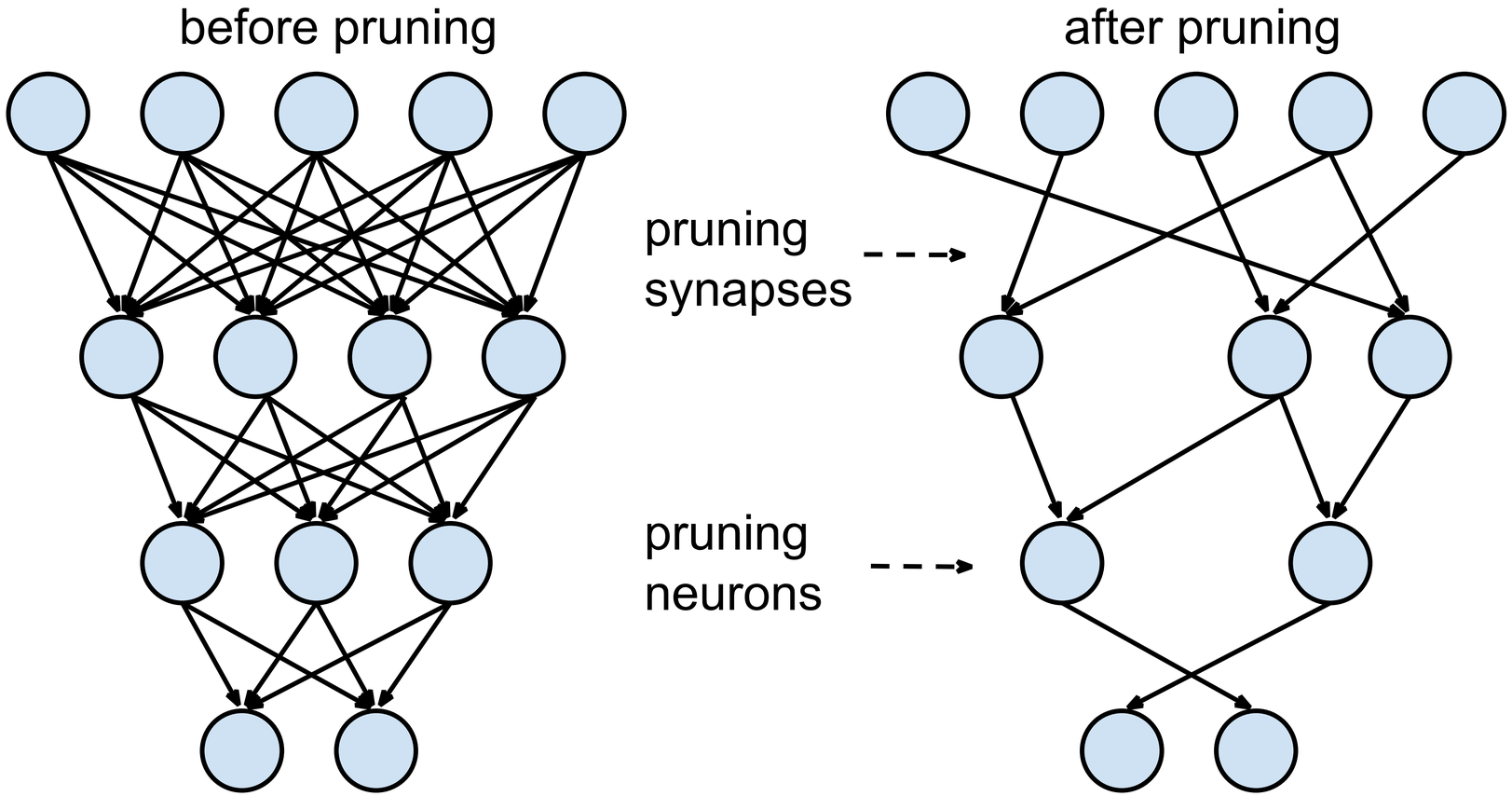
VSQ Scale Factors



Traditional Quantization	VSQ
One scale factor per matrix	Two scale factors: one per vector, one per matrix
High quantization noise	Reduced quantization noise

