“The Hot Chip is a Rack”
(AI Literally Demands we Think Outside the Box)

August 24, 2025

DC Racks Tutorial
Organized by AMD, Google®, Meta, Microsoft®, NVIDIA
8:30 am start

45 min How AI workloads shape rack system architecture (Frank Helms)
25 min Scaling fabric technologies (Darrin Vallis)
25 min Liquid cooling with Google characteristics (Jorge Padilla)
25 min Re-Architected power systems (Harsha Bojja)

10:30-10:55 am coffee break

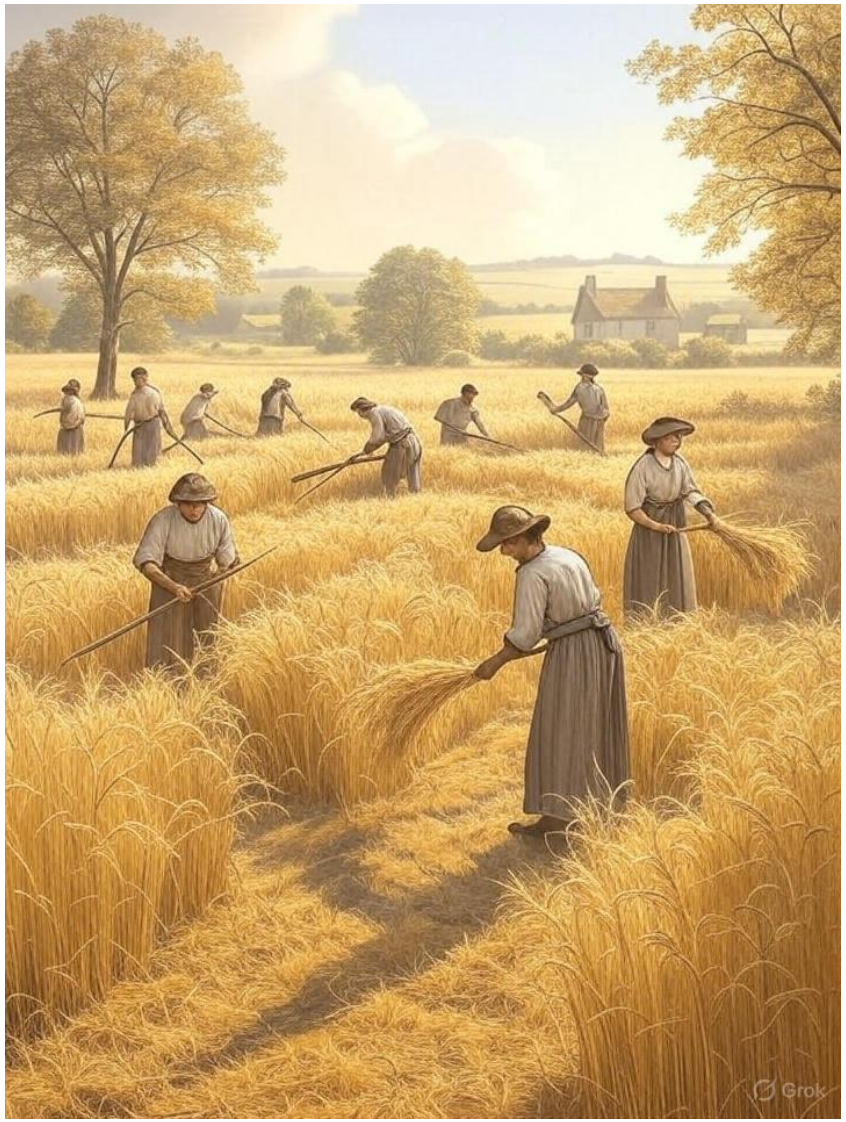
30 min Case study: NVIDIA GB200 NVL72 (John Norton)
30 min Case study: Meta NVL72 (Matt Bowman and William Arnold)
30 min Case study: Google TPU Rack (Pankaj Makhija)

Go to 12:30 max

12:30 pm Lunch starts

How AI Workloads Shape Hardware Architecture

Why Should We Care About AI?



Why Should We Care About AI?

Why Should We Care About AI?

Microsoft to spend $80bn on AI data centers in 2025

Company says US will have to be smart to win AI race with China

January 06, 2025 By: Matthew Gooding Have your say

Amazon's $150 Billion Cloud Power Play: Inside Project Rainier's AI Supercomputer

April 3, 2025 in **Cloud** Reading Time: 5 mins read

X.AI: "We're running the world's biggest supercomputer, Colossus. Built in 122 days—outpacing every estimate—it was the most powerful AI training system yet. Then we doubled it in 92 days to 200k GPUs. This is just the beginning."

In a Facebook post Friday, Zuckerberg said that Meta expects to spend $60 billion-$80 billion on CapEx in 2025, primarily on data centers and growing the company's AI development teams. That projected range is around double the $35 billion-$40 billion Meta spent on CapEx last year.

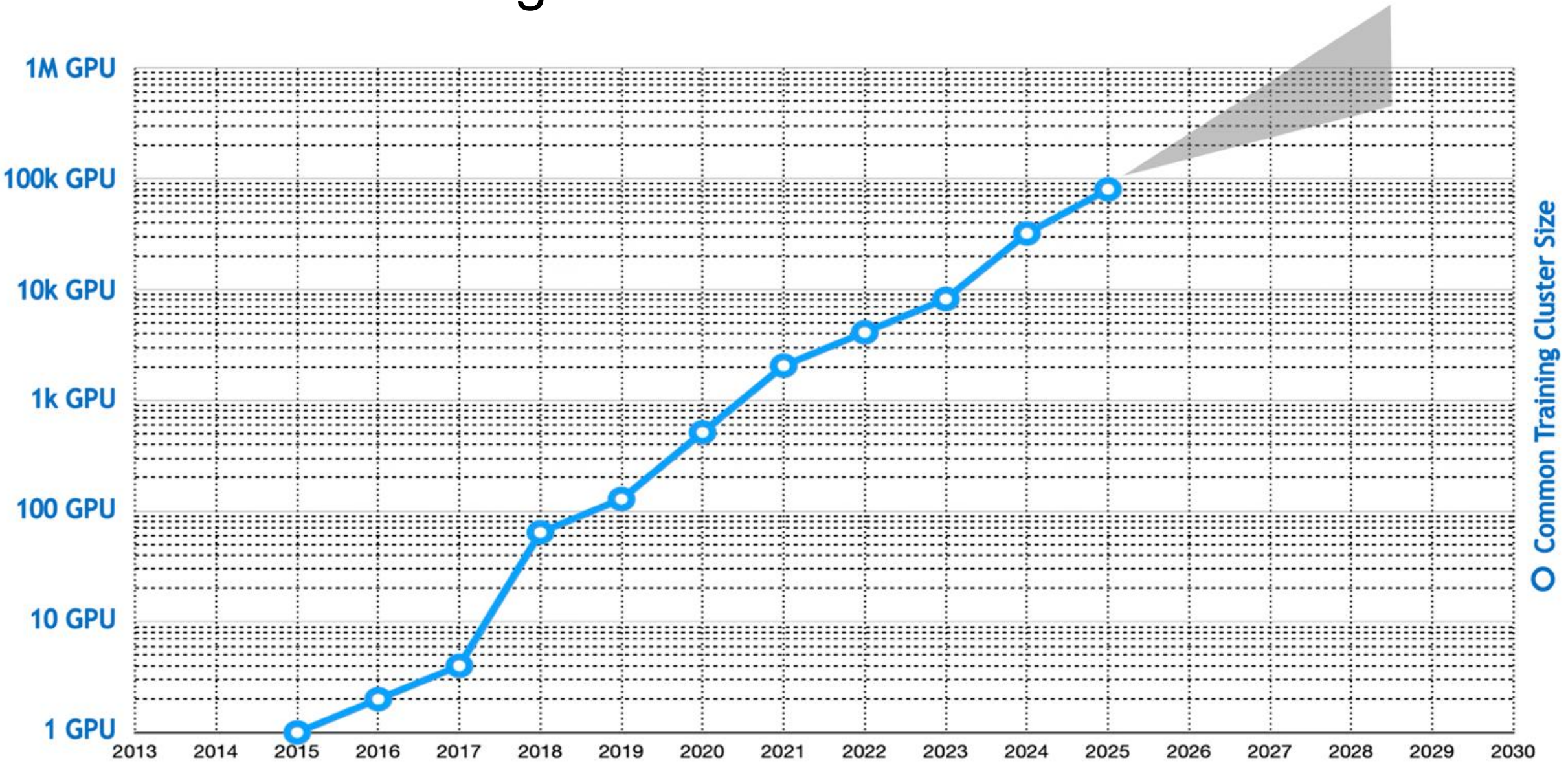
Alphabet's $75 billion AI expansion: cloud revenue slows, investors react

February 6, 2025

The Stargate Project is a new company which intends to invest $500 billion over the next four years building new AI infrastructure for OpenAI in the United States. We will begin deploying $100 billion immediately. This infrastructure will secure American leadership in AI, create hundreds of thousands of American jobs, and generate massive economic benefit for the entire world. This project will not only support the re-industrialization of the United States but also provide a strategic capability to protect the national security of America and its allies.

The initial equity funders in Stargate are SoftBank, OpenAI, Oracle, and MGX.

Common Training Cluster Size



The AI Continuum

After AGI comes

"Artificial super intelligence" (ASI)

AI that surpasses the best human minds in every domain

Experts project ASI within a few years of AGI

The race is on for

"Artificial general intelligence" (AGI)

OpenAI on AGI
"Highly autonomous systems that outperform humans at most economically valuable work"

Experts project AGI within 5 to 20 years

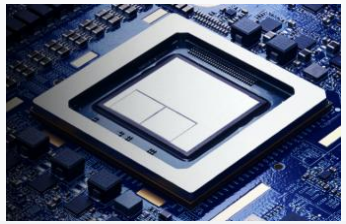
"Narrow AI" is already enabling unprecedented "knowledge worker" productivity

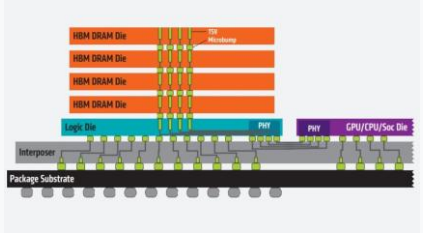
AI from Chips to Continents

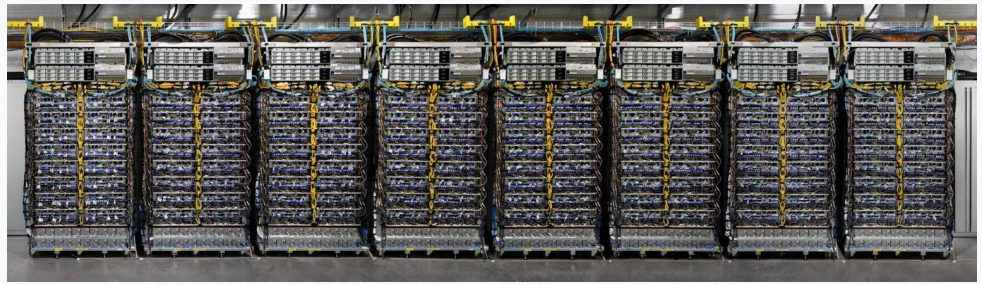
HOT CHIPS

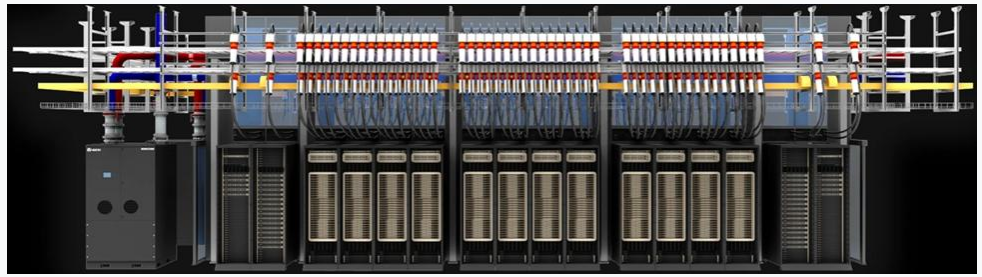
Today's focus:
Racks as the building block of clusters

HotChips
historic focus





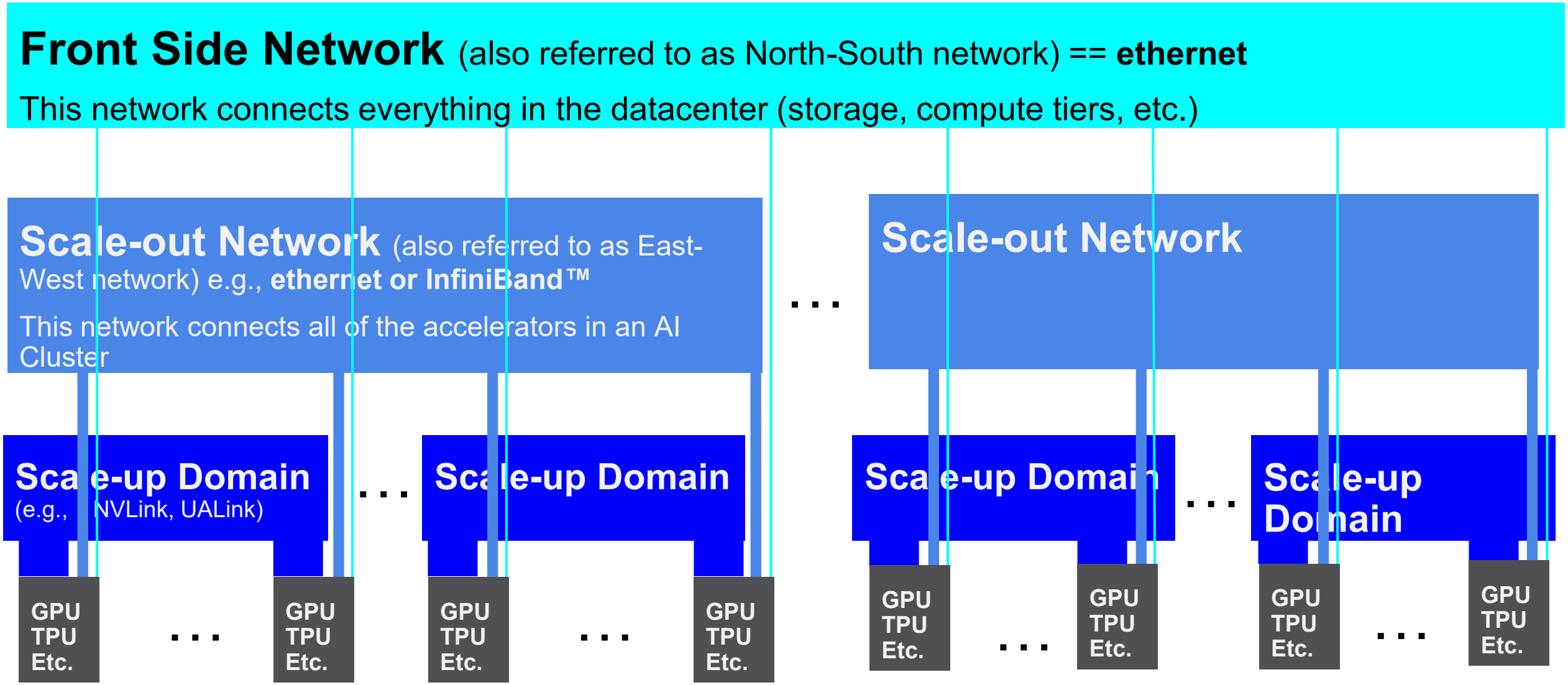




...

Clonee Data Center | Meta

Taxonomy of Interconnects for AI Accelerators

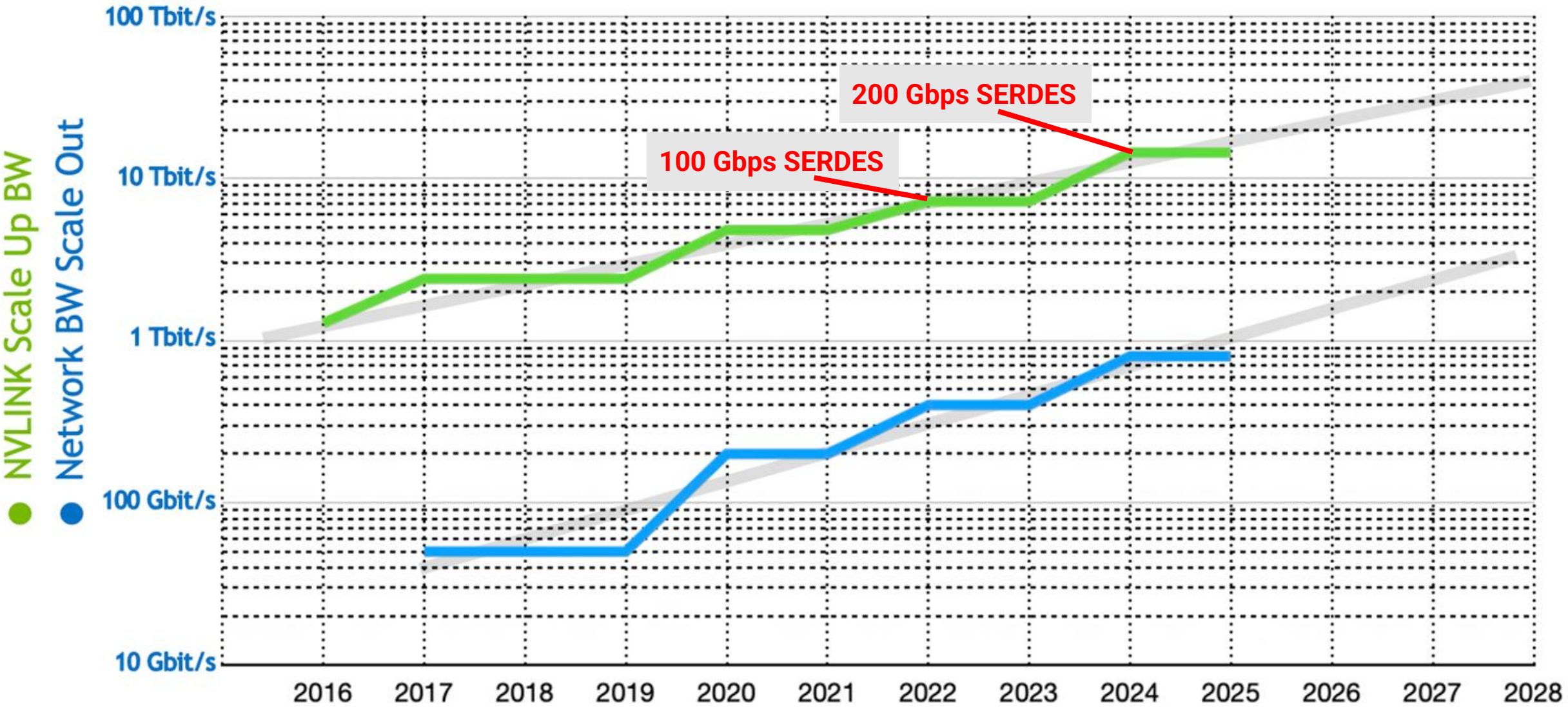


Physical Medium for Interconnect

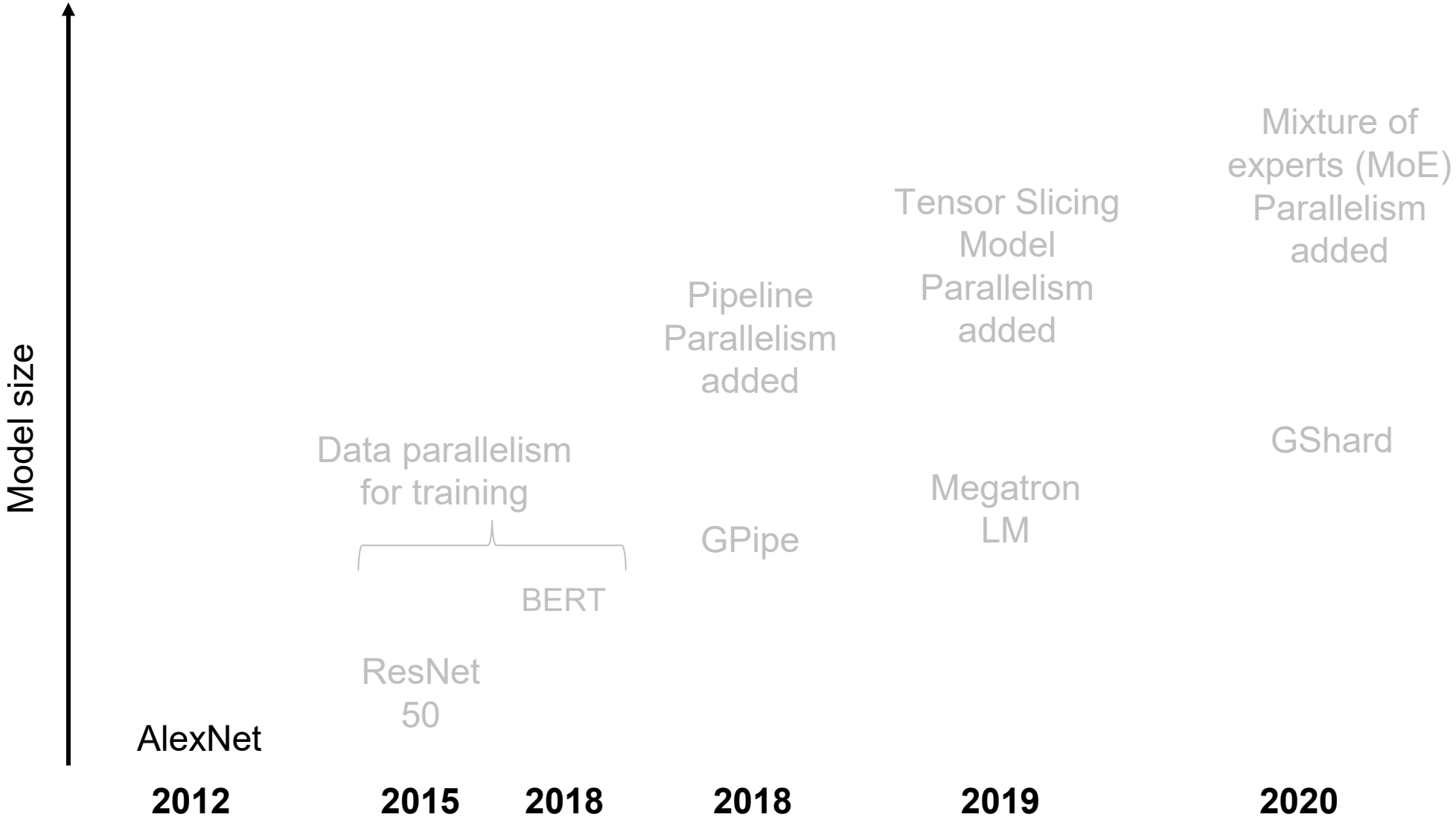
	Examples	Reach at 200Gbps	Reliability	Cost	Power	Where it is used so far
Passive Copper connectors and Cables	DAC; NVL72 NVLink cable cartridges; TPU torus	~ 1m	Highest	Lowest	~ 5 pJ/bit	Scale-up interconnect e.g., DGX, HGX, NVL72, TPU
Active electrical cables (AEC)	Retimers provide extended reach	< 5m	Second highest	Higher	~ 23 pJ/bit	In-rack to top of rack switch, and adjacent racks
Co-packaged Optics		30 to 50m		Higher	~ 5 to 10 pJ/bit	Not yet deployed at scale
Active optical cables (AOC)	Adds Optical to AEC	30 to 50m	Lower reliability than electrical	Even higher	> AEC	Between switches for frontside datacenter network; scale-out network
Optical Transceivers	Higher power higher drive optics	~ km+		Highest	Highest	Longer reach switch to switch interconnect

- Use passive copper where you can
- Use optical interconnect where you must

Scale Out & Scale Up Bandwidth per GPU

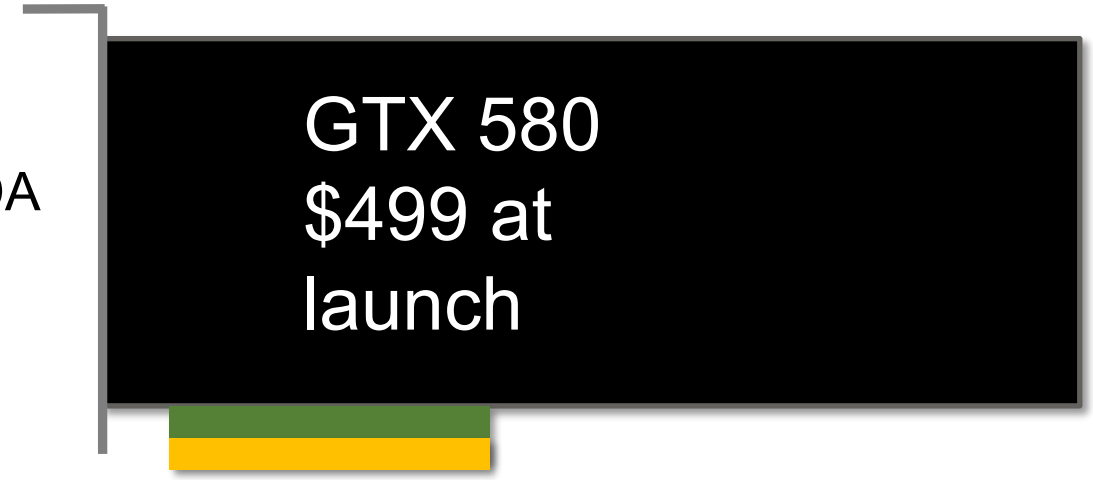


AI Model Building Blocks & Forms of Parallelism



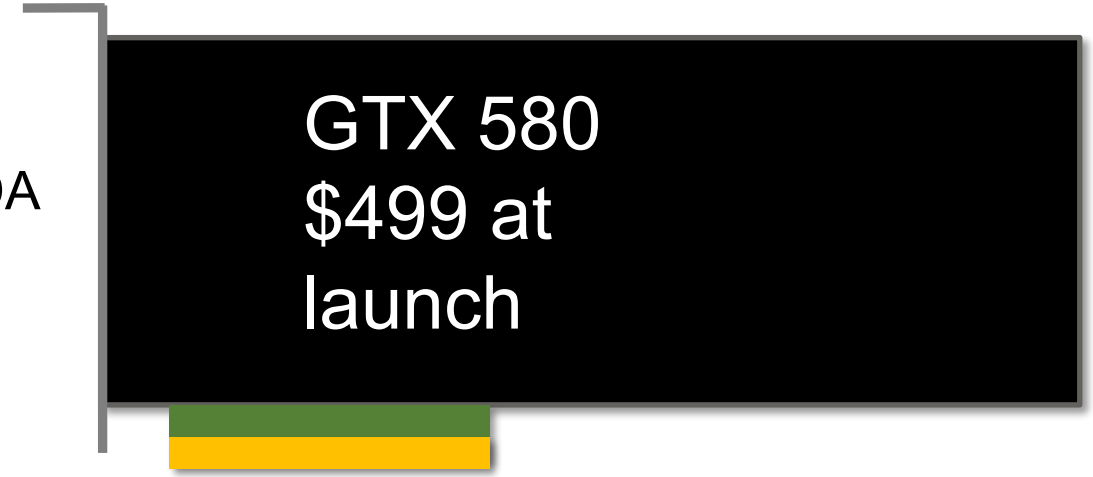
Enablers of the Beginning

- Affordable compute = a GPU
- A GPU compute programming language = CUDA



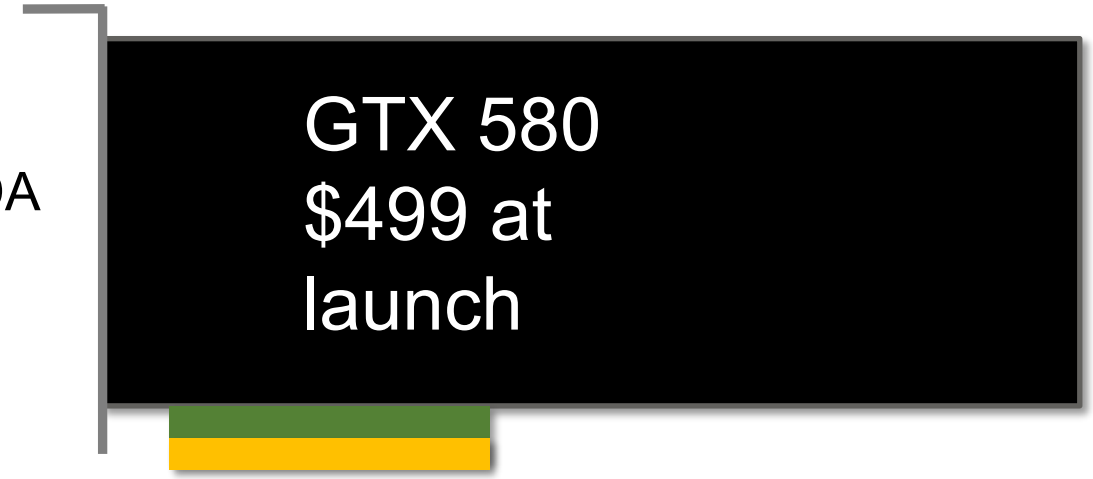
Enablers of the Beginning

- Affordable compute = a GPU
- A GPU compute programming language = CUDA
- A visionary human Fei-Fei Li
  - A labeled data set large enough to train an image classification model = ImageNet
  - Motivation: ImageNet Large Scale Visual Recognition Challenge



Enablers of the Beginning

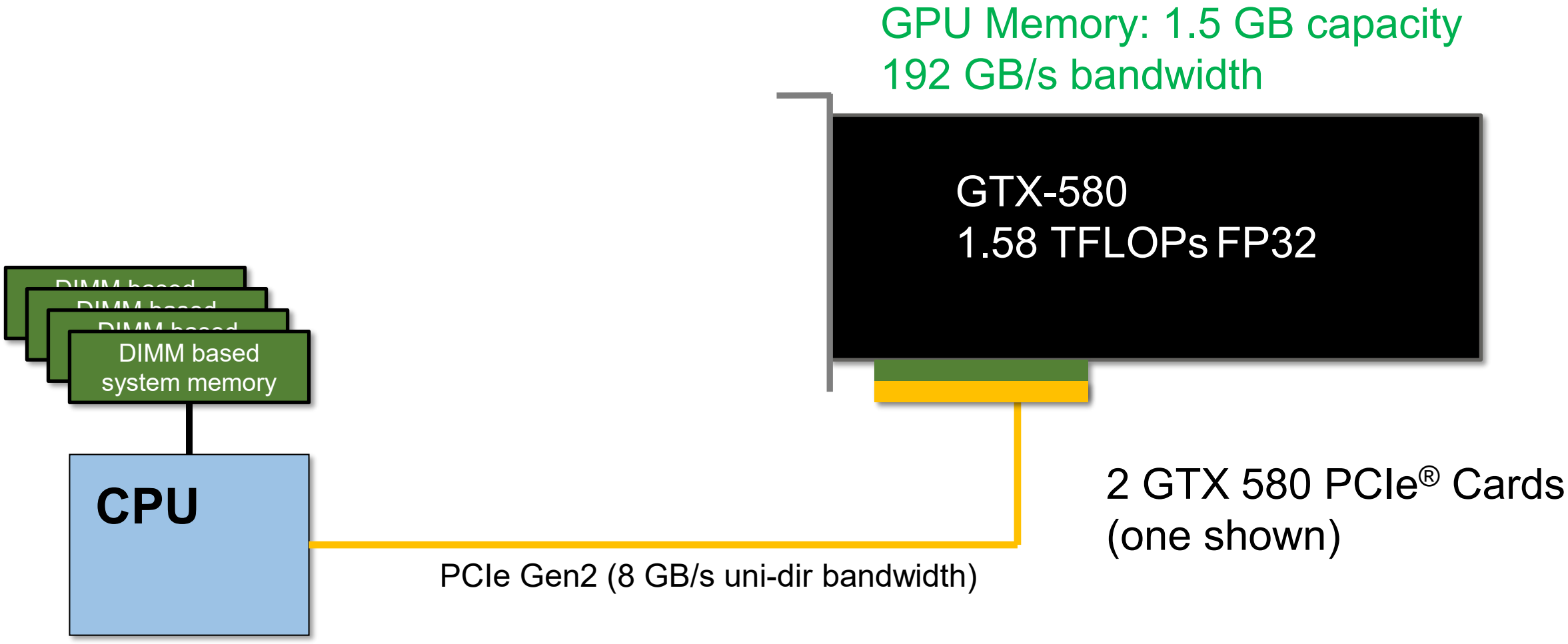
- Affordable compute = a GPU
- A GPU compute programming language = CUDA
- A visionary human Fei-Fei Li
  - A labeled data set large enough to train an image classification model = ImageNet
  - Motivation: ImageNet Large Scale Visual Recognition Challenge



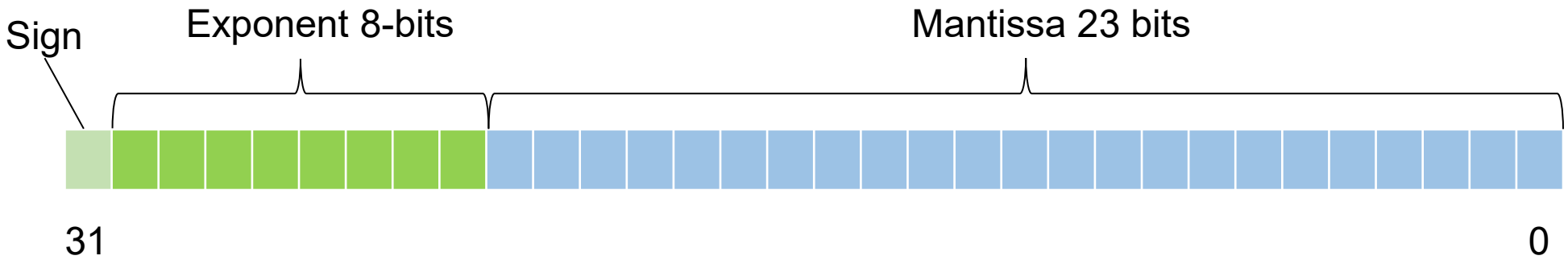
- A few more visionary humans brought the “AlexNet” Image Classification model into existence

Geoffrey Hinton Ilya Sutskever Alex Krizhevsky

The System
(Node = System Under a Single Host OS Image)



Number Representation for Training in 2012: FP32

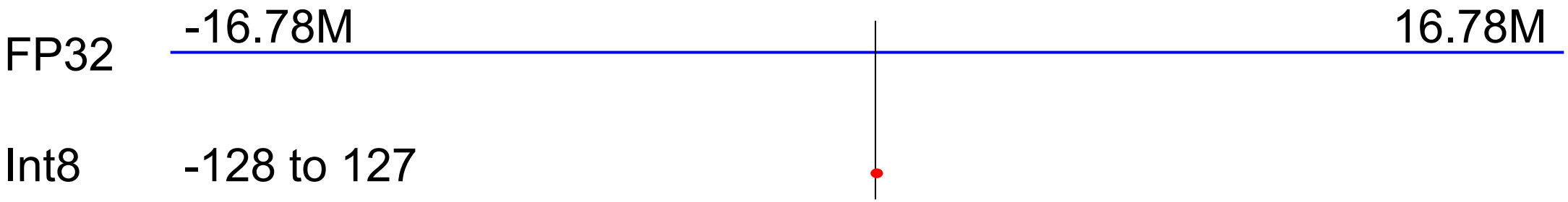


This for 2012 Training: FP32 = 32 bits for a floating-point representation

Not Int8 for Training: 8 bits and can represent 256 integer values

Why FP32 for Training in 2012?

1. Range



2. Precision within the range

Example Pi: **3.14159265358979323846264 3...**

FP32 can represent Pi as: **3.141593**

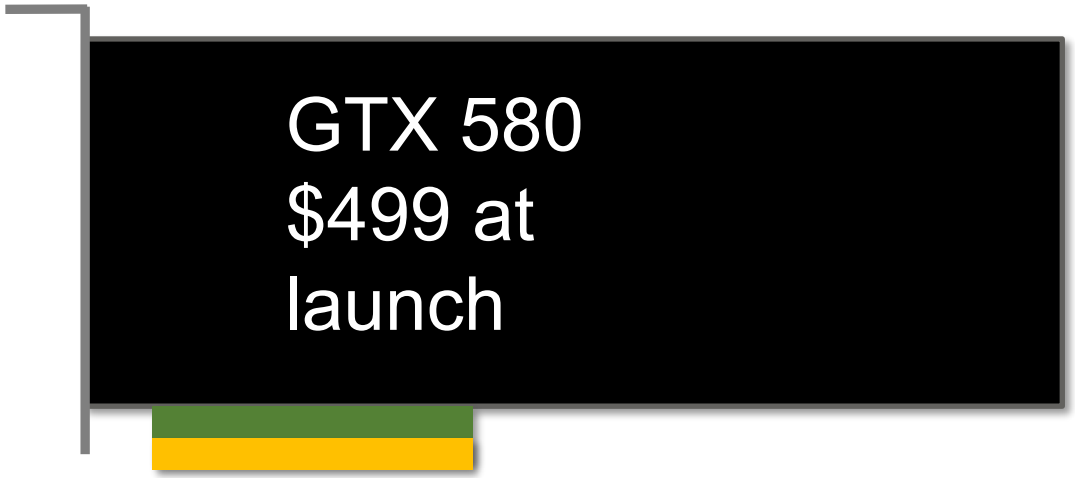
Int8 can represent Pi as **3**

Both range and precision required for model training convergence

The Beginning: 2012 AlexNet

AlexNet was trained

- On ImageNet (over 14 million labeled images covering more than 20,000 classes of images)
- In 5 to 6 days using 2 GTX 580s

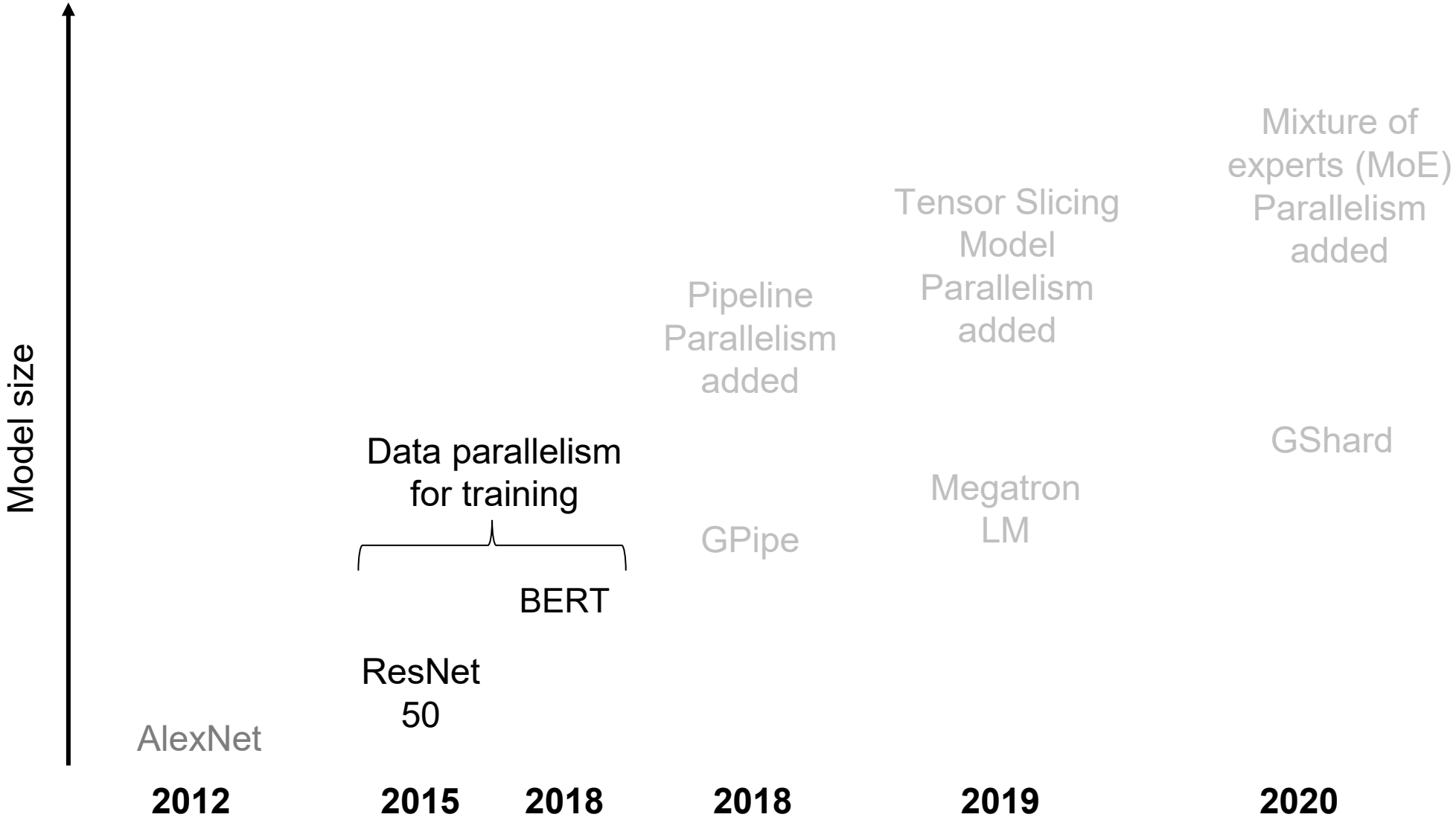


GTX 580

- TSMC 40nm
- 520 mm2 die
- 3 Billion transistors
- 16 SM
- 1.544 GHz engine clock
- 1.58 TFLOPs of FP32
- 192 GB/s of GDDR5 BW
- Card power 244 W

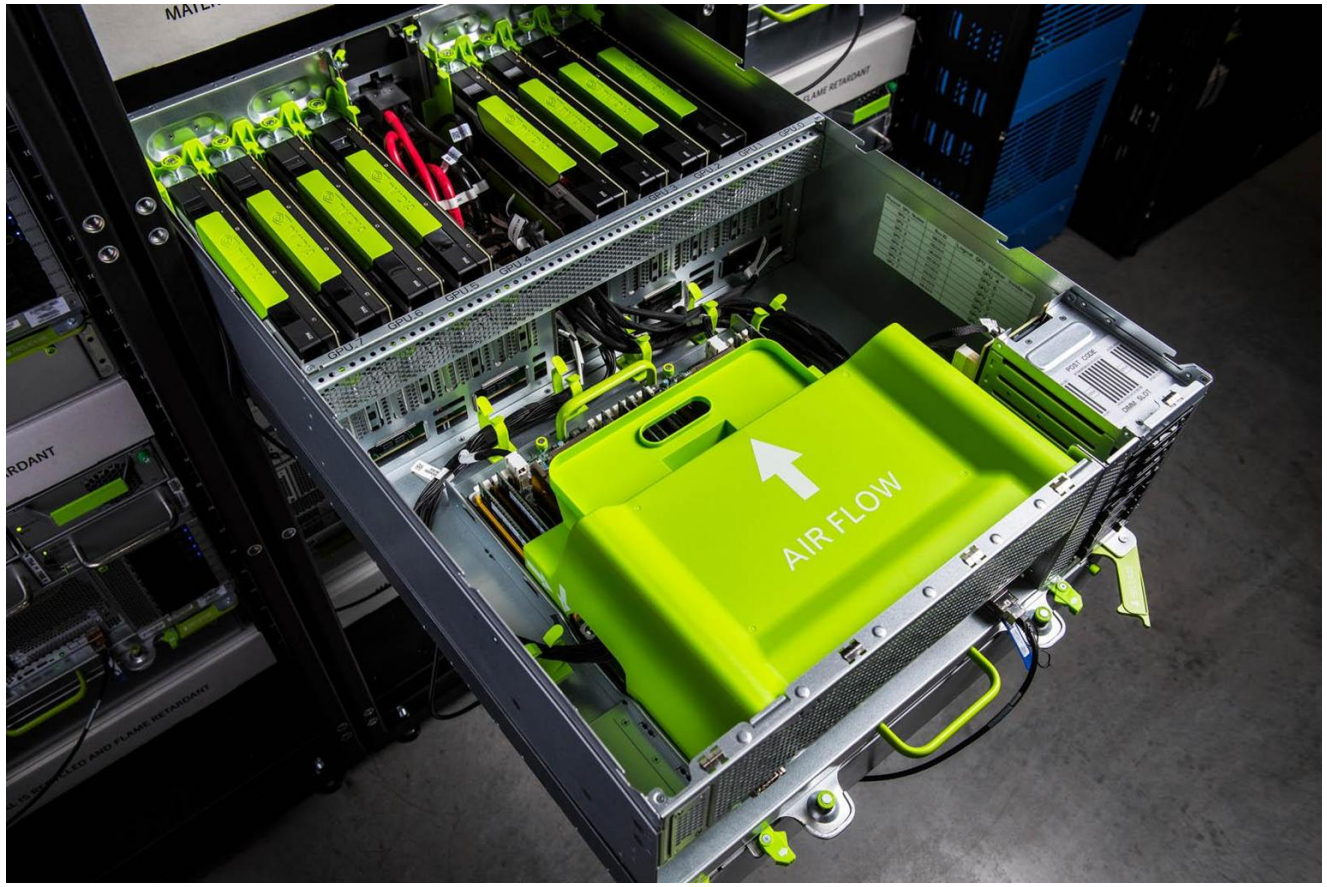
ImageNet Classification with Deep Convolutional Neural Networks

AI Model Building Blocks & Forms of Parallelism



RESNET50: Image Classification

- 8 PCIe® attached Nvidia GPUs
- Example system Facebook “Big Sur” (Dec 2015)
- K80 = 2 Kepler ASICs on a 300W PCIe card
  Per GK210 ASIC
  - TSMC 28nm
  - 561 mm^2
  - 7.1 B transistors
  - 824 MHz boost clock
  - 13 SMs
  - 4.1 TFLOPs FP32
  - 240 GB/s local GDDR5 BW