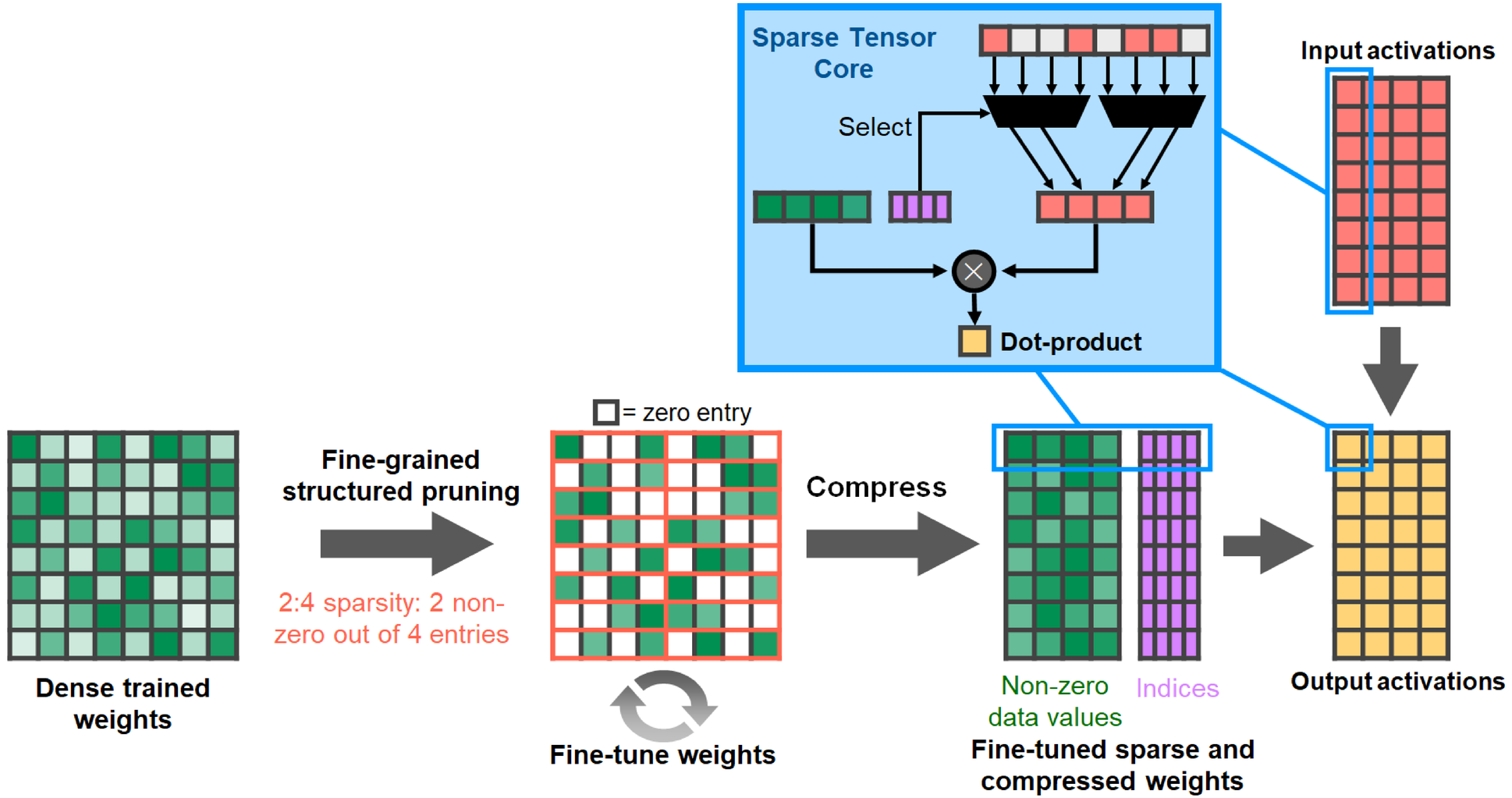
Sparsity

Pruning



Han et al. Learning both Weights and Connections for Efficient Neural Networks, NIPS 2015

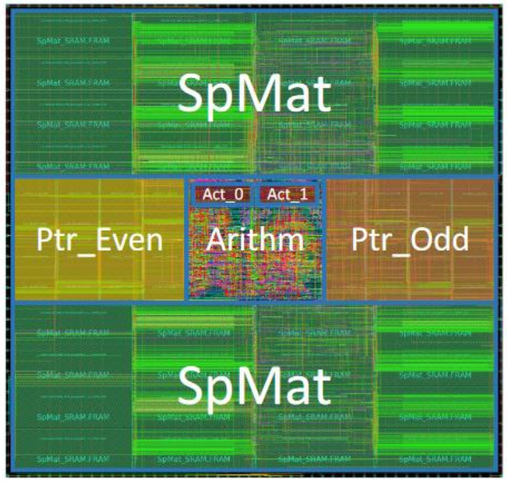
Structured Sparsity



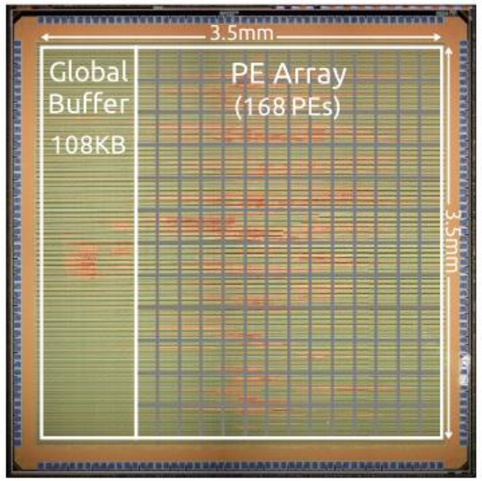
Mishra, Asit, et al. "Accelerating sparse deep neural networks." *arXiv preprint arXiv:2104.08378* (2021)