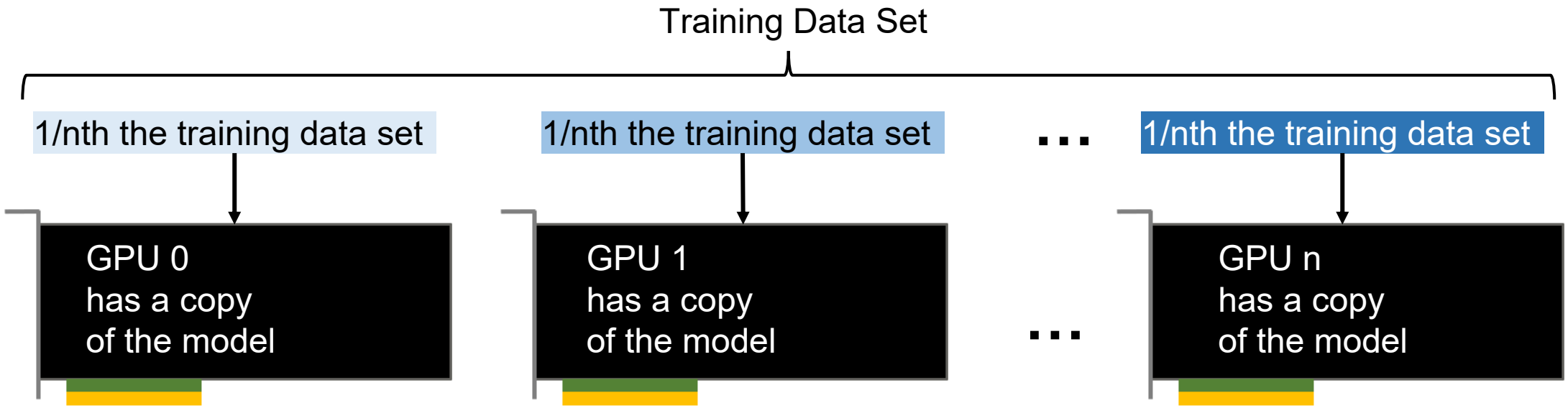


‘Big Sur’

‘Big Sur’ Hardware | Meta

Data Parallelism

- Example: a copy of the model fits in a single GPU
- But the computational throughput of the GPU is low relative to the computation required by the model crossed with the data set required to train the model
- Data parallelism enables a model to be trained faster by having multiple copies of the model train on different sets of data in parallel
- A reduction “AllReduce” is required so that every copy of the model has the modified weights at the end of each pass

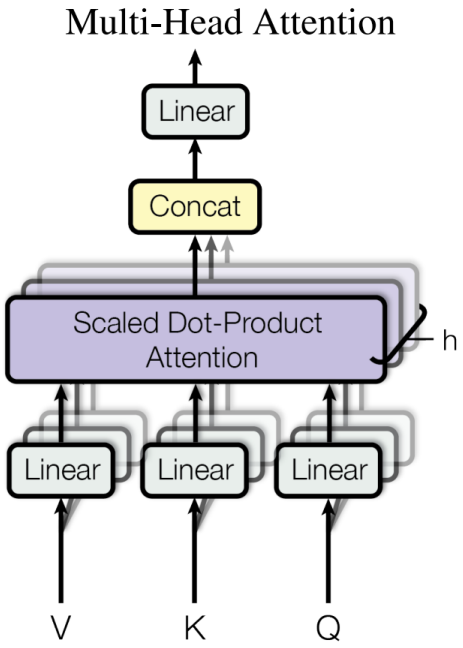


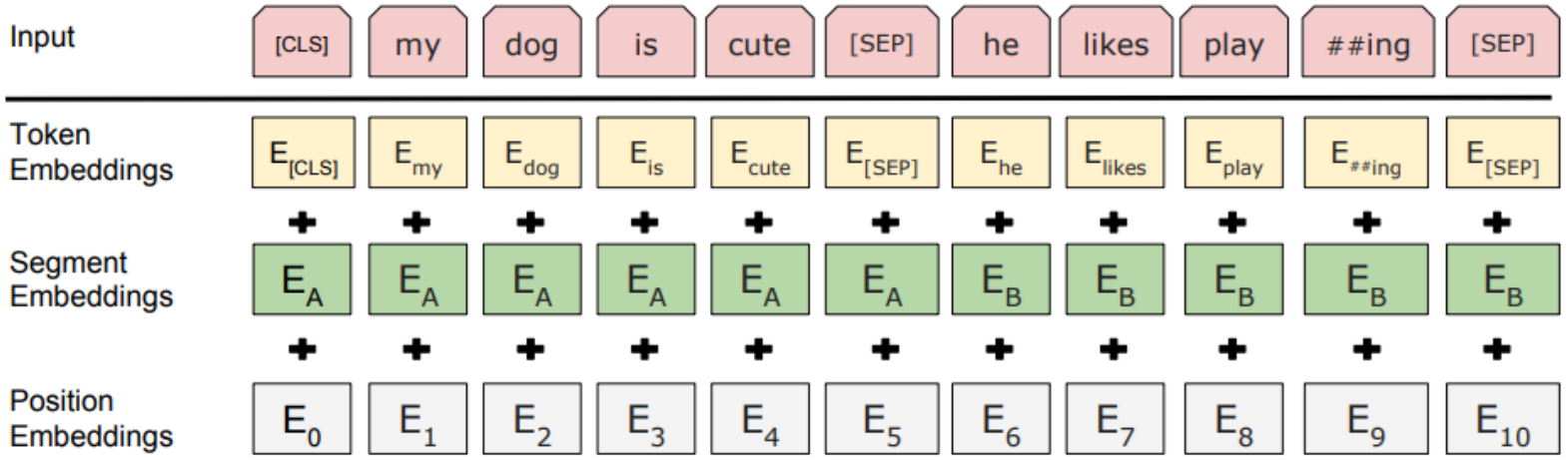
Natural Language Processing (NLP) & Transformers

BERT large: 2018 Year of the Transformer

- Bidirectional encoder representations for transformers (BERT)
- Google® TPUv2: four SOCs per board
- BERT large Training time
  - 64 TPUv2 ASICs = 4 days

"BERT is designed to pretrain deep bidirectional representations from **unlabeled text** by jointly conditioning on both left and right context in all layers"



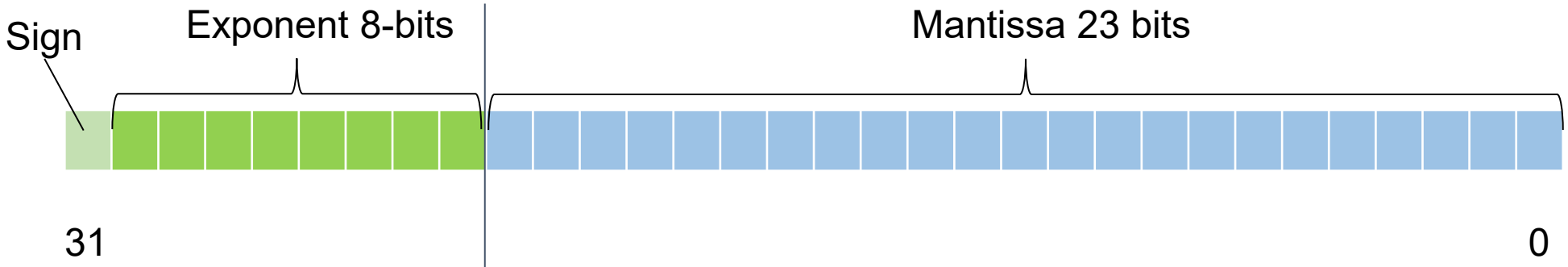


1706.03762

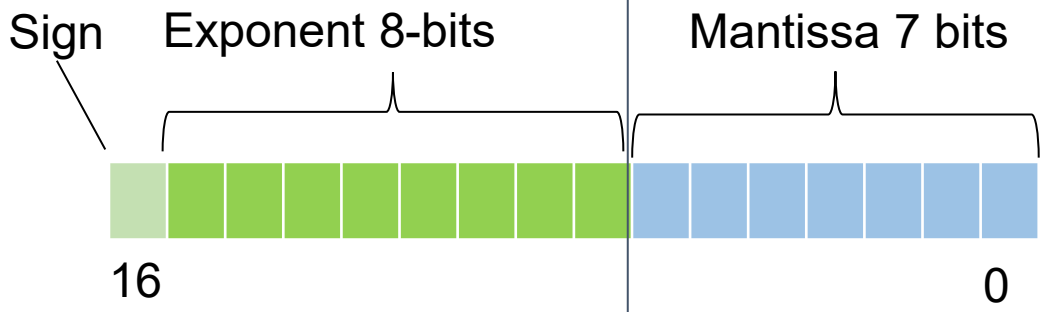
[1810.04805] BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Reduced Precision Floating Point

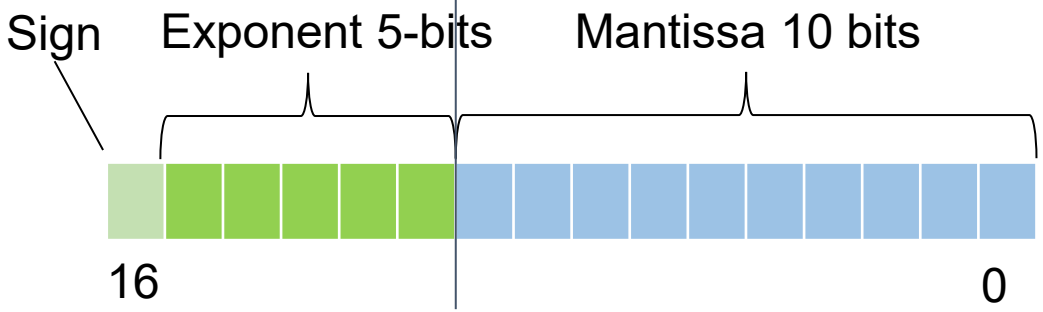
- FP32



- **TPUv2 added BF16** Same range as FP32, and sufficient precision for state-of-the-art AI model training



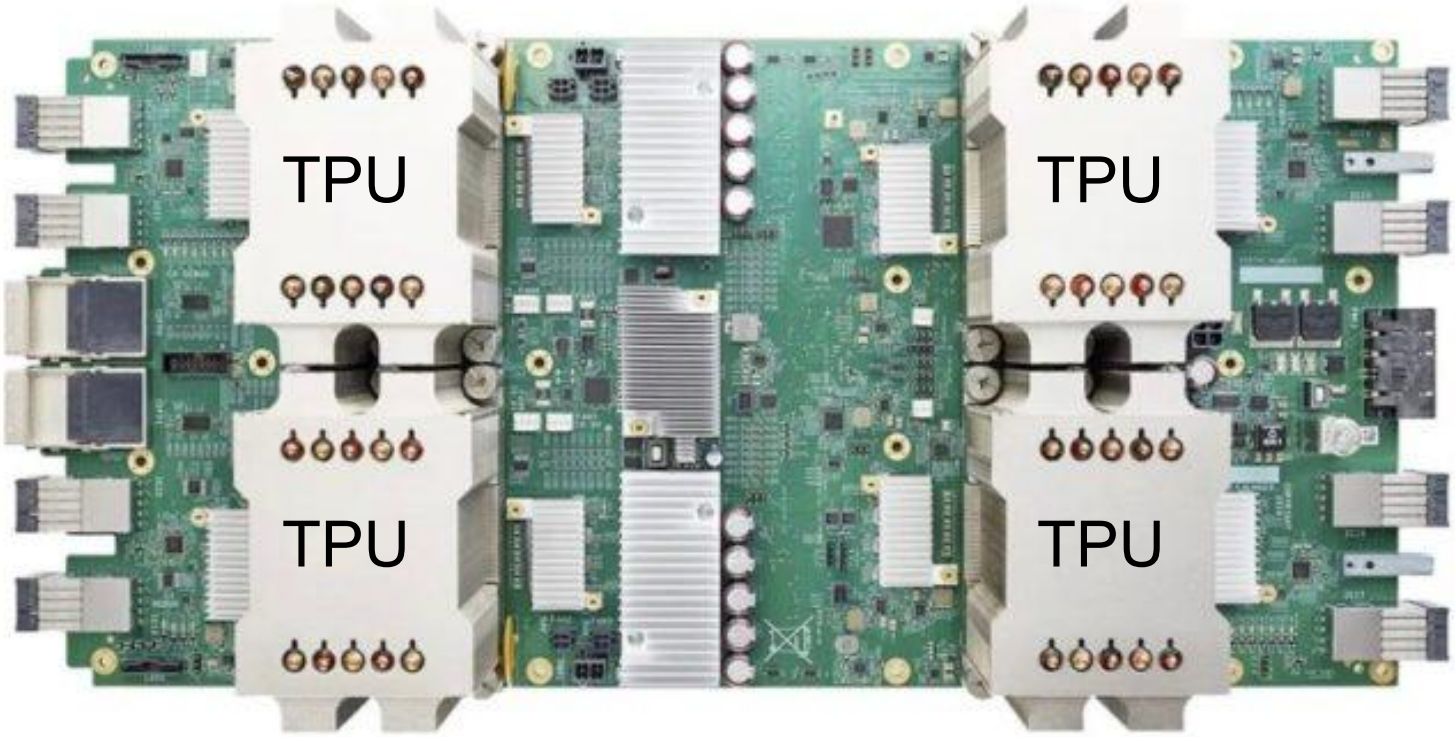
- **V100 added "Tensor Cores"** which are just more ALUs **optimized for FP16 matrix math** throughput per clock



Note: With TPUv1 Google® got int8 working for inference ~2015

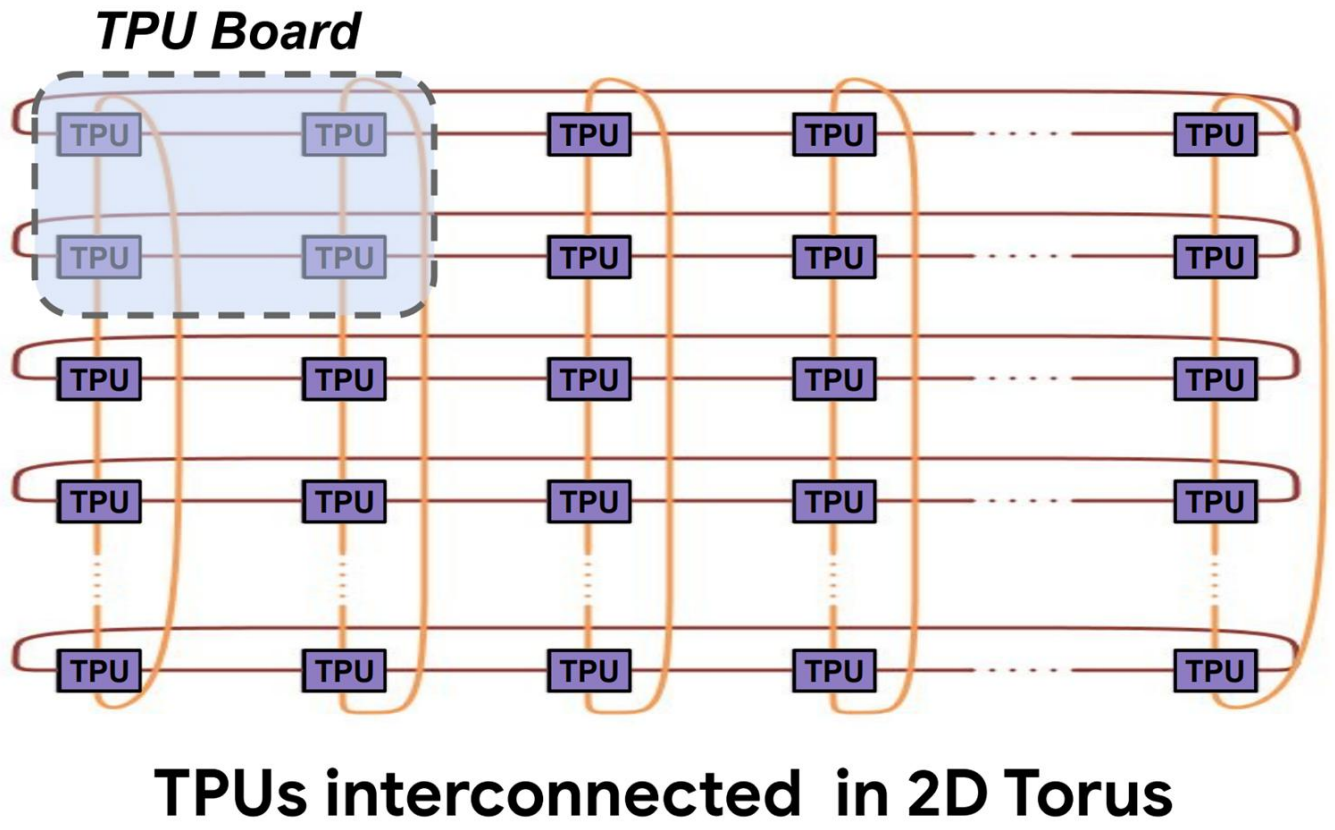
BERT large: 2018

- Google® TPUv2: four SOCs per board



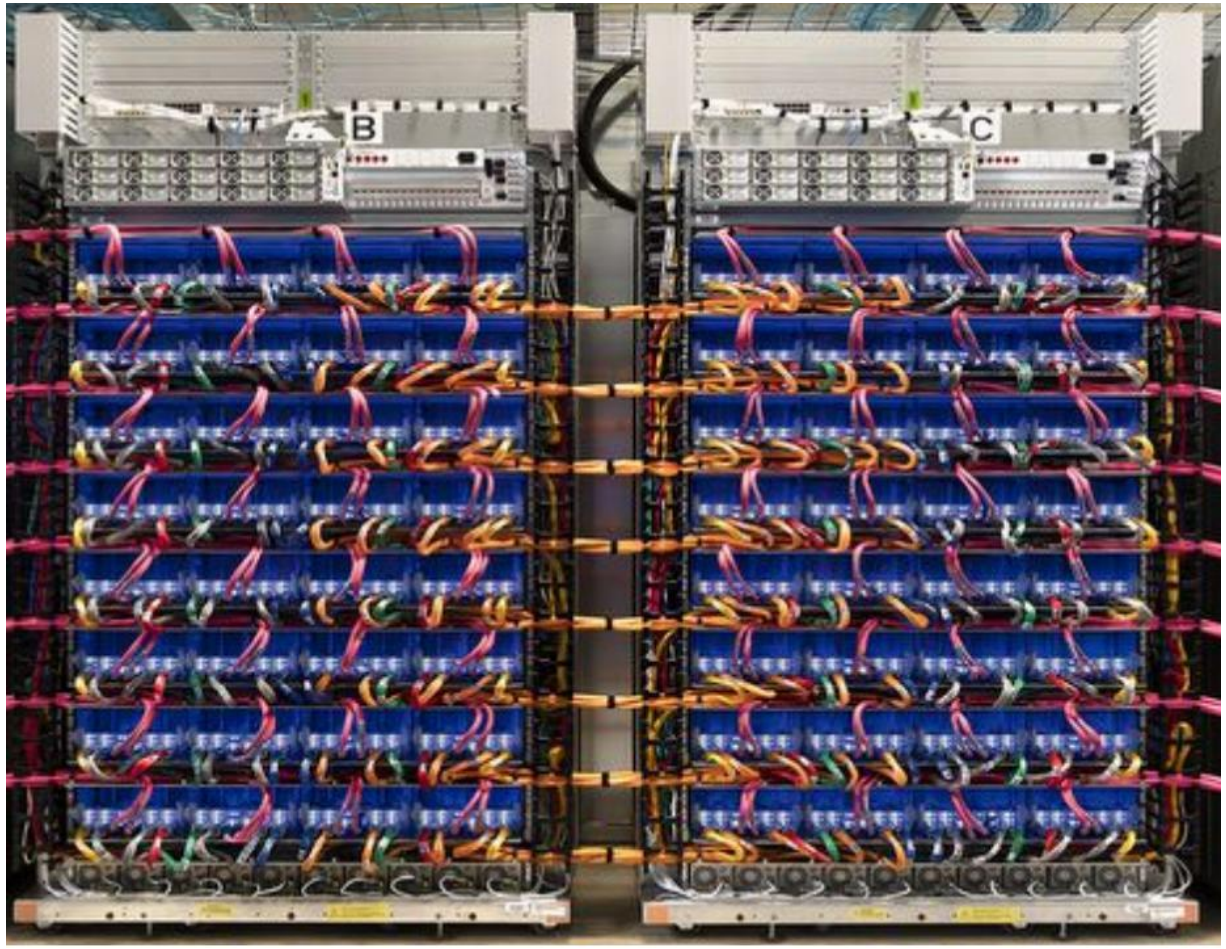
BERT large: 2018

- Google® TPUv2 Multiple Boards connected in a 2D Torus
- 4 ICI links per TPUv2 ASIC
- 4*496Gbps = 248 GB/s ICI BW per TPUv2
- 3 TFLOPs FP32 per ASIC
- 46 TFLOPS BF16 per ASIC
- Google focus became BF16