Accelerators

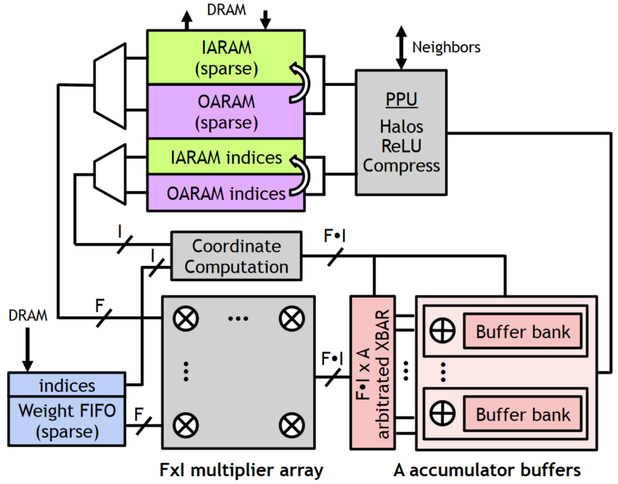
EIE (2016)



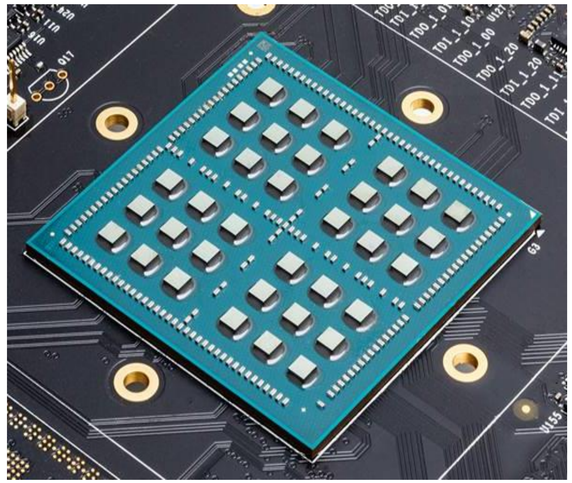
Eyeriss (2016)



SCNN (2017)



Simba (2018)



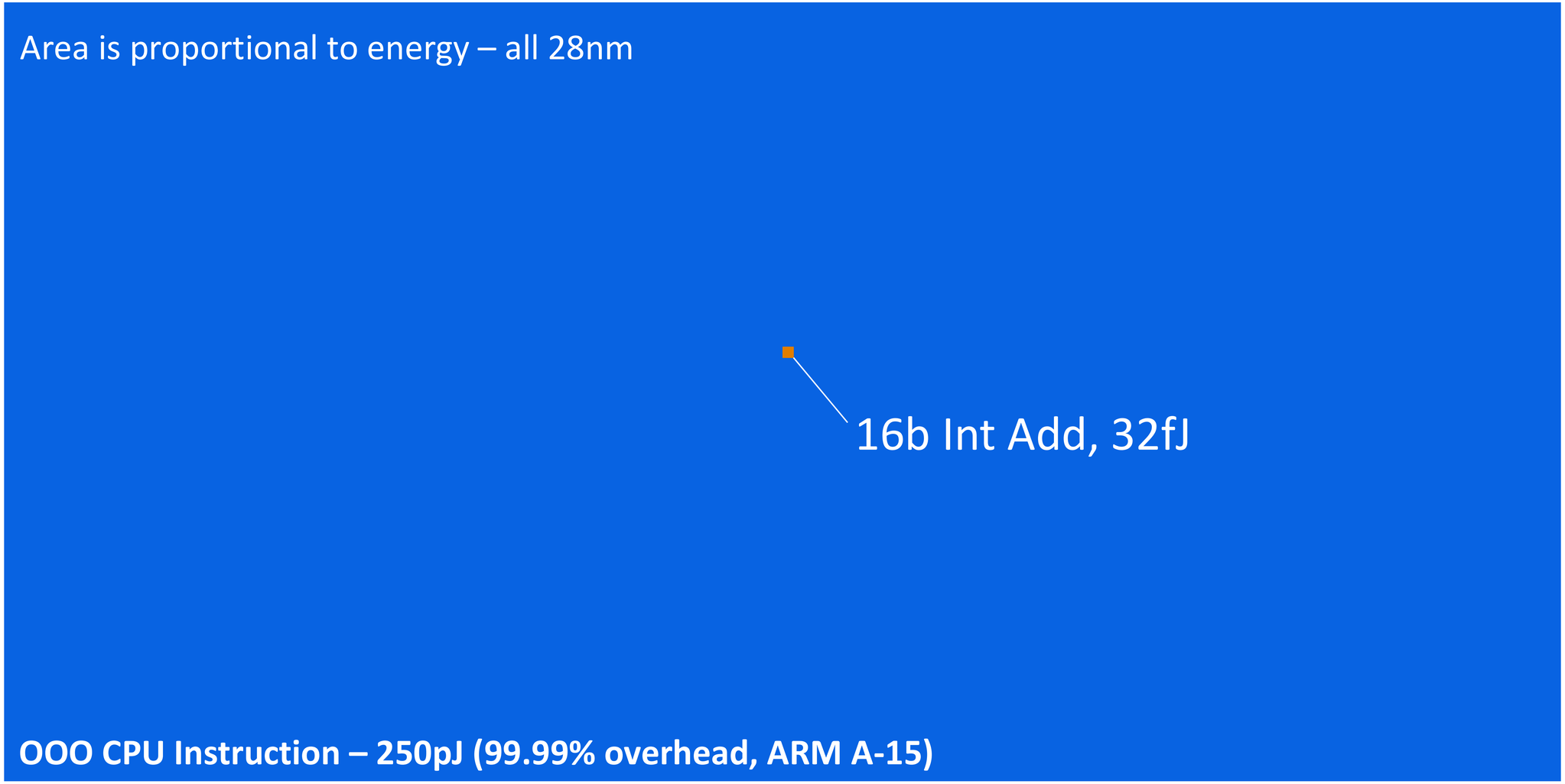
Accelerators Employ:

- Special **Data Types** and **Operations**
  - Do in 1 cycle what normally takes 10s or 100s – **10-1000x efficiency gain**
- Massive **Parallelism** – >1,000x, not 16x – with **Locality**
  - This gives performance, not efficiency
- Optimized **Memory**
  - High bandwidth (**and low energy**) for specific data structures and operations
- Reduced or Amortized **Overhead**
  - **10,000x efficiency gain** for simple operations
- Algorithm-Architecture **Co-Design**

Fast Accelerators since 1985

- **Mossim Simulation Engine**: Dally, W.J. and Bryant, R.E., 1985. A hardware architecture for switch-level simulation. *IEEE Trans. CAD*, *4*(3), pp.239-250.
- **MARS Accelerator**: Agrawal, P. and Dally, W.J., 1990. A hardware logic simulation system. *IEEE Trans. CAD*, *9*(1), pp.19-29.
- **Reconfigurable Arithmetic Processor**: Fiske, S. and Dally, W.J., 1988. *The reconfigurable arithmetic processor* . ISCA 1988.
- **Imagine**: Kapasi, U.J., Rixner, S., Dally, W.J., Khailany, B., Ahn, J.H., Mattson, P. and Owens, J.D., 2003. Programmable stream processors. *Computer*, *36*(8), pp.54-62.
- **ELM**: Dally, W.J., Balfour, J., Black-Shaffer, D., Chen, J., Harting, R.C., Parikh, V., Park, J. and Sheffield, D., 2008. Efficient embedded computing. *Computer*, *41*(7).
- **EIE**: Han, S., Liu, X., Mao, H., Pu, J., Pedram, A., Horowitz, M.A. and Dally, W.J., 2016, June. EIE: efficient inference engine on compressed deep neural network, ISCA 2016
- **SCNN**:Parashar, A., Rhu, M., Mukkara, A., Puglielli, A., Venkatesan, R., Khailany, B., Emer, J., Keckler, S.W. and Dally, W.J., 2017, June. Scnn: An accelerator for compressed-sparse convolutional neural networks, ISCA 2017
- **Darwin**: Turakhia, Bejerano, and Dally, “Darwin: A Genomics Co-processor provides up to 15,000× acceleration on long read assembly”, ASPLOS 2018.
- **SATiN**: Zhuo, Rucker, Wang, and Dally, “Hardware for Boolean Satisfiability Inference,”

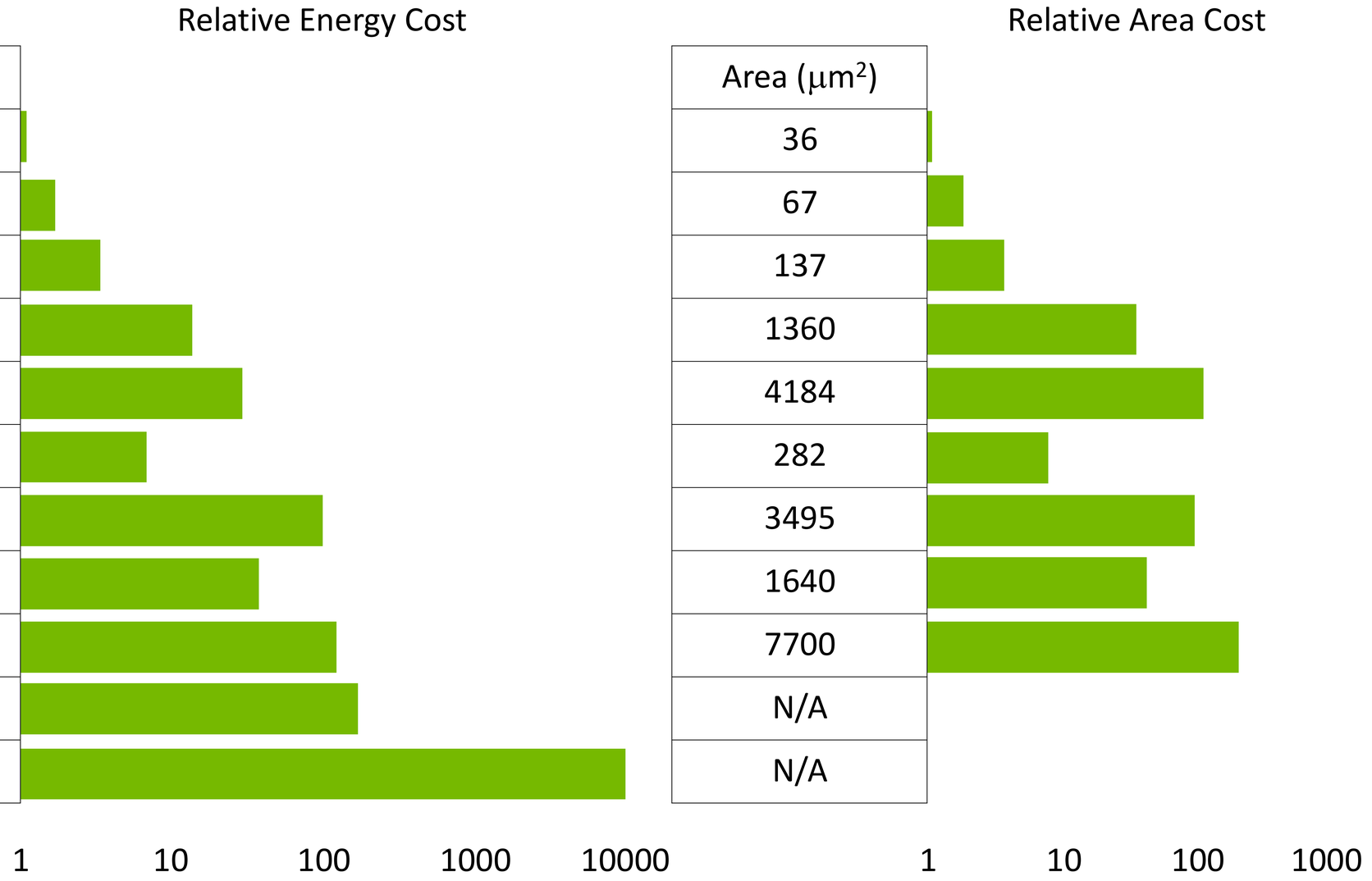
Eliminating Instruction Overhead



Evangelos Vasilakis. 2015. *An Instruction Level Energy Characterization of Arm Processors.* Foundation of Research and Technology Hellas, Inst. of Computer Science, Tech. Rep. FORTH-ICS/TR-450 (2015)