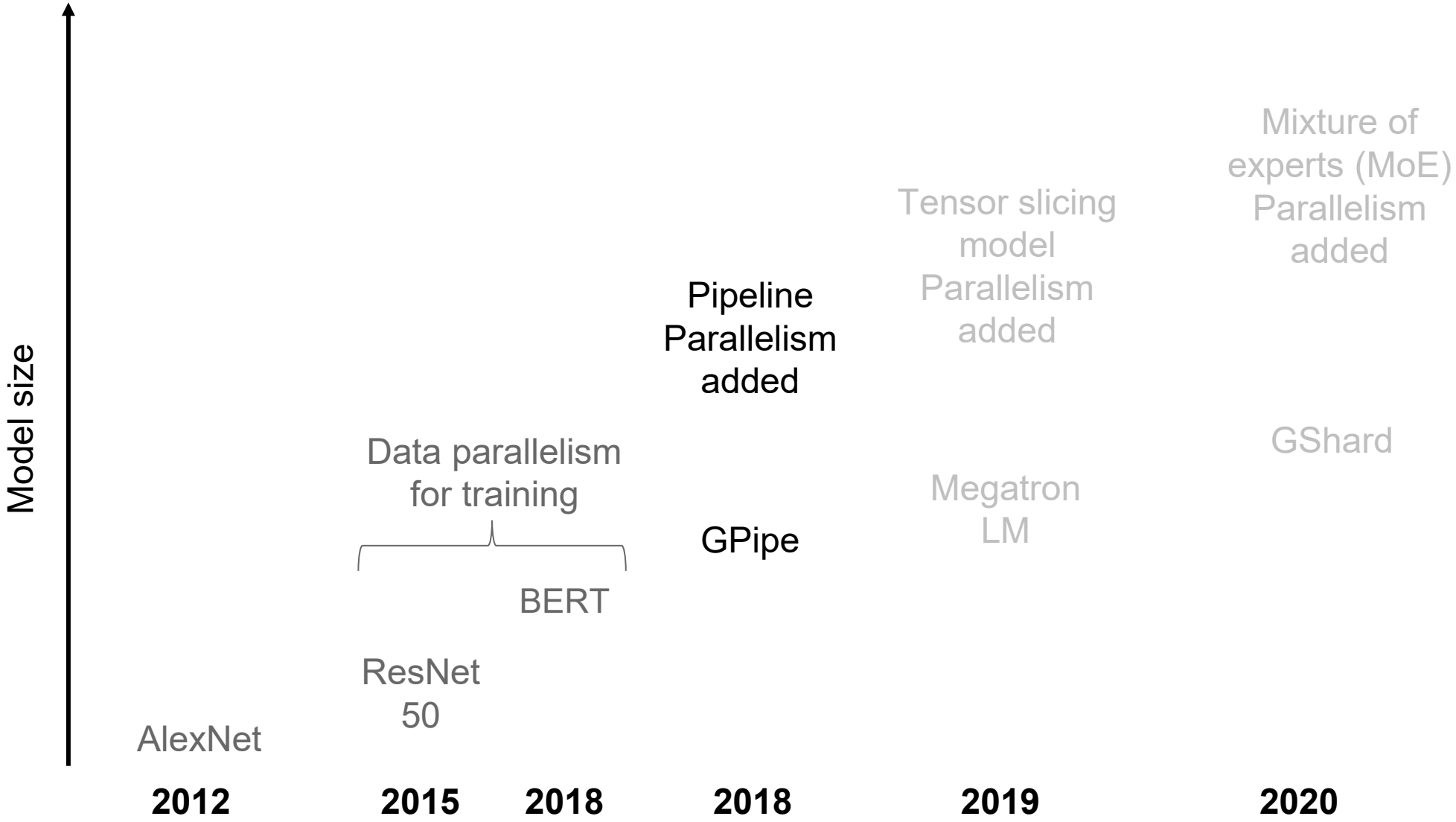


BERT large: 2018

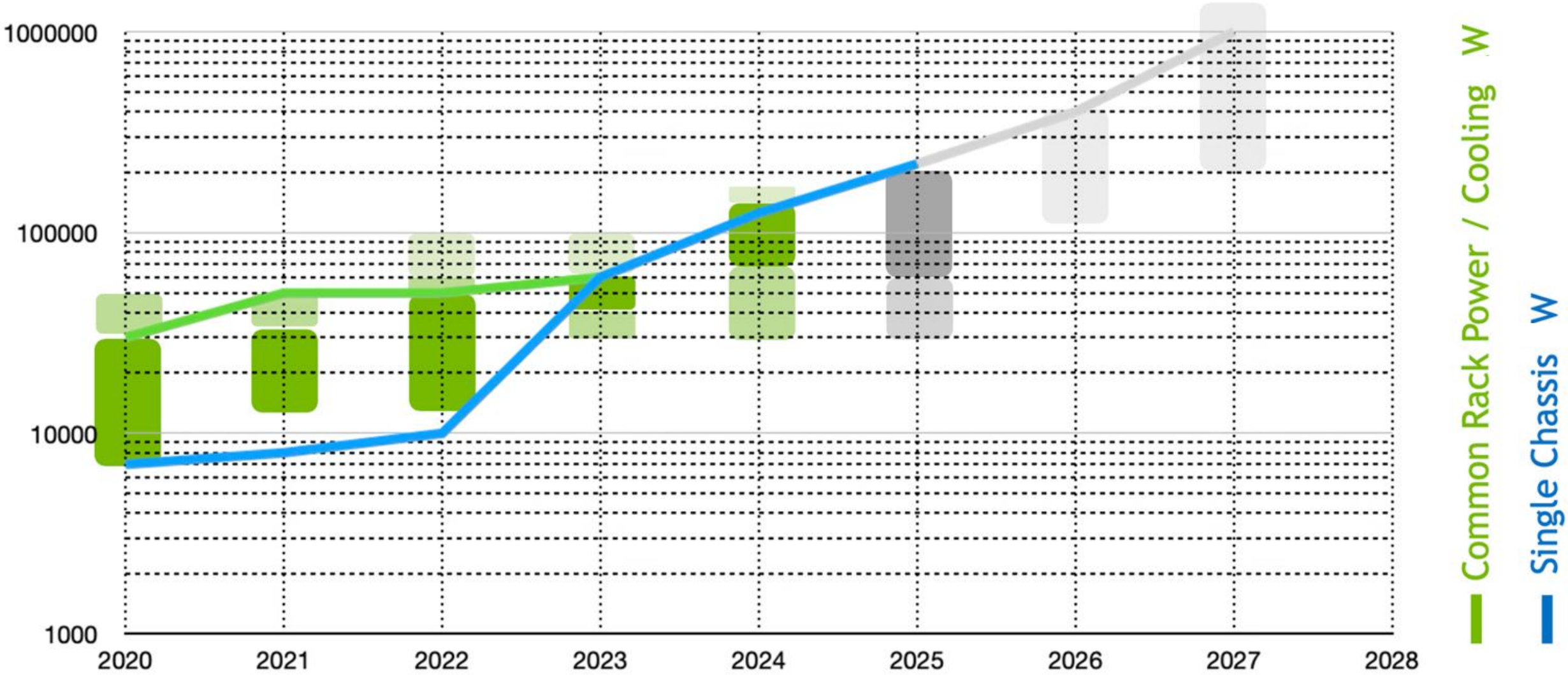
- 256 TPUv2s connected in a 2D Torus across 2 racks
- 64 TPU SOCs used to train BERT large
- BERT large training time
  - 64 TPUv2 ASICs = 4 days



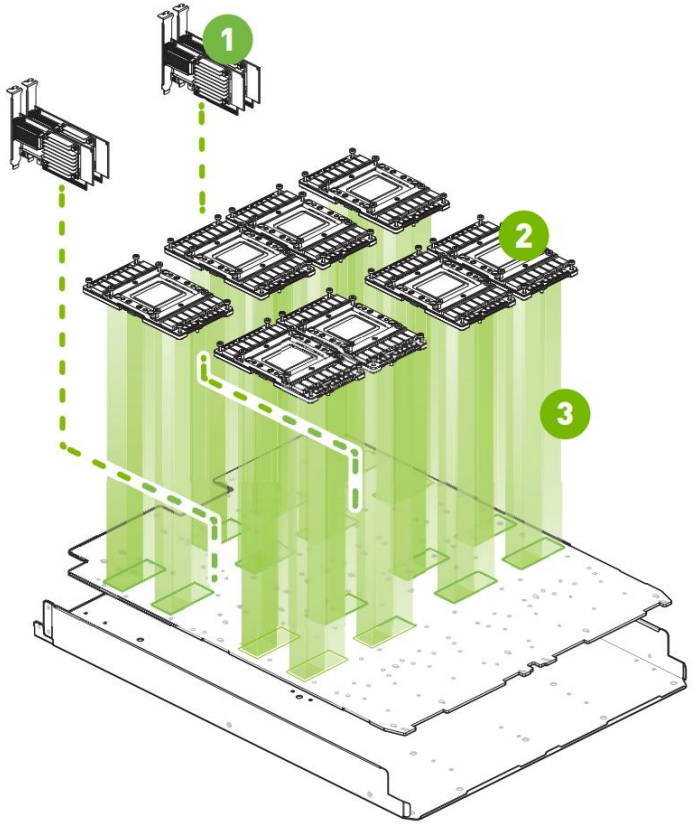
AI Model Building Blocks & Forms of Parallelism

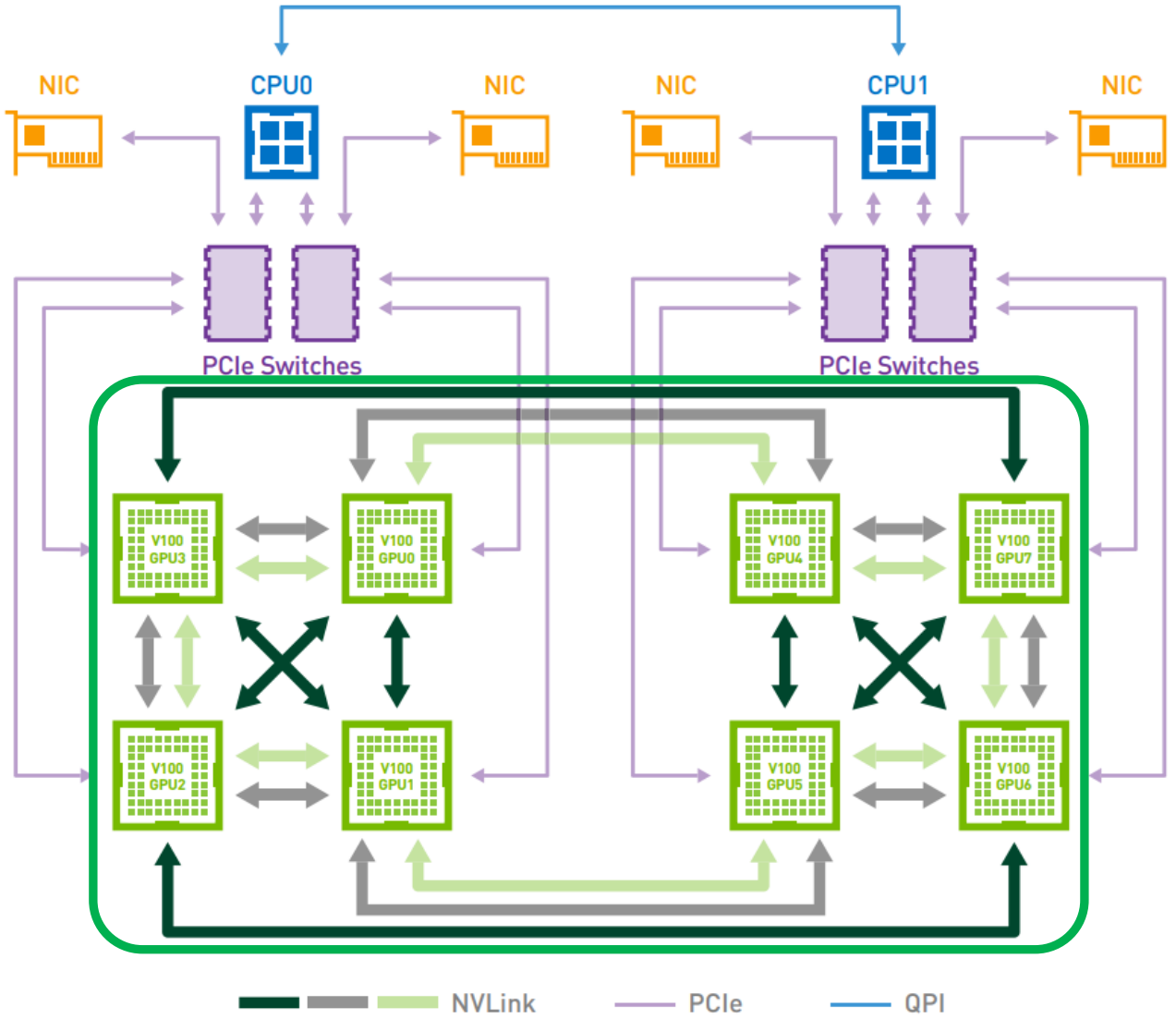


Common Hyperscale Rack Power Available



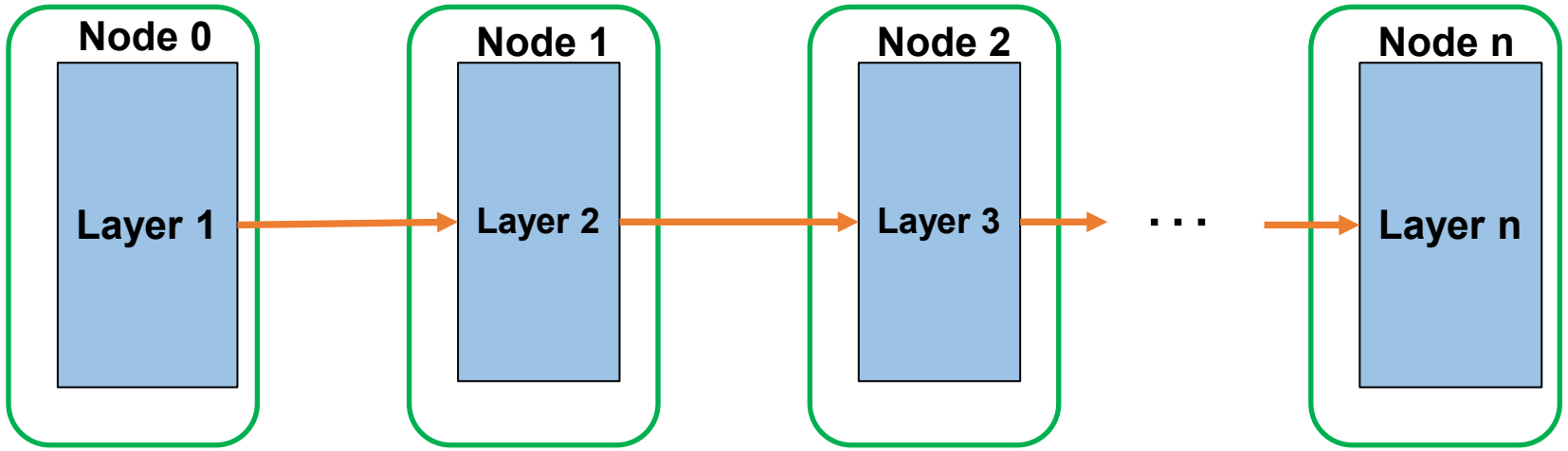
DGX-1 (V100 systems)



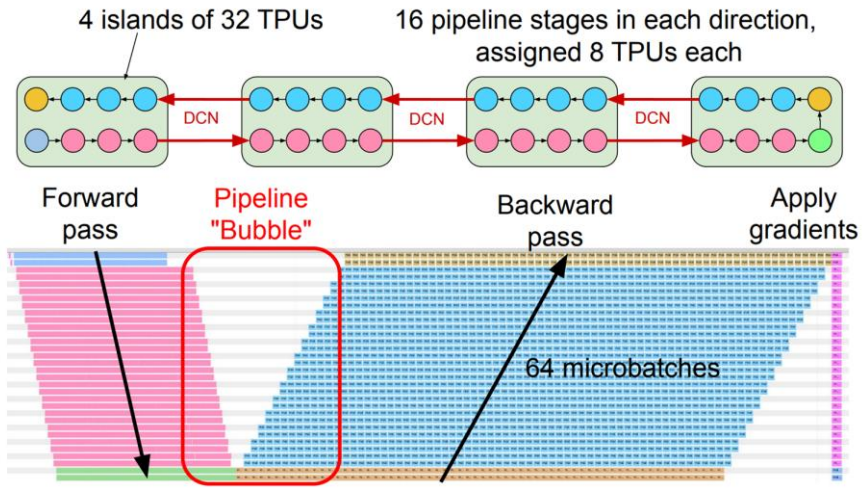


NVIDIA DGX-1 with Tesla V100 System Architecture White paper

Pipeline Parallelism



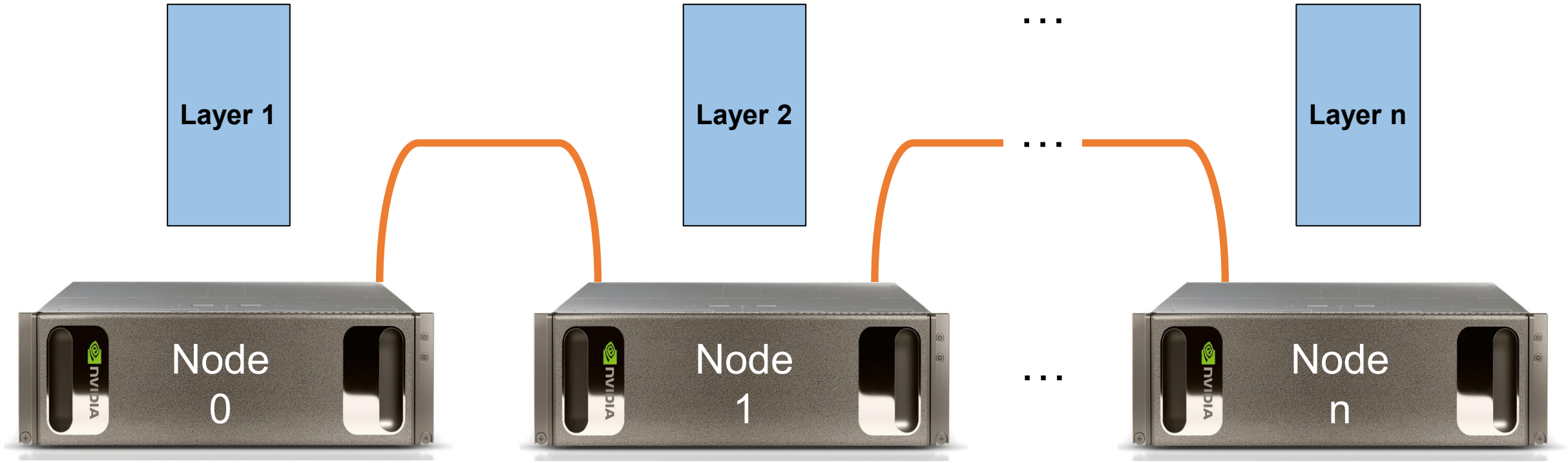
- Have multiple nodes each work on one or more layers of a network in a pipeline
- Used in existing systems because inter-node communication BW is a factor of 1/8th or less the in-node GPU to GPU communication BW
- 0.xN-Factor performance improvement (where N = # of GPUs)



https://arxiv.org/pdf/2203.12533

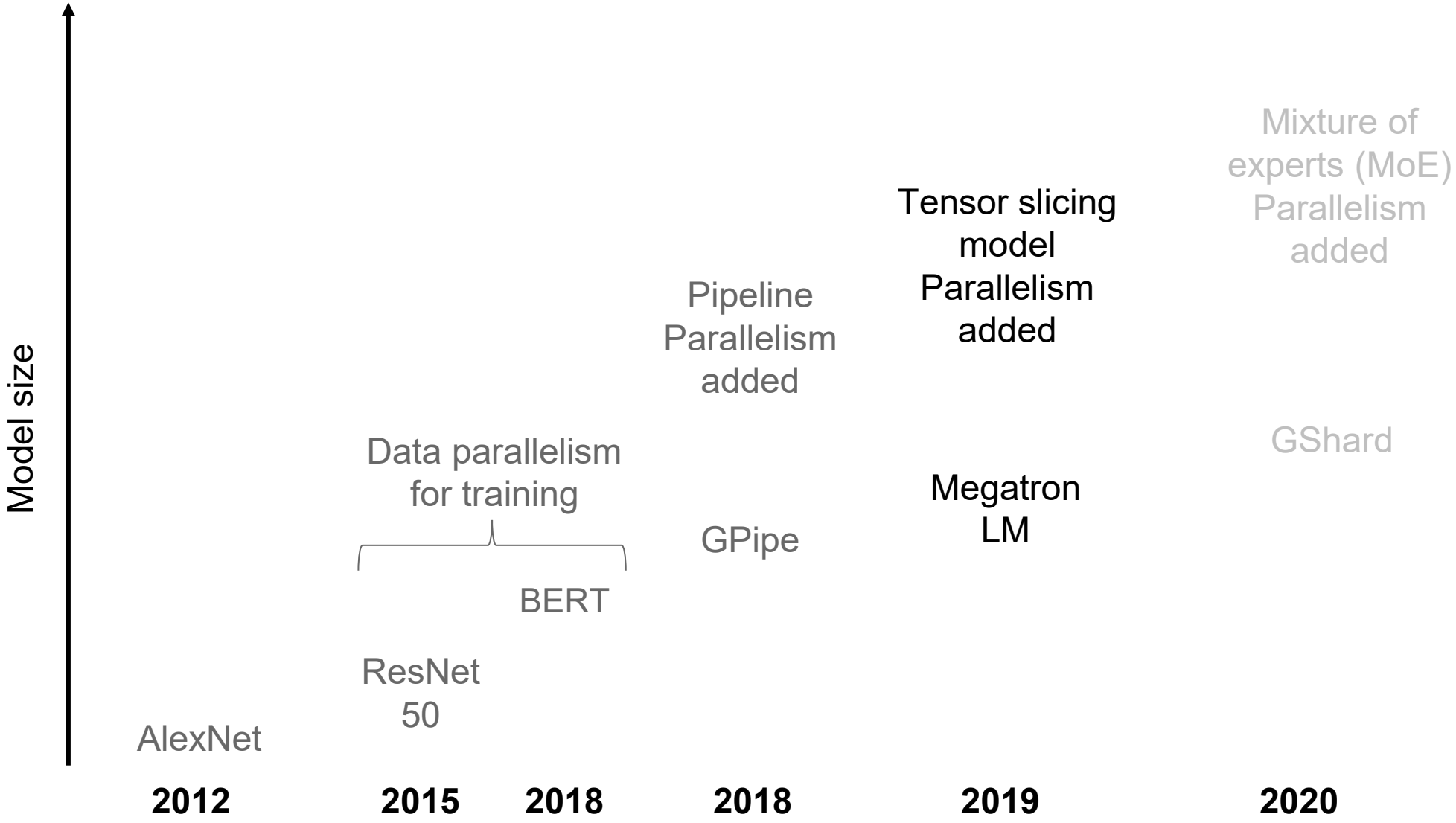
Pipeline Parallelism: with DGX-1 (V100 systems)

Example



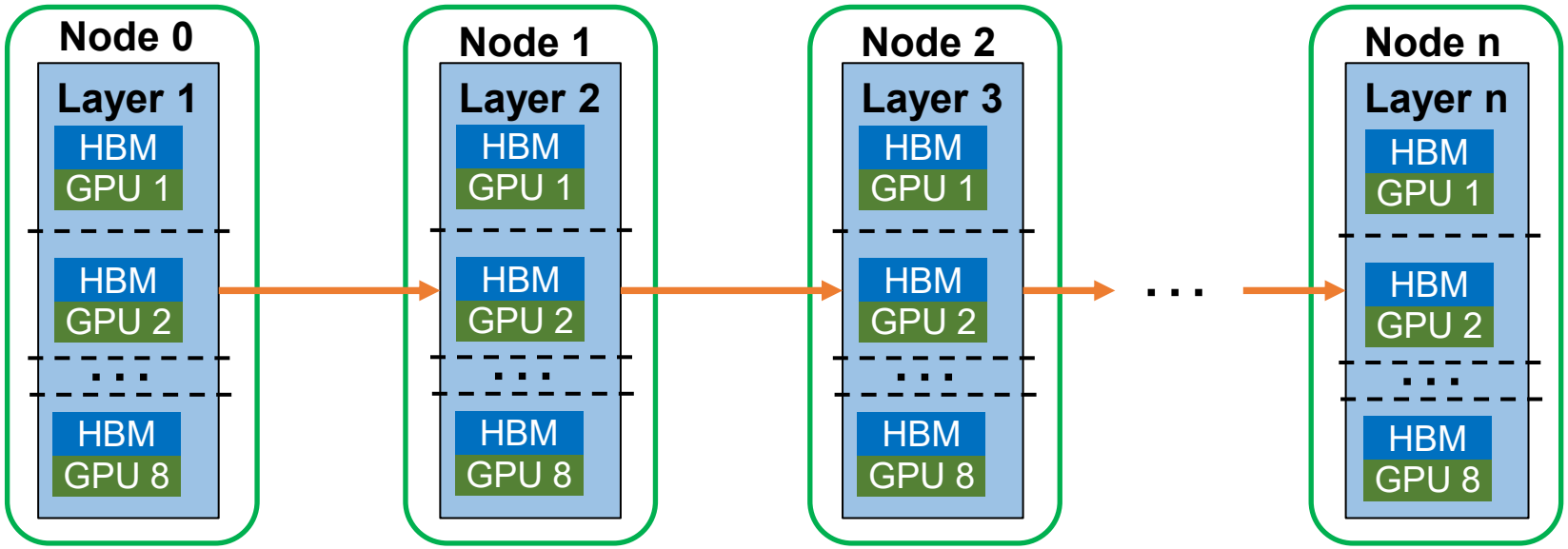
Strong Scaling as more compute can be applied to the model without increasing global batch size

AI Model Building Blocks & Forms of Parallelism



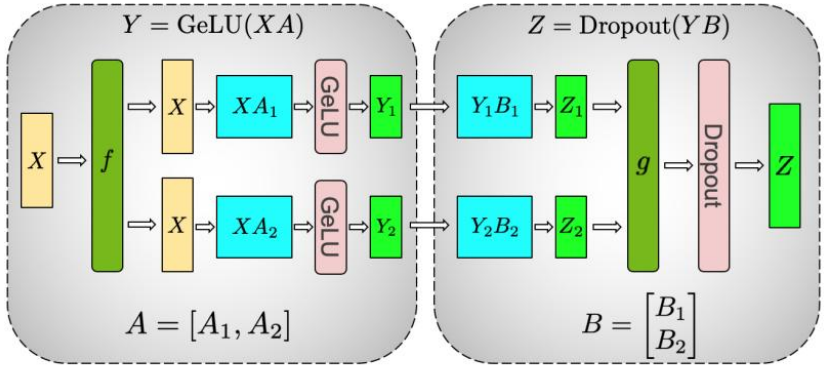
Tensor Slicing Model Parallelism

(Also referred to as Horizontal Parallelism)