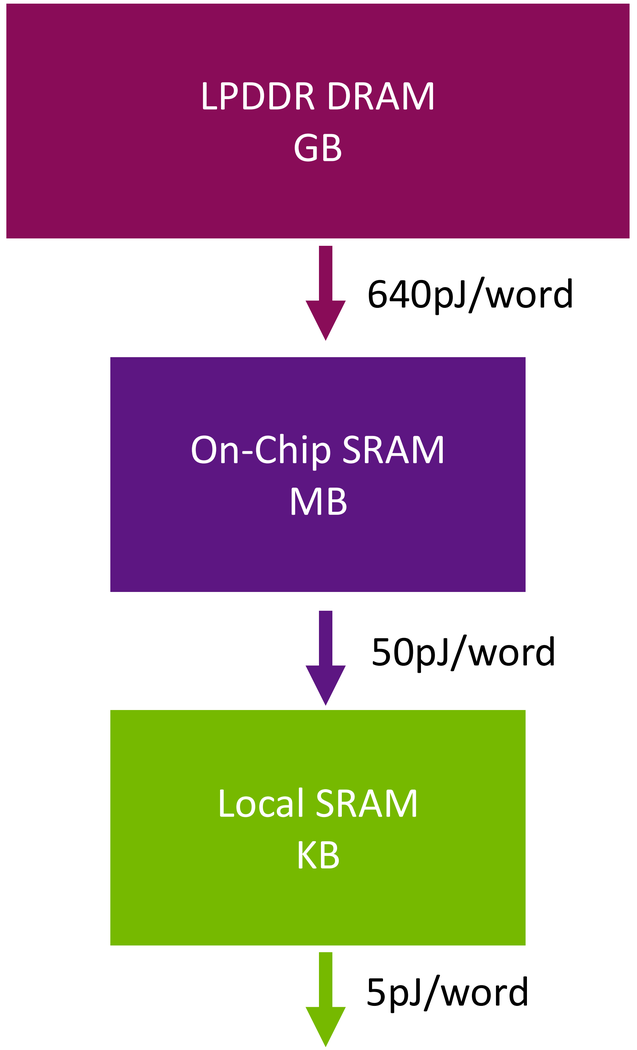
Cost of Operations

Operation:	Energy (pJ)	Area (μm^2)
8b Add	0.03	36
16b Add	0.05	67
32b Add	0.1	137
16b FP Add	0.4	1360
32b FP Add	0.9	4184
8b Mult	0.2	282
32b Mult	3.1	3495
16b FP Mult	1.1	1640
32b FP Mult	3.7	7700
32b SRAM Read (8KB)	5	N/A
32b DRAM Read	640	N/A



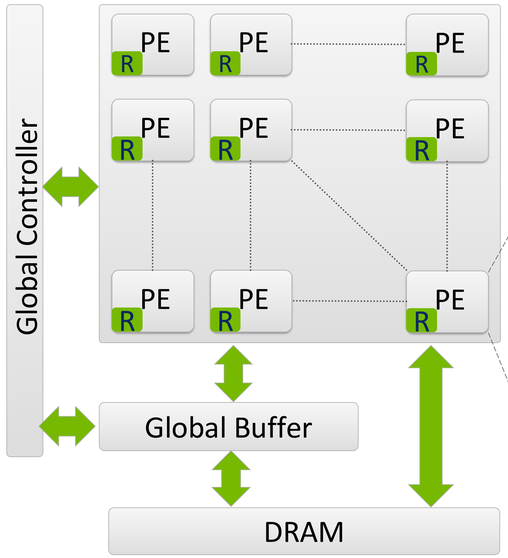
Energy numbers are from Mark Horowitz “Computing’s Energy Problem (and what we can do about it)”, ISSCC 2014
Area numbers are from synthesized result using Design Compiler under TSMC 45nm tech node. FP units used DesignWare Library.

The Importance of Staying Local

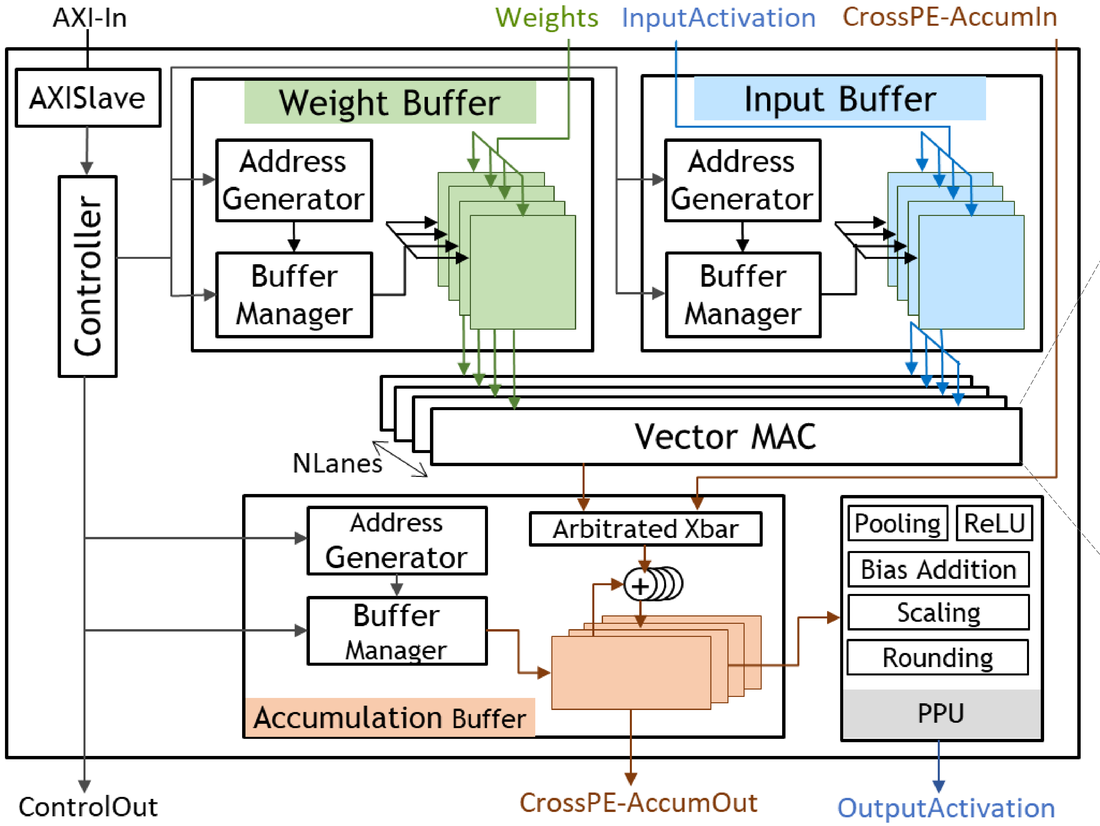


Magnet

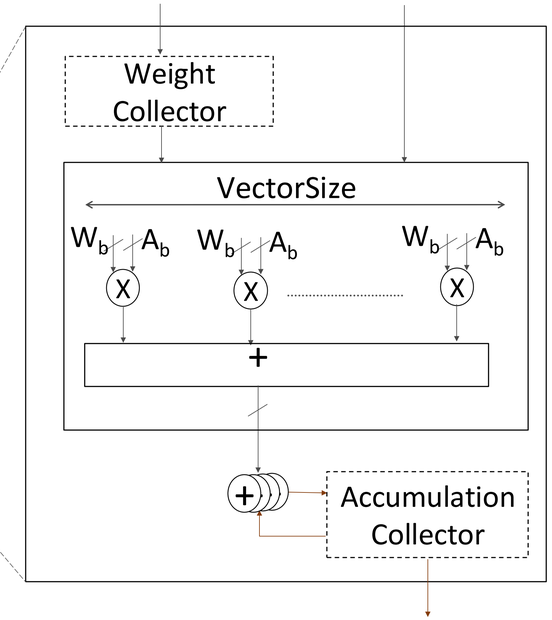
Configurable using synthesizable SystemC, HW generated using HLS tools



MAGNet System



Processing Element (PE)



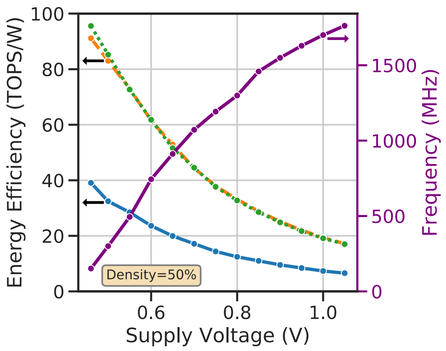
Vector MAC unit

[Venkatesan et al., ICCAD 2019]

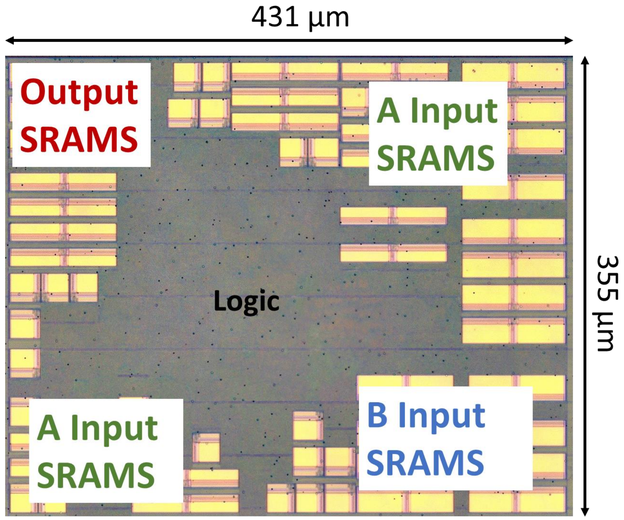
Energy-efficient DL Inference accelerator

Transformers, VS-Quant INT4, TSMC 5nm

- Efficient architecture
  - Used MAGNet *[Venkatesan et al., ICCAD 2019]* to design a low-precision DL inference accelerator for Transformers
  - Multi-level dataflow to improve data reuse and energy efficiency
- Low-precision data format: VS-Quant INT4
  - Hardware-software techniques to tolerate quantization error
  - Enable low cost multiply-accumulate (MAC) operations
  - Reduce storage and data movement
- Special function units