- Slice a model's matrix math across GPUs in a node
- Strong scaling
- ~N-Factor performance improvement (where N = # of GPUs)
- AllReduce Communication pattern
  - Blocking communication

Tensor Slicing Model Parallelism Example

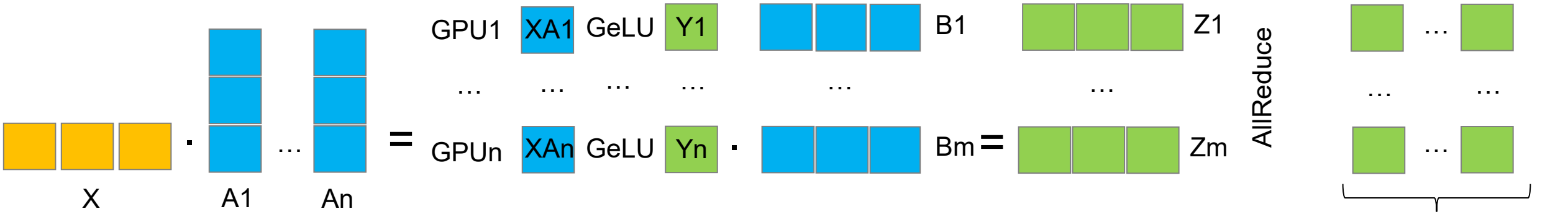


(a) MLP

A and B are the weight matrix

X it the input vector

Y is the output vector



Weight matrix column wise split across GPUs

Weight matrix row wise split across GPUs

Every GPU has a copy of n x m fully reduced activation matrix

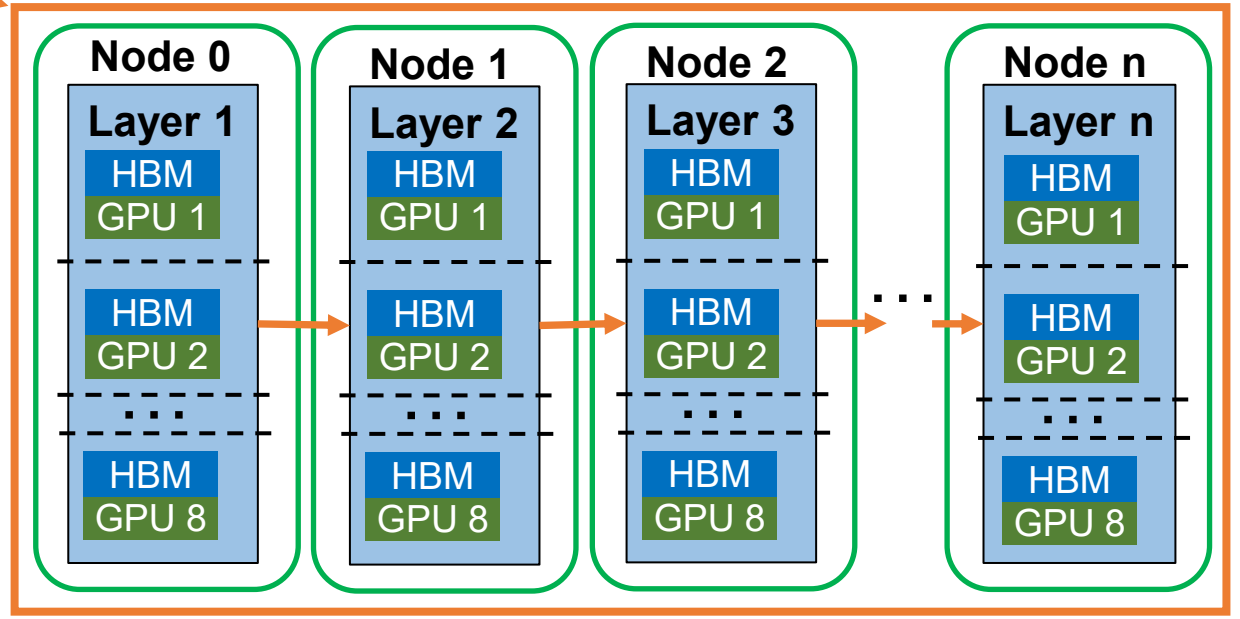
- Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism (arxiv.org)
- Model Parallelism — transformers 4.7.0 documentation (huggingface.co)

Forms of Parallelism We've Discussed So Far

	What	Blocking Communication	Where	Per GPU package Bandwidth and latency requirements (2H 2026)
Tensor Slicing Model Parallelism (Horizontal Parallelism)	Matrix math split across GPUs Not complete until communication between GPUs occurs	Yes	Generally, Scale-up domain High Bandwidth, low latency	1.8 TB/s uni-dir BW < 1us pin to pin
Pipeline Parallelism	Output of one layer of a network propagated to the next layer	Yes	Generally, ethernet or Infiniband™ connecting systems	200 GB/s uni-dir BW > 2us pin to pin
Data Parallelism	AllReduce communication across all model replicas being used to train the model	Effectively No	ethernet or Infiniband All interconnect hierarchies connecting GPUs used to train a model	200 GB/s uni-dir BW to > 10 GB/s < 10 us to milliseconds

Model replica

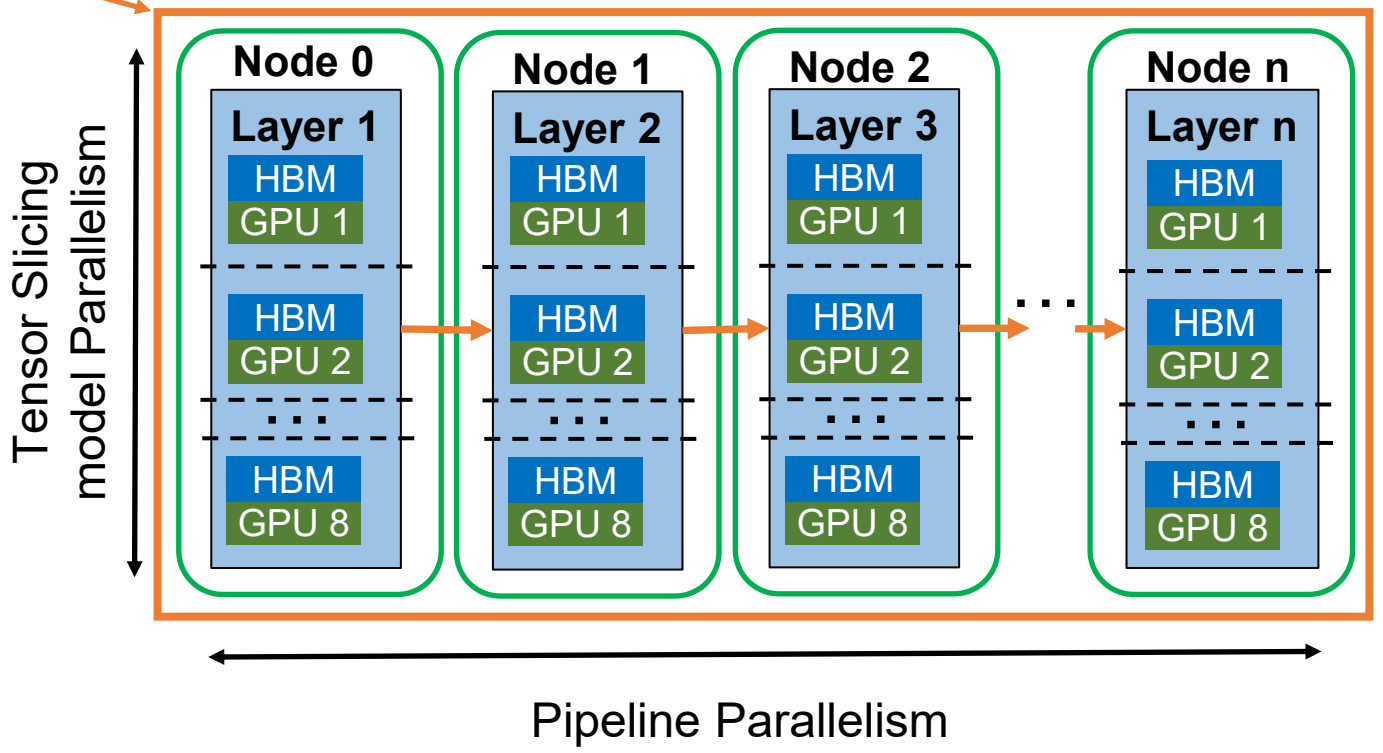
- Each “model replica” composed of interconnected nodes of GPUs holds one copy of the model



Model replica

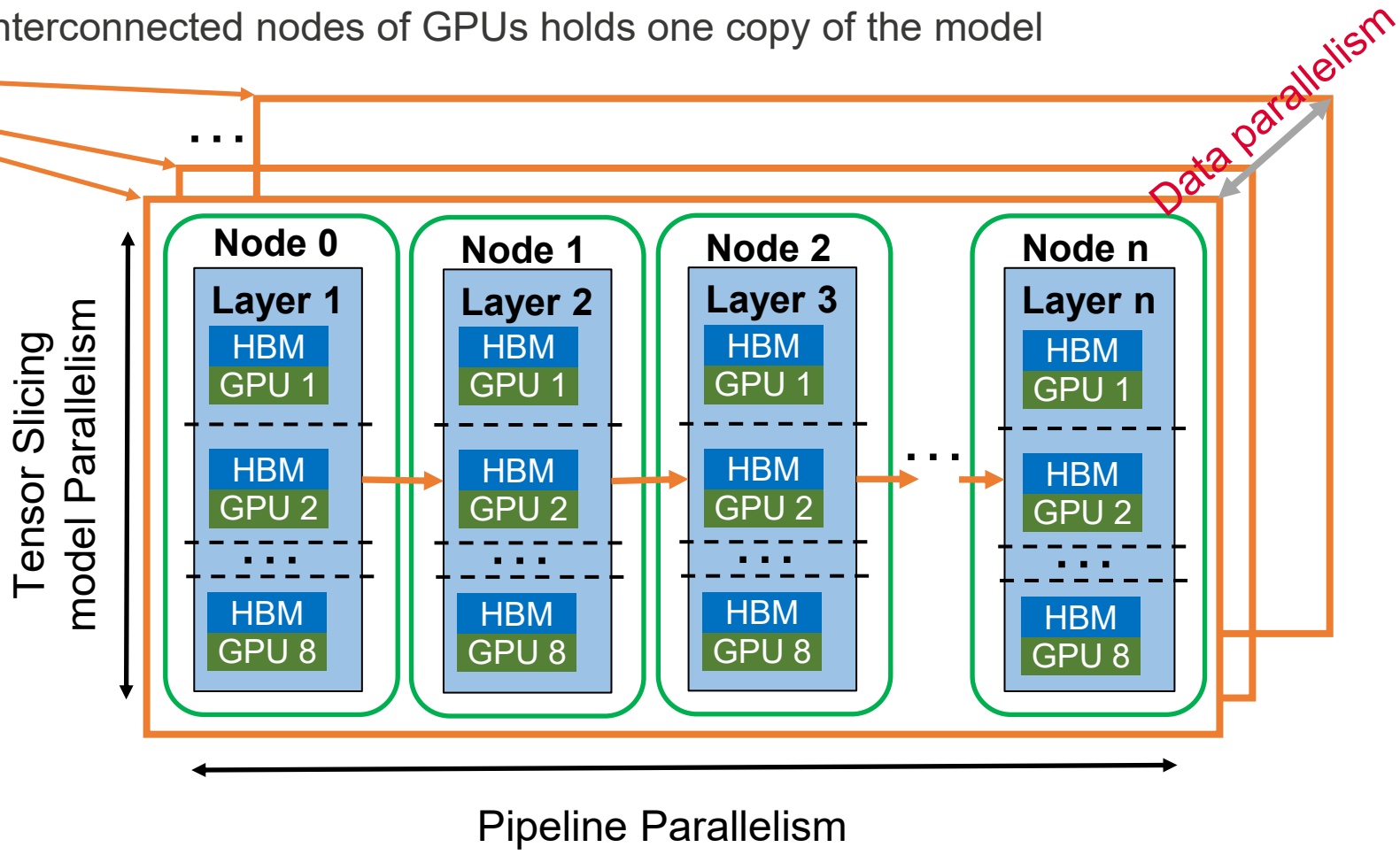
- Each “model replica” composed of interconnected nodes of GPUs holds one copy of the model

- Strong scaling with # of GPUs within each model replica



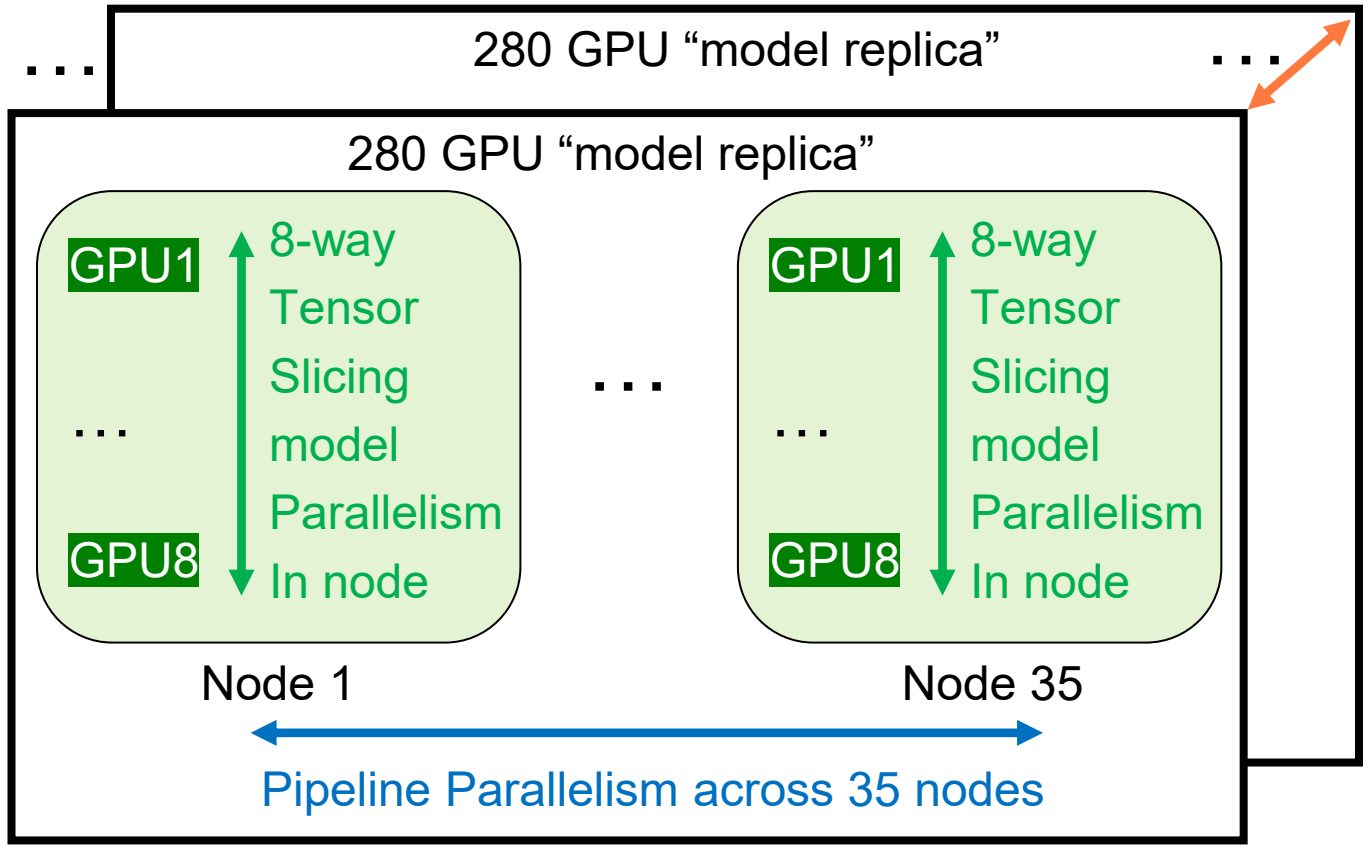
Model replica

- Each “model replica” composed of interconnected nodes of GPUs holds one copy of the model
- Strong scaling with # of GPUs within each model replica
- Data parallelism across model Replicas



2021: 530B Parameter Megatron Turning NLG

- Microsoft® / NVIDIA 530Billion Parameter Megatron-Turning NLG using 280 GPU “model replicas”
- 530B parameters of FP16 = 1.06 TB
- 105-layer network
- Pipeline Parallelism across 35 nodes
  - 1.06TB / 35 nodes = 30.3 GB/node
  - 3 layers / node = 10.1 GB / layer
- 8-way Tensor slicing per node
  - 1.26 GB weights / GPU / layer
- Data Parallelism across
- 8, 10, and 12 model replicas
- Iteration times
  - 60.1, 50.2, and 44.4 seconds
- Teraflops / GPU
  - 126, 121, and 113



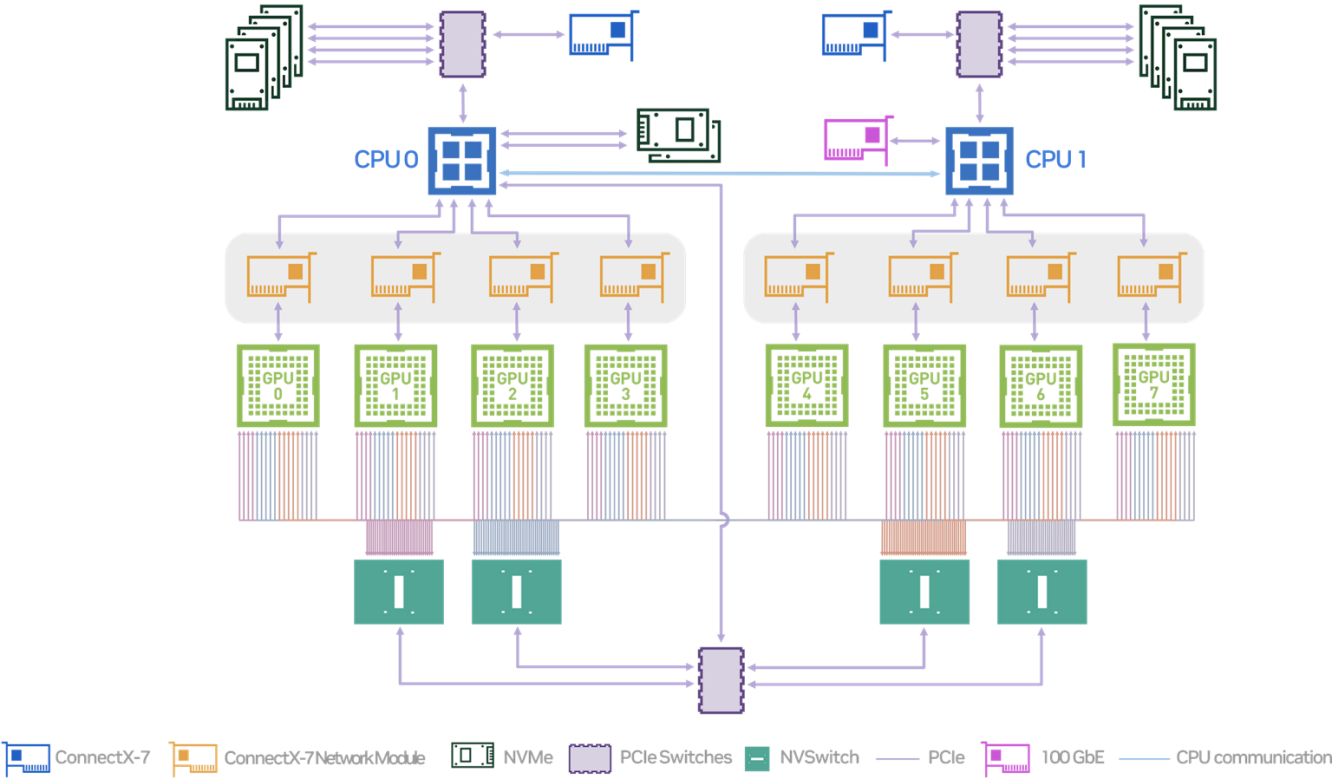
Using DeepSpeed and Megatron to Train Megatron-Turing NLG 530B, the World’s Largest and Most Powerful Generative Language Model | NVIDIA Technical Blog

Using DeepSpeed and Megatron to Train Megatron-Turing NLG 530B, the World’s Largest and Most Powerful Generative Language Model - Microsoft Research

DGX & HGX H100 (2H 2022)

DGX H100 systems

8 NVLink/Switch connected H100 GPUs



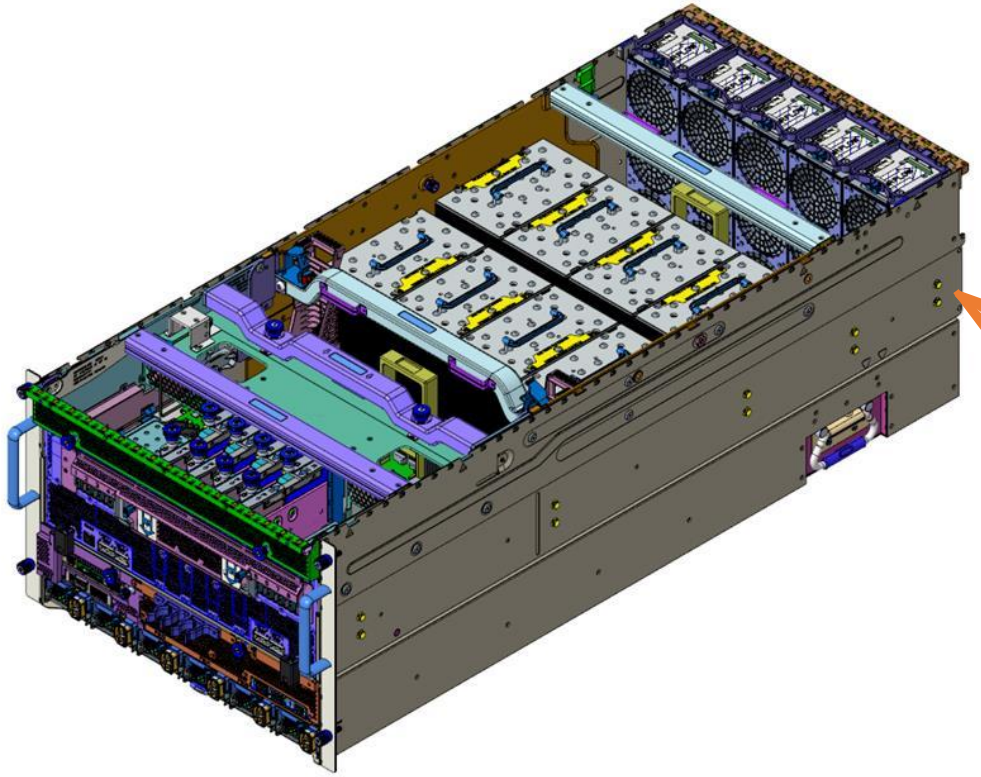
Nvidia HGX H100 Board (for OEM / ODM Systems)

Nvidia donated the HGX board specs to OCP

Introduction to NVIDIA DGX H100/H200 Systems — NVIDIA DGX H100/H200 User Guide

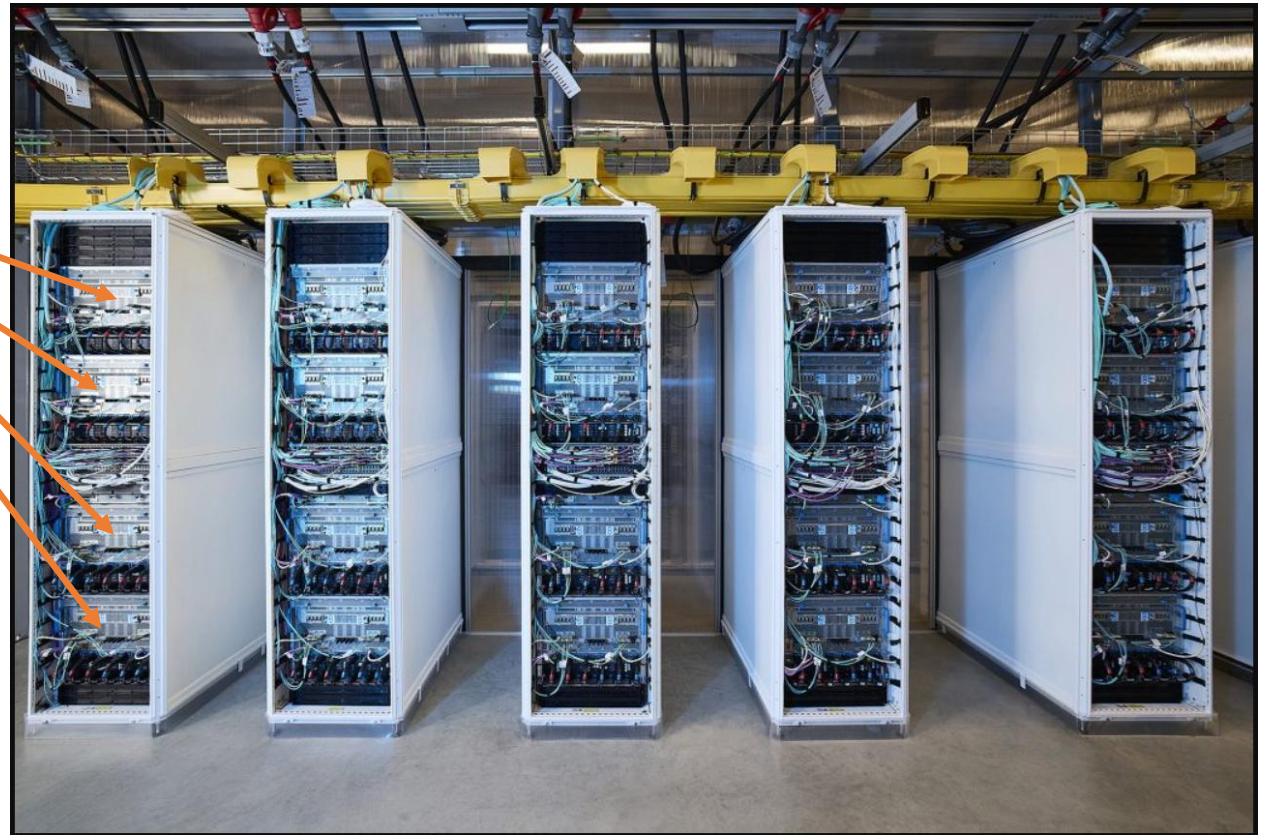
NVIDIA Contributes H100 Tensor Core-based HGX Baseboard Physical Spec to OCP OAI Project! » Open Compute Project

MI300X systems example (May 2024)



8 MI300X per chassis

4 Chassis per Rack = 32 GPUs per rack

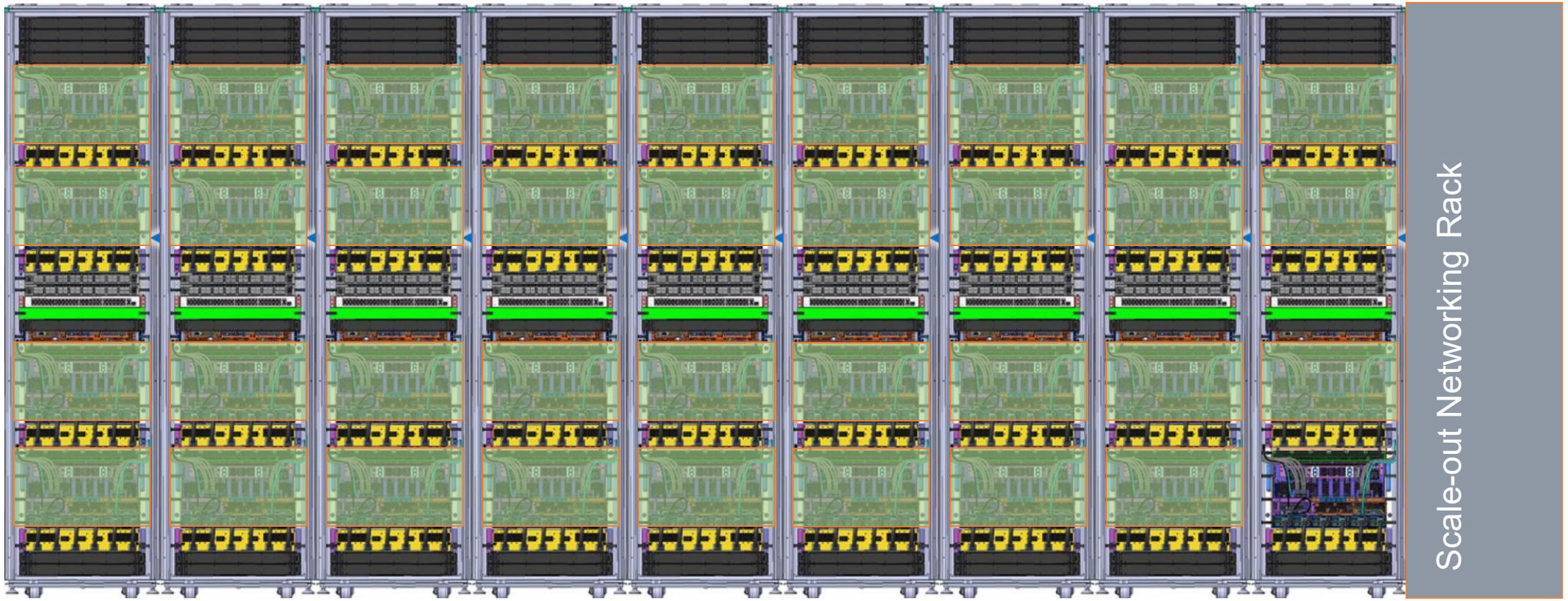


Azure® ND-MI300X (2024)

Cluster of racks example of a “model replica”

Per copy of the model if parallelism was mapped like “Megatron Turing NLG”

= Tensor slicing model parallelism high bandwidth Infinity Fabric™ interconnect within the UBB8

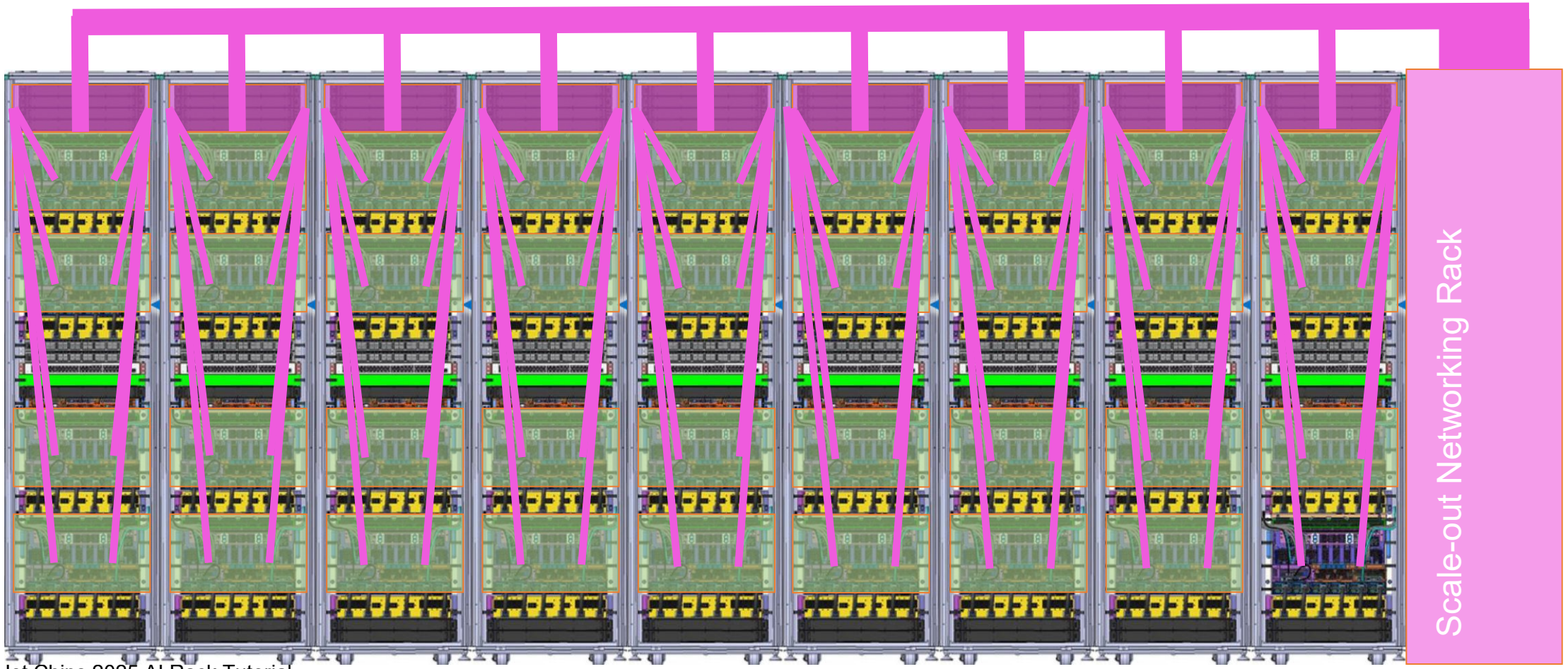


Cluster of racks example of a “model replica”

Per copy of the model if parallelism was mapped like “Megatron Turing NLG”

= Tensor slicing model parallelism high bandwidth Infinity Fabric™ interconnect within the UBB8

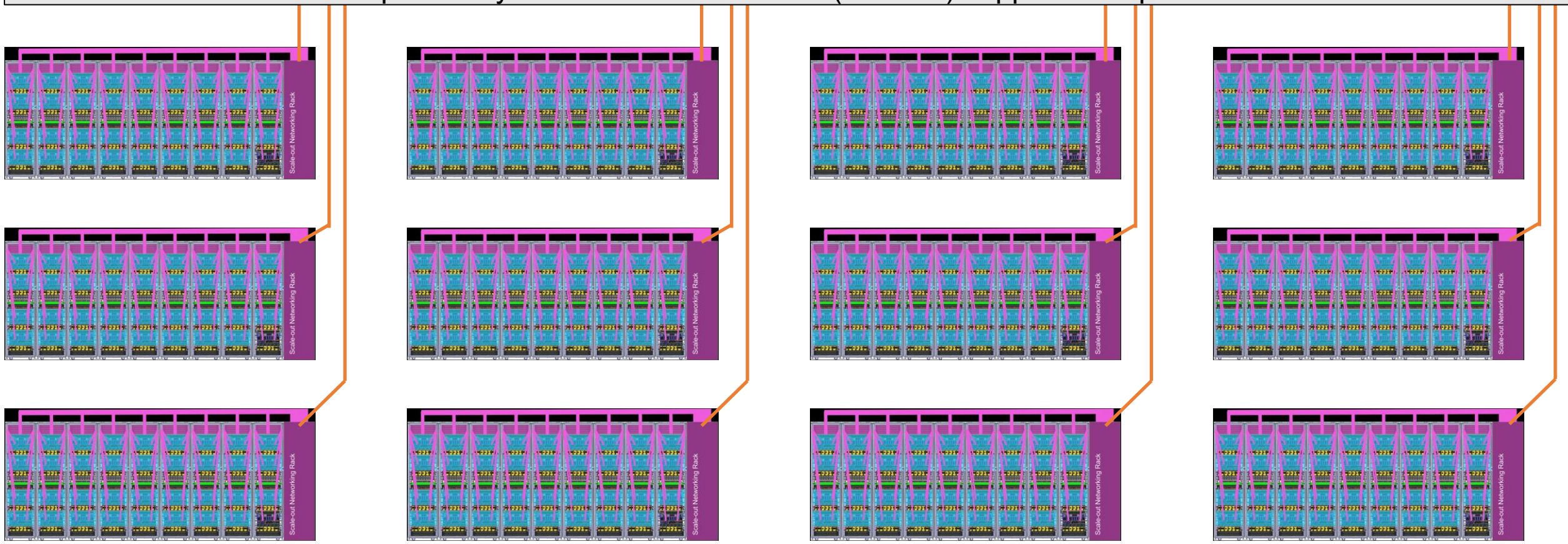
= Pipeline parallelism connects nodes (Systems) via scale-out ethernet (or InfiniBand™) NICs at 1/9th the scale-up Infinity Fabric Bandwidth per GPU



Cluster Level Example Across 12 “model replicas”

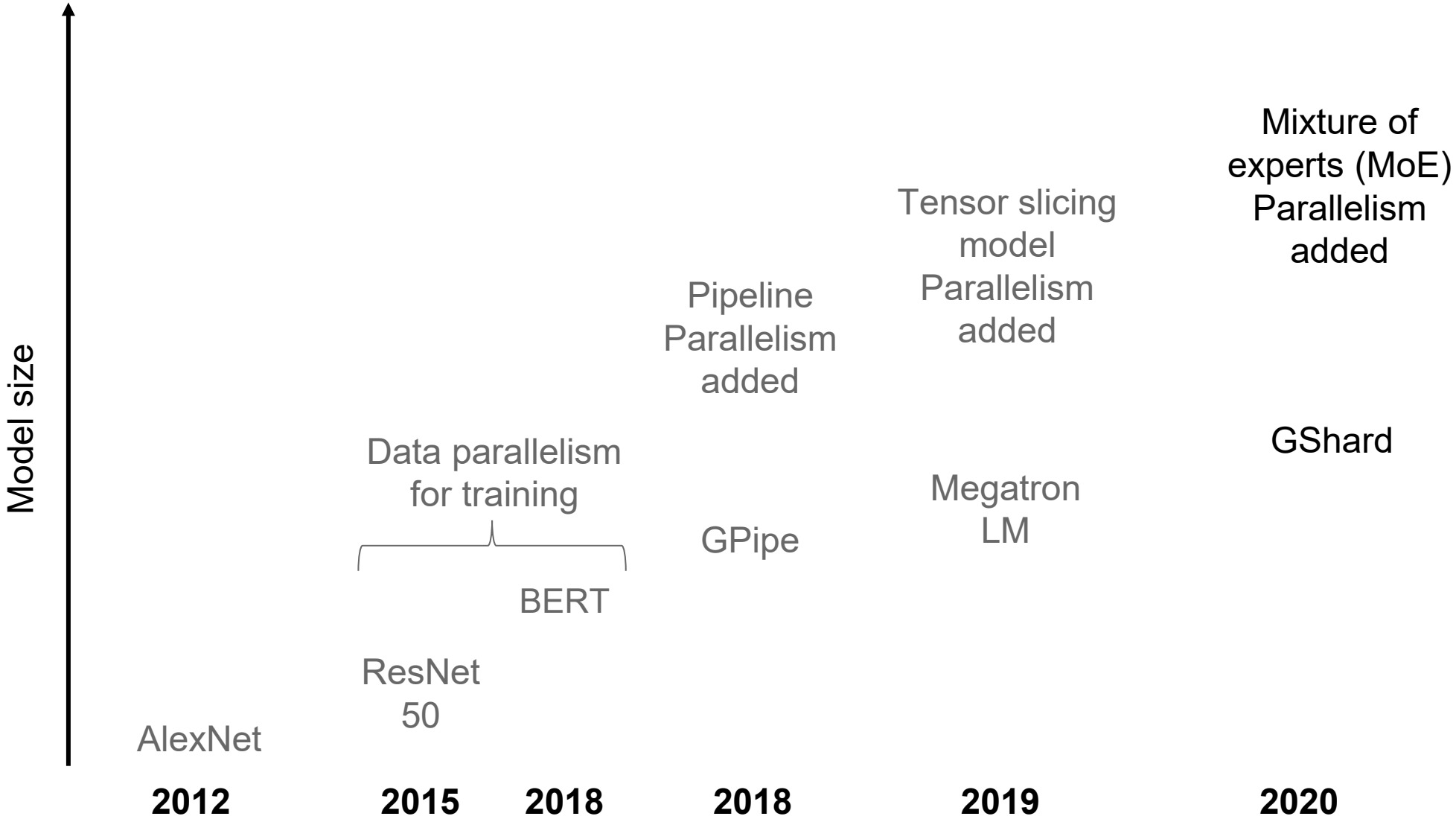
12 copies of the model = 12 * 280 GPUs = 3,360 GPUs

Example 3rd layer of scale-out switches (in racks) support data parallelism



Training time with A100s was 3 months! Who can wait that long? Thus, training clusters are getting larger

AI Model Building Blocks & Forms of Parallelism



Dense models vs. Mixture of experts (MoE) A Form of Sparsity

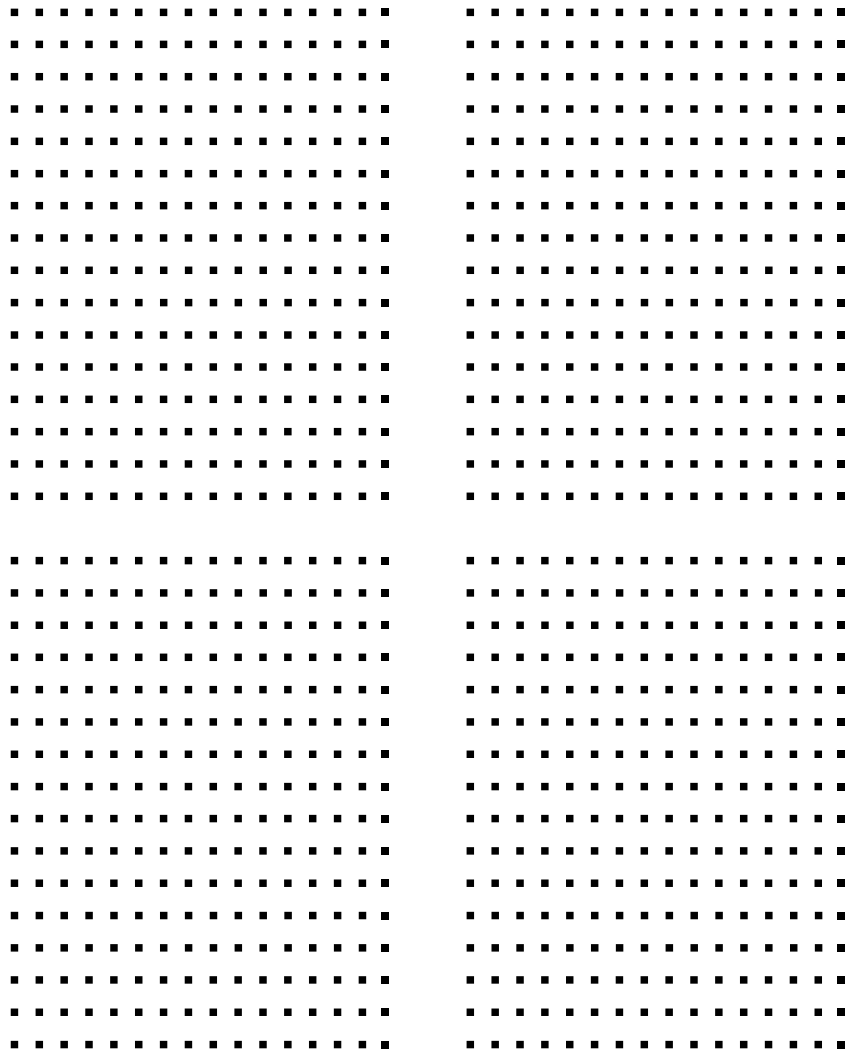
Dense models

- All GPUs implementing model parallelism have the full / same capability
- Thus 100% computational throughput is required with Dense models

Dense model Example: All Food

American, Chinese, British, French, German, Mexican, Indian, Italian, Japanese, Spanish, etc.

1024 GPUs. Each dot represents a GPU:



Dense models vs. Mixture of experts (MoE) A Form of Sparsity

Dense models

- All GPUs implementing model parallelism have the full/same capability
- Thus 100% computational throughput is required with Dense models

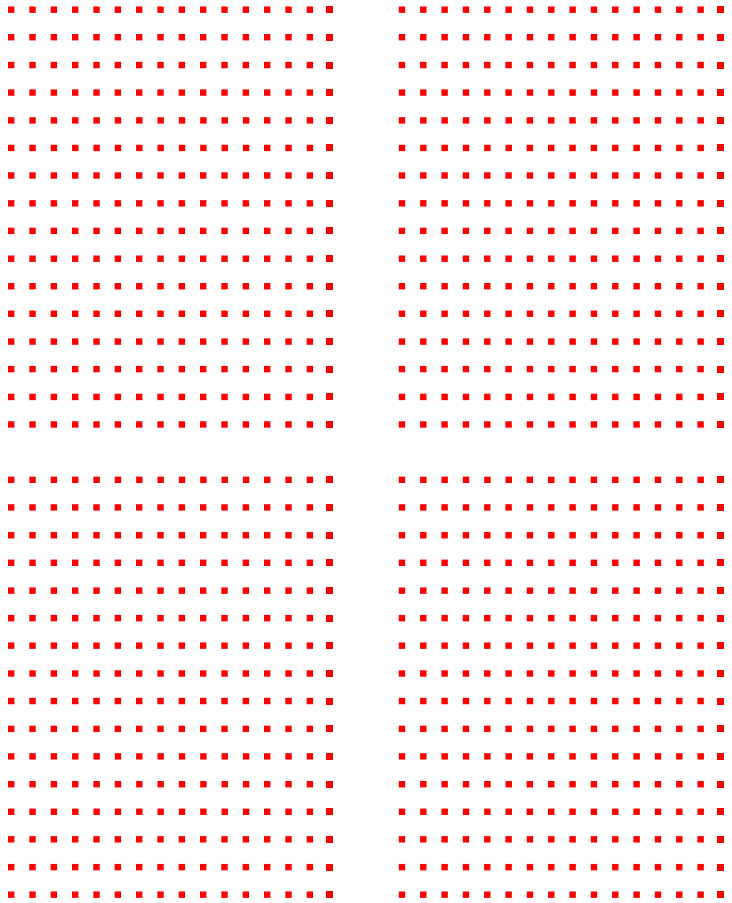
Dense model Example: All Food

American, Chinese, British, French, German, Mexican, Indian, Italian, Japanese, Spanish, etc.