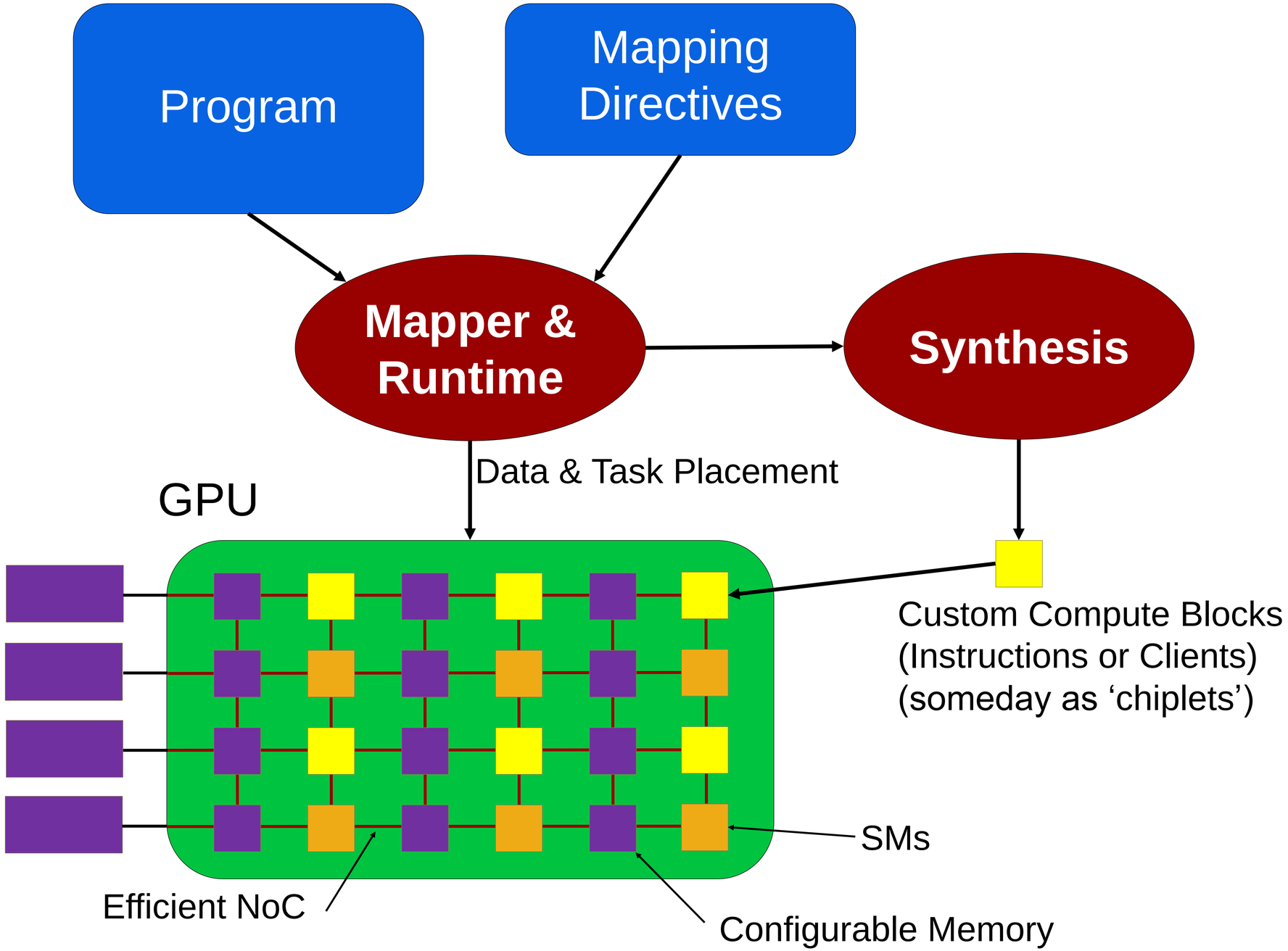


- 95.6 TOPS/W with 50%-dense 4-bit input matrices with VSQ enabled at 0.46V
- 0.8% energy overhead from VSQ support with 50%-dense inputs at 0.67V



- TSMC 5nm
- 1024 4-bit MACs/cycle (512 8-bit)
- 0.153 mm^2 chip
- Voltage range: 0.46V – 1.05V
- Frequency range: 152 MHz – 1760 MHz

[Keller, Venkatesan, et al., "A 95.6-TOPS/W Deep Learning Inference Accelerator with Per-Vector Scaled 4-bit Quantization in 5nm", JSSC 2023]


Program
Mapping Directives
Mapper & Runtime
Synthesis
Data & Task Placement
GPU
Custom Compute Blocks
(Instructions or Clients)
(someday as ‘chiplets’)
SMs
Efficient NoC
Configurable Memory

Conclusion

Conclusion

- Deep Learning was enabled by hardware and its progress is limited by hardware
- 1000x in last 10 years
  - Number representation, complex ops, sparsity
- Logarithmic numbers
  - Lowest worst-case error for a given number of bits
  - Can ‘factor out’ hard parts of an add
- Optimum clipping
  - Minimize MSE by trading quantization noise for clipping noise
- VS-Quant
  - Separate scale factor for each small vector – 16 to 64 scalars
- Accelerators – Testbeds for GPU ‘cores’
  - Test chip validates concepts and measures efficiency
  - 95.6 TOPS/W on BERT with negligible accuracy loss