Hamburger Computational Cost

100% Computation Cost

1024 GPUs. Each dot represents a GPU

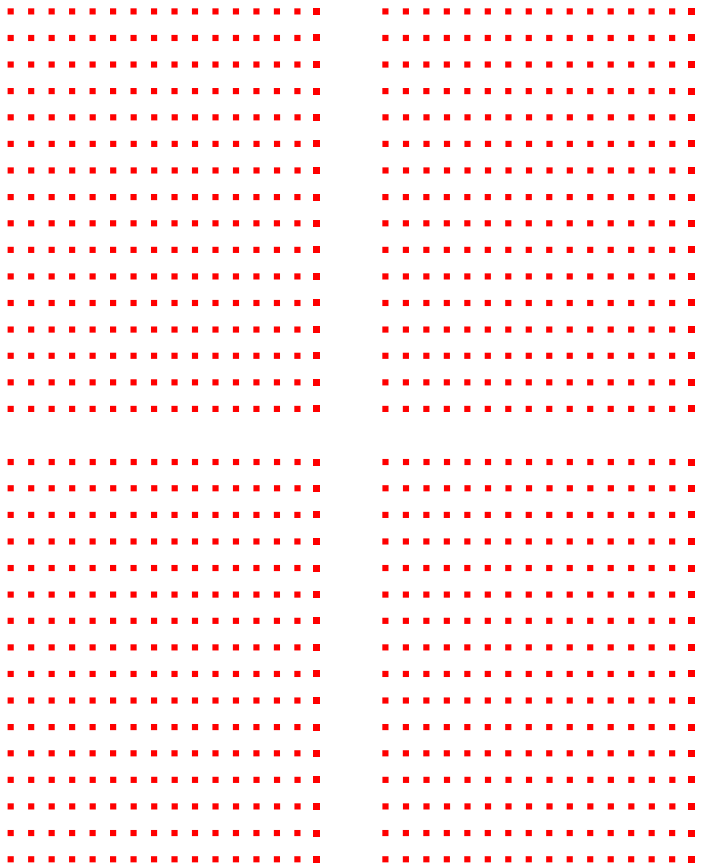


All GPUs do all the work in the layers they are working on

Computationally Intensive, thus Power Intensive

Dense models vs. Sparsity Based on MoEs

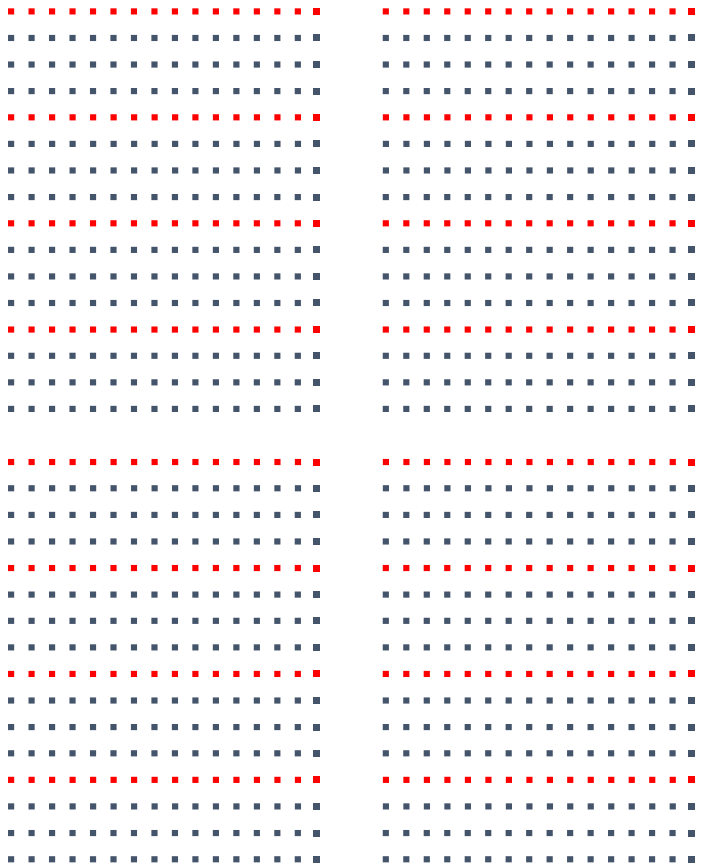
Example Dense model with a 1024 GPU model Replica



All GPUs do all the work in the layers they are working on
Computationally Intensive, thus Power Intensive

Example Mixture of experts: 1024 GPU model Replica divided into 64 experts. 16 GPUs per expert,

- Only 16 active experts at a time.



A fraction of the GPUs do the work.
Lower computational requirements, lower power

Dense models vs. Mixture of experts (MoE) A Form of Sparsity

Dense models

- All GPUs implementing model parallelism have the full / same capability
- Thus 100% computational throughput is required with Dense models

Dense model Example: All Food

American, Chinese, British, French, German, Mexican, Indian, Italian, Japanese, Spanish, etc.

Hamburger Computational Cost

100% Computation Cost

MoE models

- **Form of Sparsity**
- Subsets of GPUs are experts
- Only a fraction of experts are active during inference therefore a fraction of the power and compute resources are required

Dense models vs. Mixture of experts (MoE) A Form of Sparsity

Dense models

- All GPUs implementing model parallelism have the full/same capability
- Thus 100% computational throughput is required with Dense models

MoE models

- **Form of Sparsity**
- Subsets of GPUs are experts
- Only a fraction of experts are active during inference therefore a fraction of the power and compute resources are required

Dense model Example: All Food

American, Chinese, British, French, German, Mexican, Indian, Italian, Japanese, Spanish, etc.

Hamburger Computational Cost

100% Computation Cost

Mixture of experts model Example: Food by Continents

American, Mexican

British, French, German, Italian Spanish,

Chinese, Indian, Japanese

Dense models vs. Mixture of experts (MoE) A Form of Sparsity

Dense models

- All GPUs implementing model parallelism have the full/same capability
- Thus 100% computational throughput is required with Dense models

MoE models

- **Form of Sparsity**
- Subsets of GPUs are experts
- Only a fraction of experts are active during inference therefore a fraction of the power and compute resources are required

Dense model Example: All Food

American, Chinese, British, French, German, Mexican, Indian, Italian, Japanese, Spanish, etc.

Mixture of experts model Example: Food by Continents

American, Mexican

British, French, German, Italian Spanish,

Chinese, Indian, Japanese

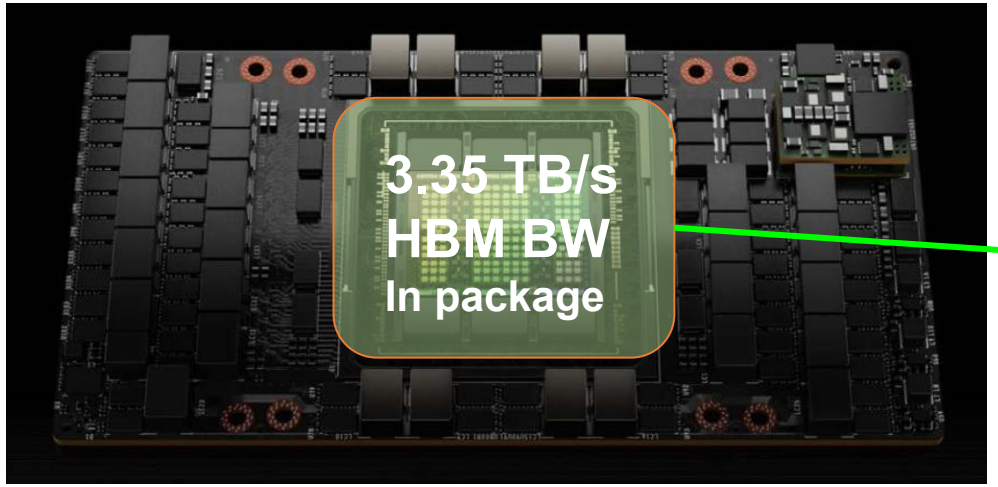
Hamburger Computational Cost

100% Computation Cost

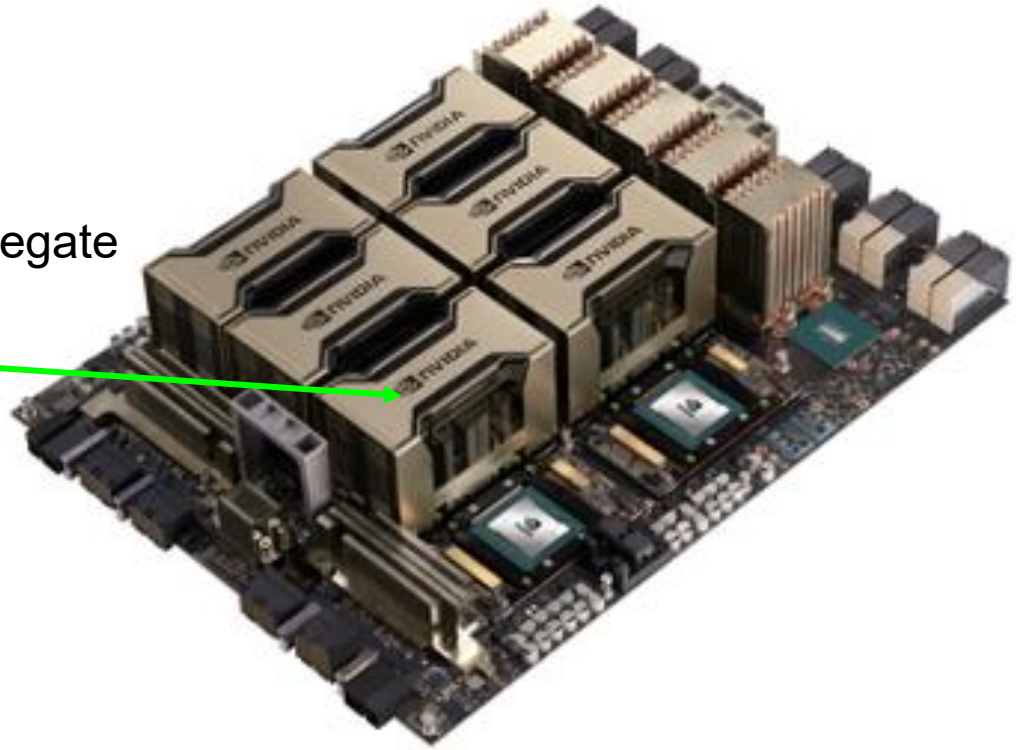
Fractional computation cost

DeepSeekMoE and DeepSeek-V3

- DeepSeek Observations
  - Matrix math within a GPU package benefits from **> an order of magnitude**
    - More interconnect **bandwidth** than scale-up interconnects such as NVLink or Infinity Fabric (XGMI)
    - Lower interconnect **latency** than scale-up interconnects
    - Lower **power** for the data transferred than scale-up interconnects



NVLink 900 GB/s aggregate BW to 7 other GPUs



NVIDIA H100 GPU Datasheet

DeepSeekMoE: Sparsity to the Maximum

Example Mixture of experts: 1024 GPU model Replica divided into 64 experts. 16 GPUs per expert,

- Only 16 active experts = 256 active GPUs

MoE before DeepSeek

DeepSeekMoE

Large less-efficient experts

- Experts have capability overlap
- Experts have tensor slicing communication between packages

DeepSeekMoE: Sparsity to the Maximum

Example Mixture of experts: 1024 GPU model Replica divided into 64 experts. 16 GPUs per expert,

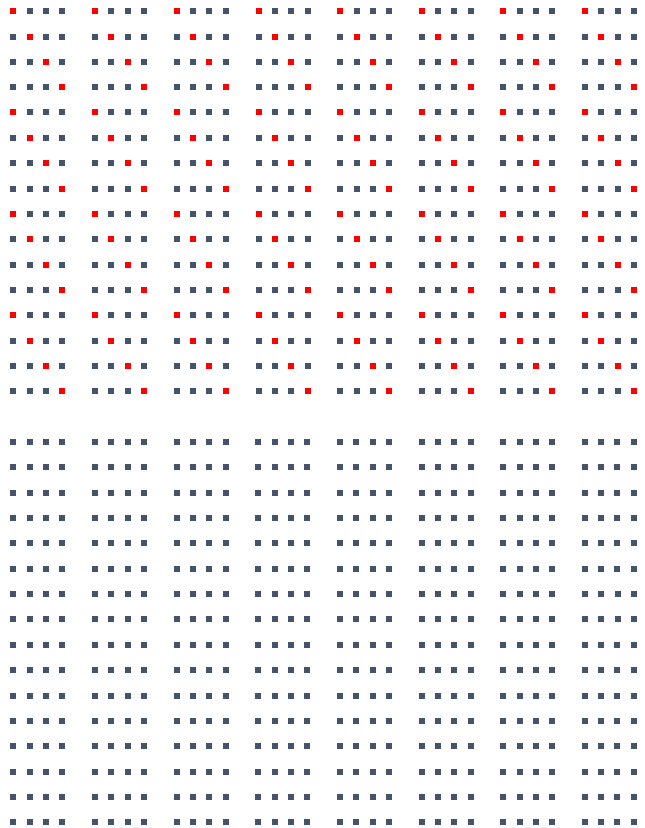
- Only 16 active experts = 256 active GPUs

MoE before DeepSeek

DeepSeekMoE

Many smaller, more specialized experts fit inside a single GPU package.

- 128 active experts = 128 active GPUs



Large less-efficient experts

- Experts have capability overlap
- Experts have tensor slicing communication between packages

Maximum efficiency experts

- Minimum expert overlap
- Experts fit in a single GPU package

DeepSeekMoE and DeepSeek-V3

- DeepSeek Observations
  - Fewer large experts that require tensor slicing across multiple GPUs are less efficient than many smaller (more specialize experts) that keep expert matrix math inside a GPU package

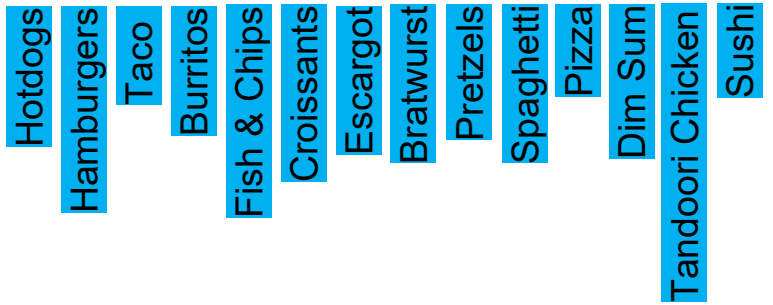
MoE with large experts Example:
Food by Continents

American, Mexican

British, French, German, Italian Spanish,

Chinese, Indian, Japanese

MoE with fine grain experts that fit in a GPU package
Example:



Etc.

DeepSeekMoE and DeepSeek-V3

- DeepSeek Observations:
  - Fewer large experts that require tensor slicing across multiple GPUs are less efficient than many smaller (more specialize experts) that keep expert matrix math inside a GPU package

MoE with large experts Example
Food by Continents

American, Mexican

British, French, German, Italian Spanish,

Chinese, Indian, Japanese

MoE with fine grain experts that fit in a GPU package
Example

Hotdogs, Hamburgers, Taco, Burritos, Fish & Chips, Croissants, Escargot, Bratwurst, Pretzels, Spaghetti, Pizza, Dim Sum, Tandoori Chicken, Sushi

Etc.

Margarita Pizza Computation & Communication Cost:

Relatively larger

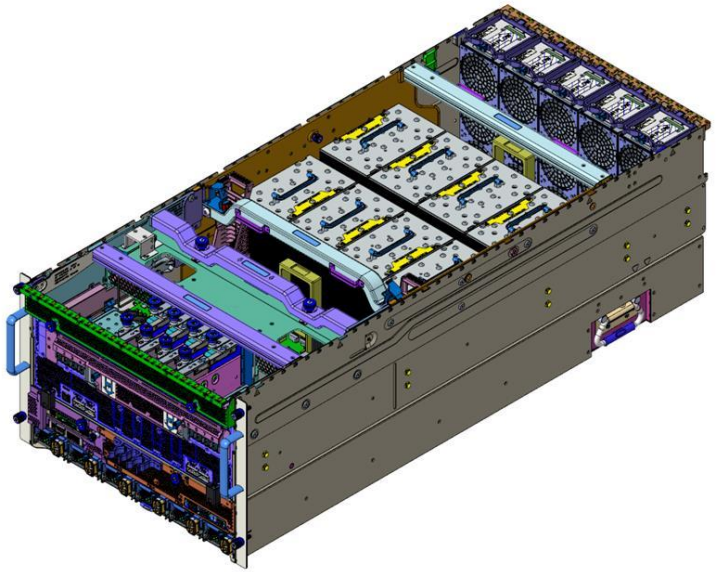
Vs.

Pizza

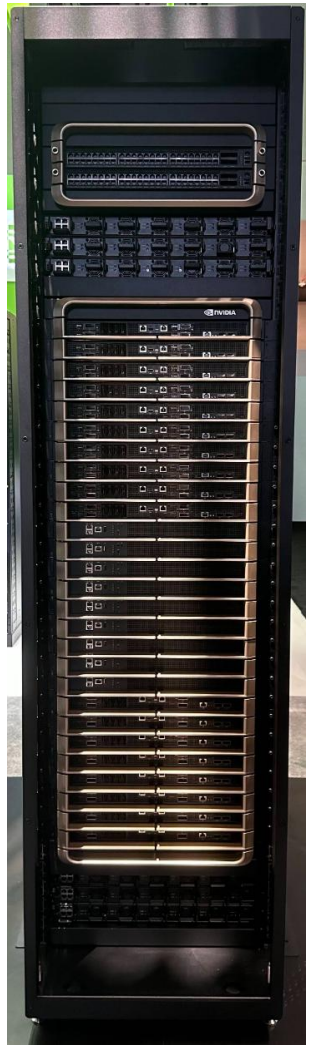
DeepSeekMoE and DeepSeek-V3

Margarita Pizza expert Computation & Communication Cost

Relatively larger experts spanning multiple GPUs



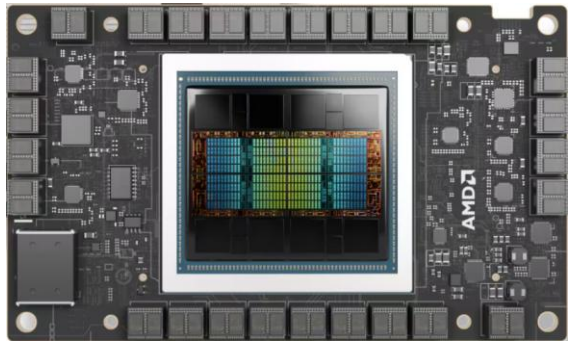
or



Vs.

Pizza

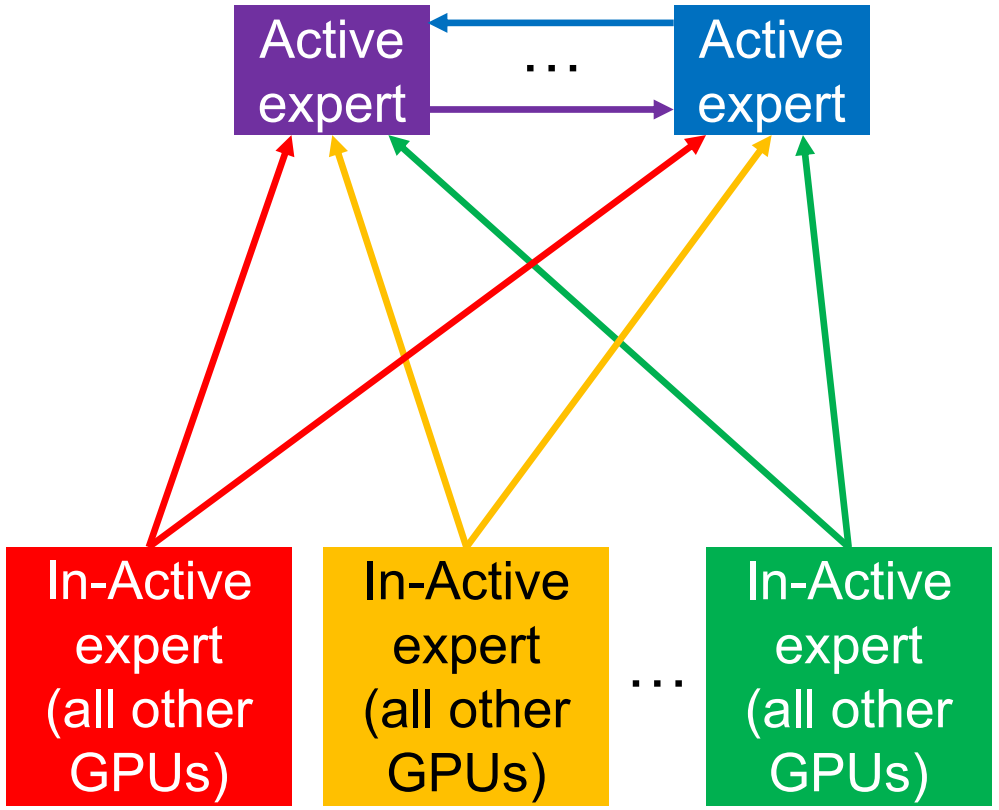
Fine grain experts fitting in a single GPU package



MoE Communication Pattern

Before expert's matrix math

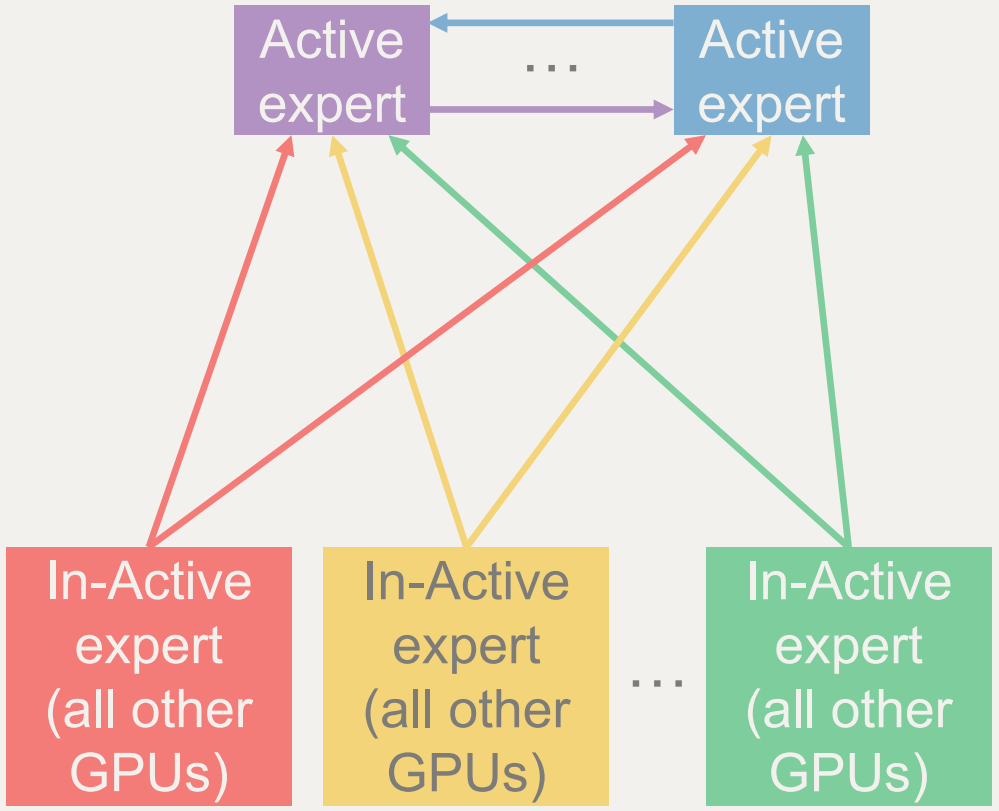
- All GPU's activations to Active experts



Selective expert Activation

MoE Communication Pattern

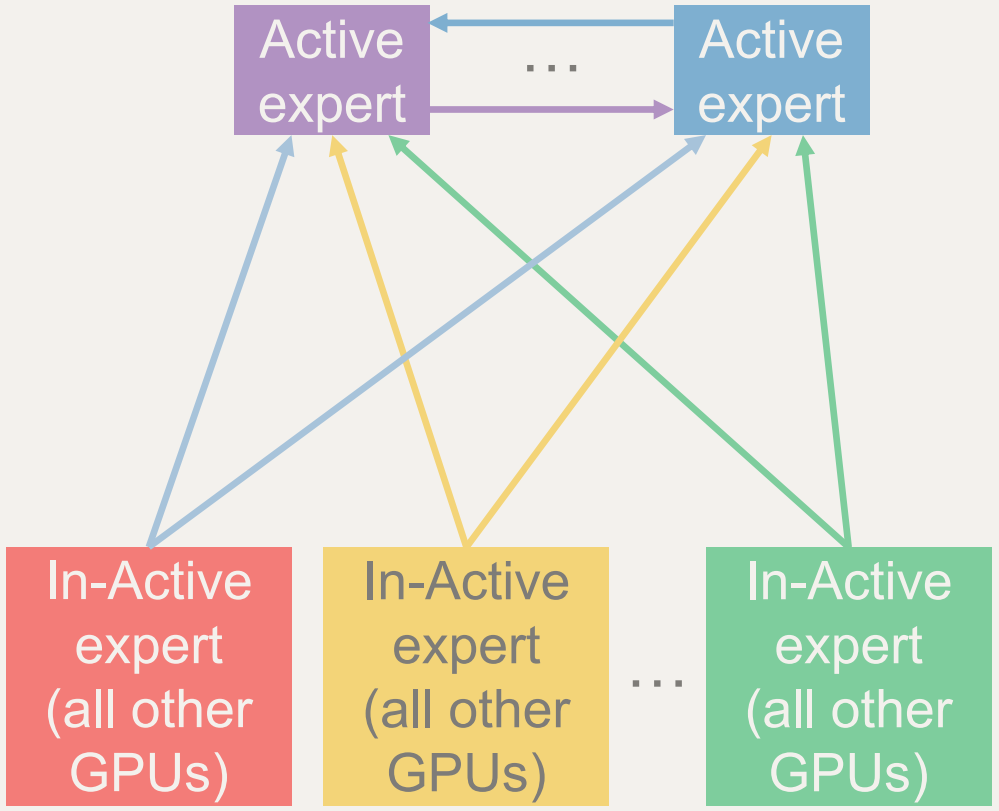
Experts do work



Experts do their work

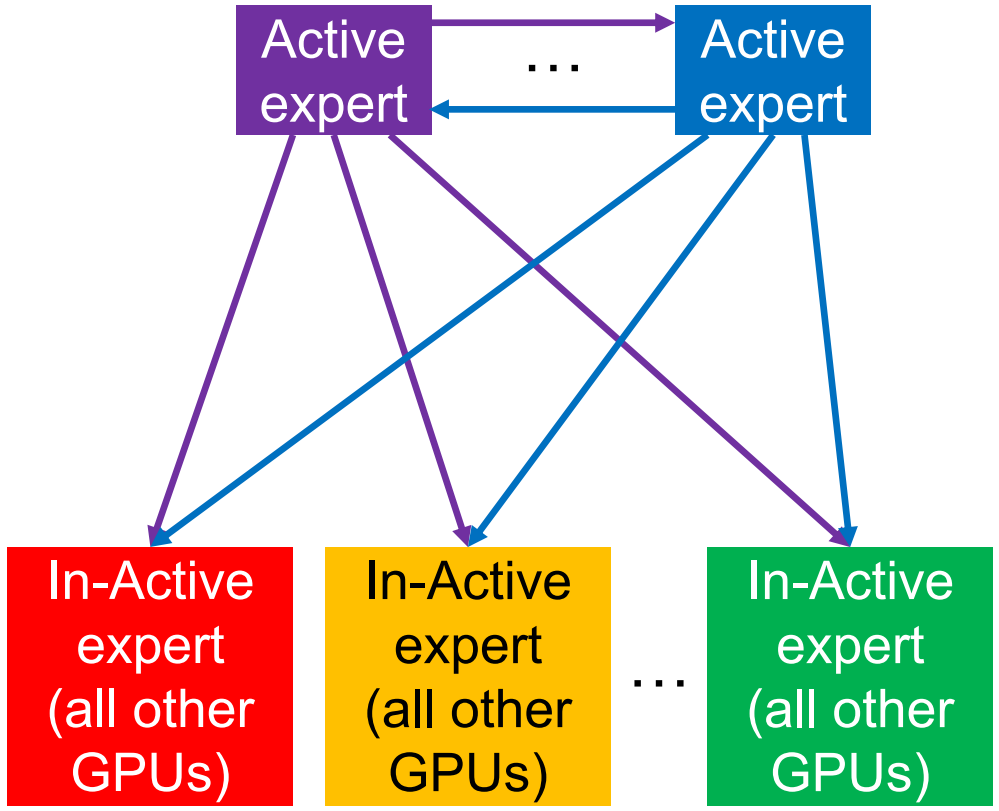
MoE Communication Pattern

After expert's work

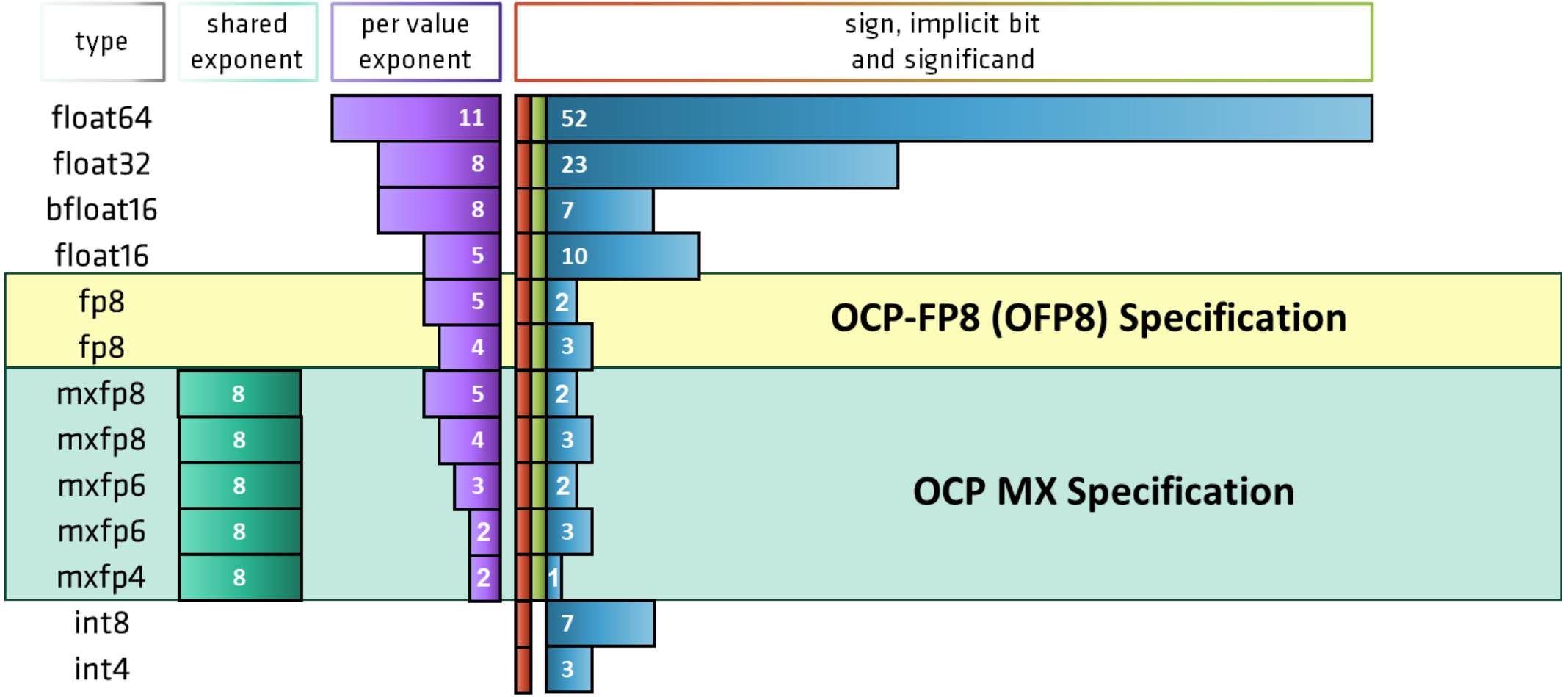


Experts do their work

- Active experts broadcast their results to the other GPUs:

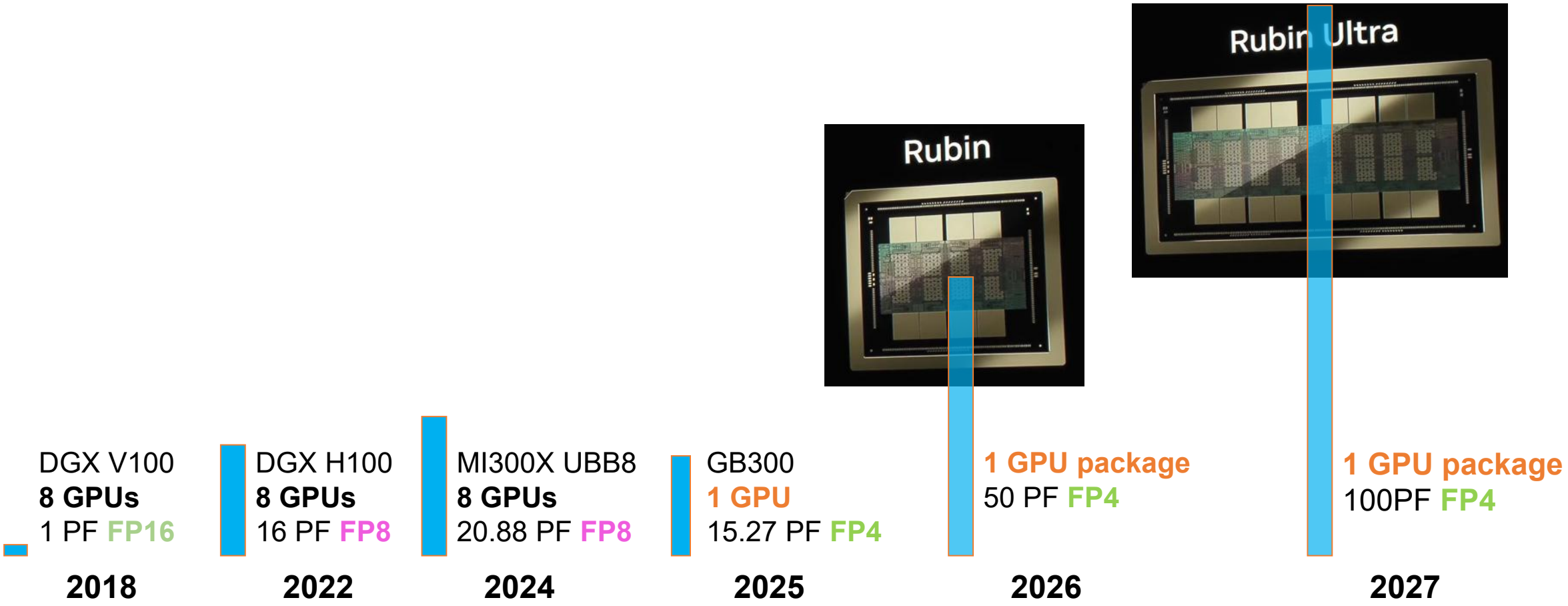


Industry Advancements in Reduced Precision



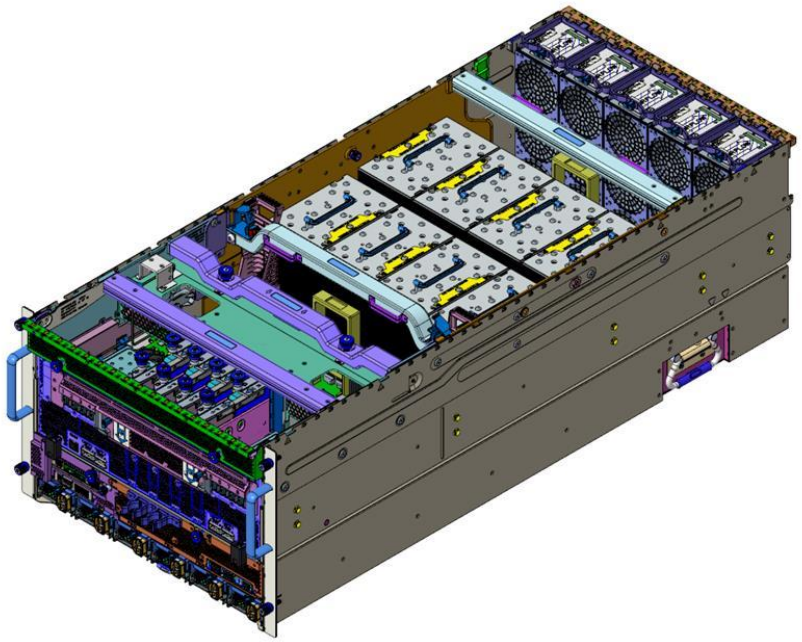
- AMD, Arm®, Intel®, Meta, Microsoft®, NVIDIA, and Qualcomm Standardize Next-Generation Narrow Precision Data Formats for AI » Open Compute Project
- OCP Microscaling Formats (MX) Specification Version 1.0

Growth: Computational Throughput in a Single GPU Package (Dense FLOPS)



Growth: Scale-Up Domain

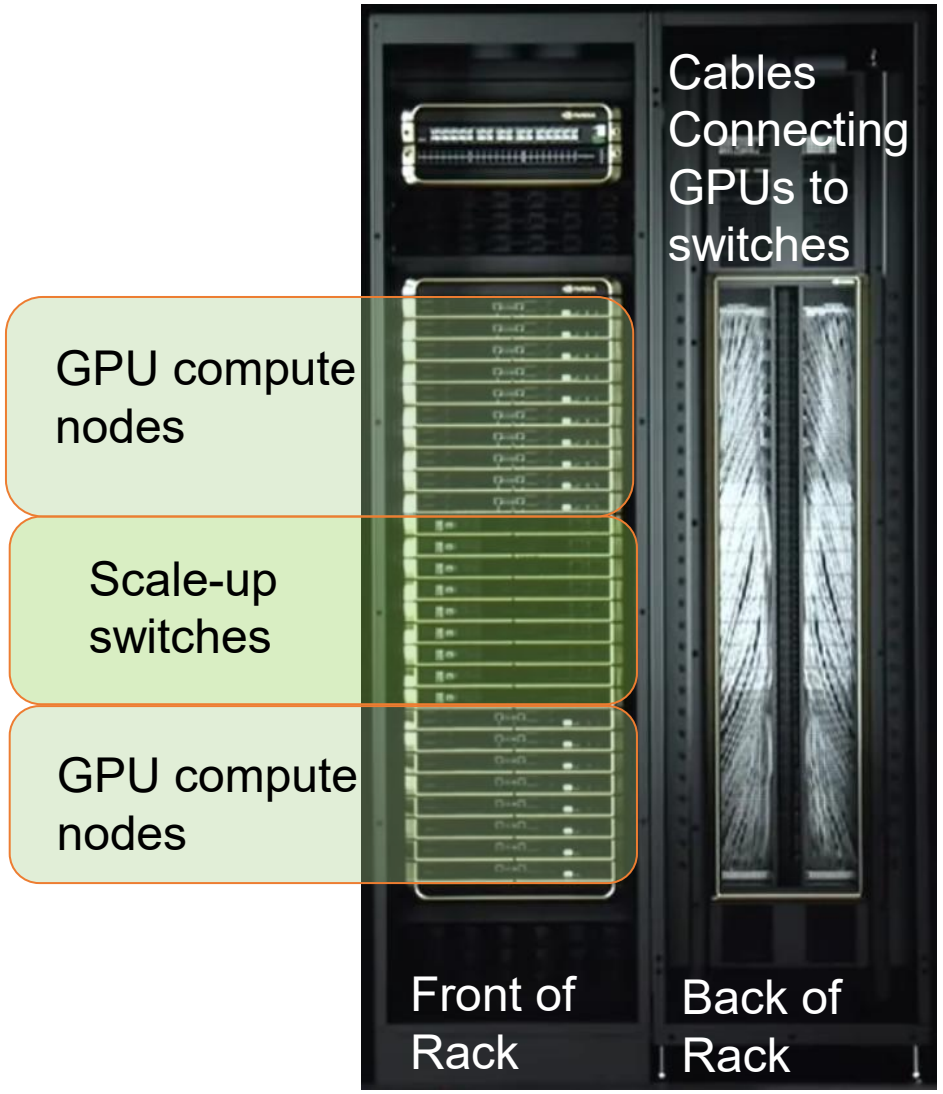
Moving from a system of 8 GPU packages



To many systems

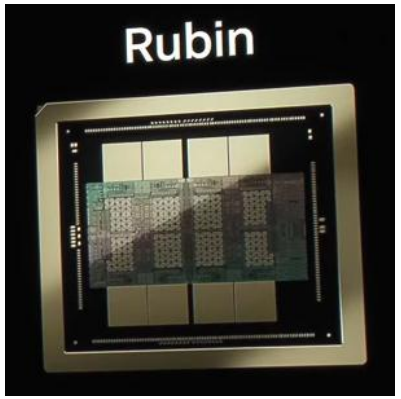
With a shared scale-up domain in a rack

GB300 NVL72



Three Levels of GPU Interconnect

Relative Interconnect BW per Reticle Size GPU



~2 reticles

In a GPU package

1.8 TB/s uni-dir

e.g., 72

Scale-up between GPU packages

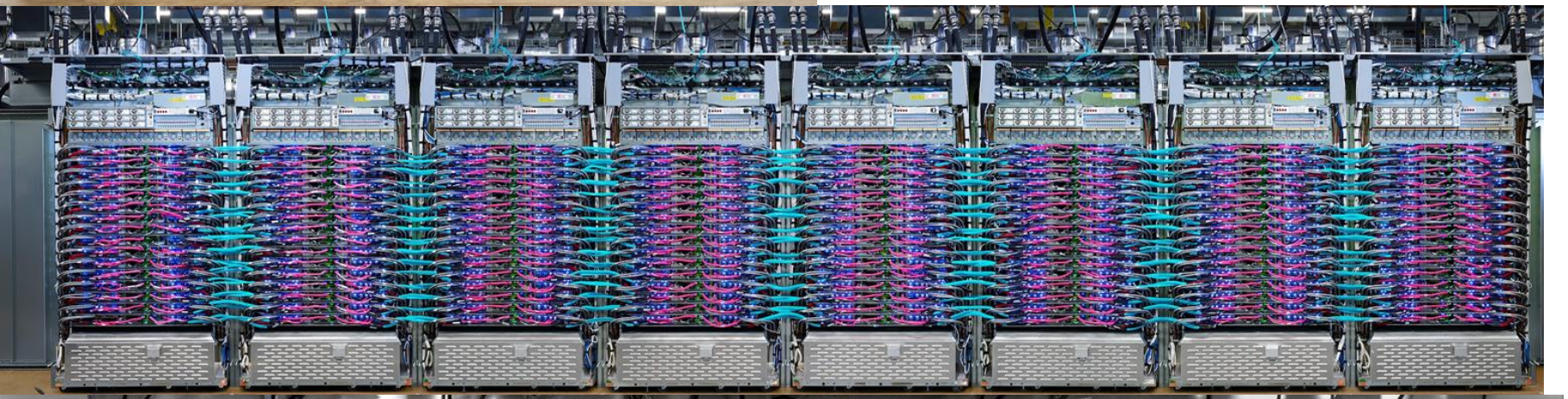
Hundreds of thousands

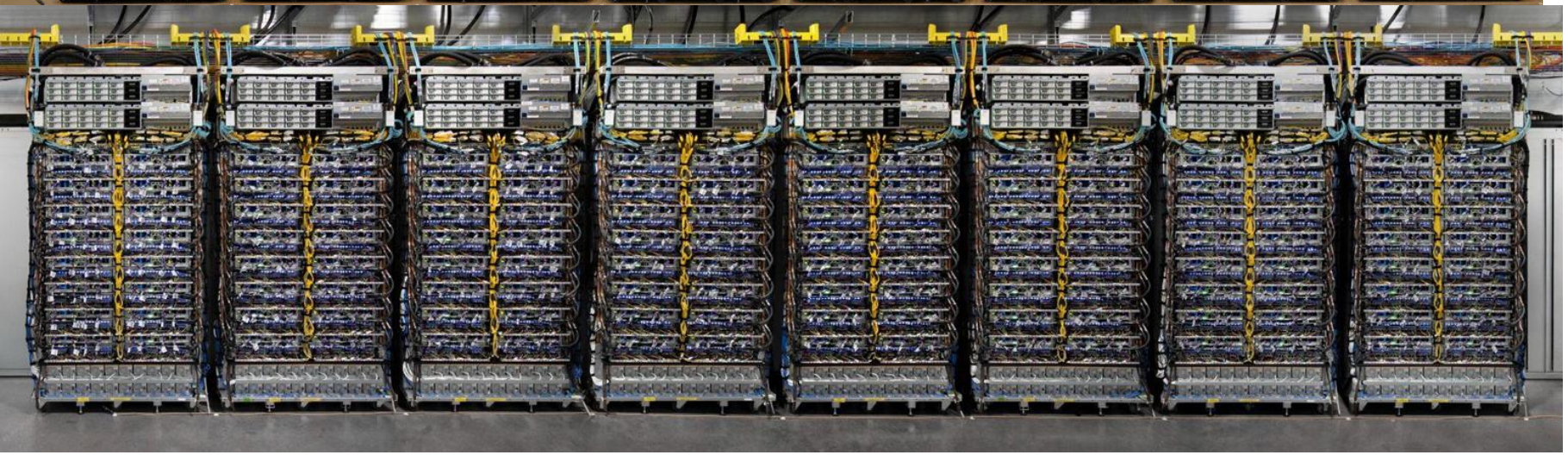
Clonee Data Center | Meta

0.2 TB/s uni-dir
(to for example 16K GPUs and significant BW taper above that)

RDMA Scale-out per GPU
(ethernet or InfiniBand™)

#s of GPUs





Scale *= 4;

TPU v2: 256 nodes
electrical, 2D torus

TPU v3: 1024 nodes
electrical, 2D torus

TPU v4: 4096 nodes
electrical in- rack
optical across racks
3D torus
8/64 racks shown
(7 more rows in depth)

GPU Cluster Implementation Challenges

- Scale-up and Scale-out BW / GPU has been doubling every 2 years
- Copper interconnect reach is decreasing with increased SERDES bit rates
- There is a limit to SERDES electrical bit rates
- Doubling the number of SERDES lanes increases connector size and cable bulk
- Limited copper reach is driving GPU and scale-up switch proximity (rack density)
- More capable multi-reticle sized GPUs are driving power per GPU up and forcing the use of liquid cooling
- Rack power density is growing as a function of power per GPU * # of GPUs per rack
- Rack compute density driving power supply disaggregation

References

- Clonee Data Center | Meta
- ‘Big Sur’ Hardware | Meta
- NVIDIA DGX-1 with Tesla V100 System Architecture White paper
- NVIDIA Contributes H100 Tensor Core-based HGX Baseboard Physical Spec to OCP OAI Project! » Open Compute Project
- Introduction to NVIDIA DGX H100/H200 Systems — NVIDIA DGX H100/H200 User Guide
- AMD, Arm, Intel, Meta, Microsoft, NVIDIA, and Qualcomm Standardize Next-Generation Narrow Precision Data Formats for AI » Open Compute Project
- OCP Microscaling Formats (MX) Specification Version 1.0
- https://arxiv.org/pdf/2203.12533
- ImageNet Classification with Deep Convolutional Neural Networks
- 1706.03762
- [1810.04805] BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
- Model Parallelism — transformers 4.7.0 documentation
- NVIDIA V100S Datasheet
- Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism
- Designed for AI Reasoning Performance & Efficiency | NVIDIA GB300 NVL72
- NVIDIA DGX GB300 Datasheet
- AMD Instinct™ MI300X Accelerators