

peaceful sharing while training

Pinar Tözün

Associate Professor, IT University of Copenhagen

pito@itu.dk, pinartozun.com, [@pinartozun](https://twitter.com/pinartozun)

unsustainable growth of deep learning

2023

Gemini Ultra

GPT-4

PaLM (540B)

GPT-3 175B (davinci)

Megatron-Turing NLG 530B

Llama 2 70B

UNIVERSITY of WASHINGTON

RoBERTa Large

LaMDA

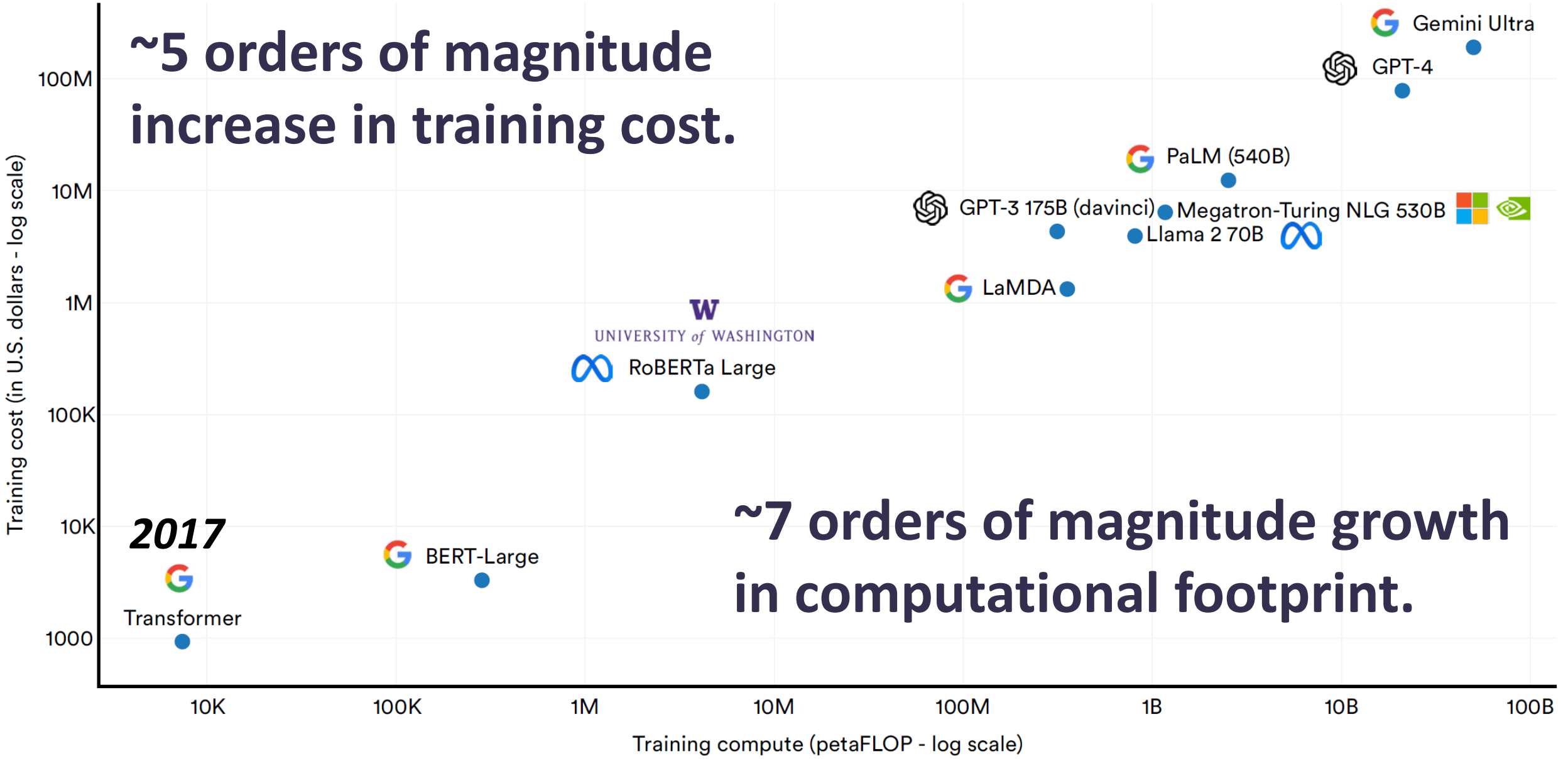
~5 orders of magnitude increase in training cost.

~7 orders of magnitude growth in computational footprint.

2017

Transformer

BERT-Large



hardware underutilization

NVIDIA H200

141GB memory

50MB L2 cache

4.8TB/s
memory bandwidth

in the meanwhile, on pre-H200 GPUs ...

- *@ITU*, many ML jobs utilize ***less than 50% of GPU resources***
e.g., transfer learning, small models
- ***in real-world****, ***~52% GPU utilization***
on average for 100,000 jobs

can we do better while using fewer resources?

sharing for deep learning training

- GPU sharing

[An Analysis of Collocation on GPUs for Deep Learning Training](#)

Ties Robroek, Ehsan Yousefzadeh-Asl-Miandoab, Pinar Tözün.
EuroMLSys 2024

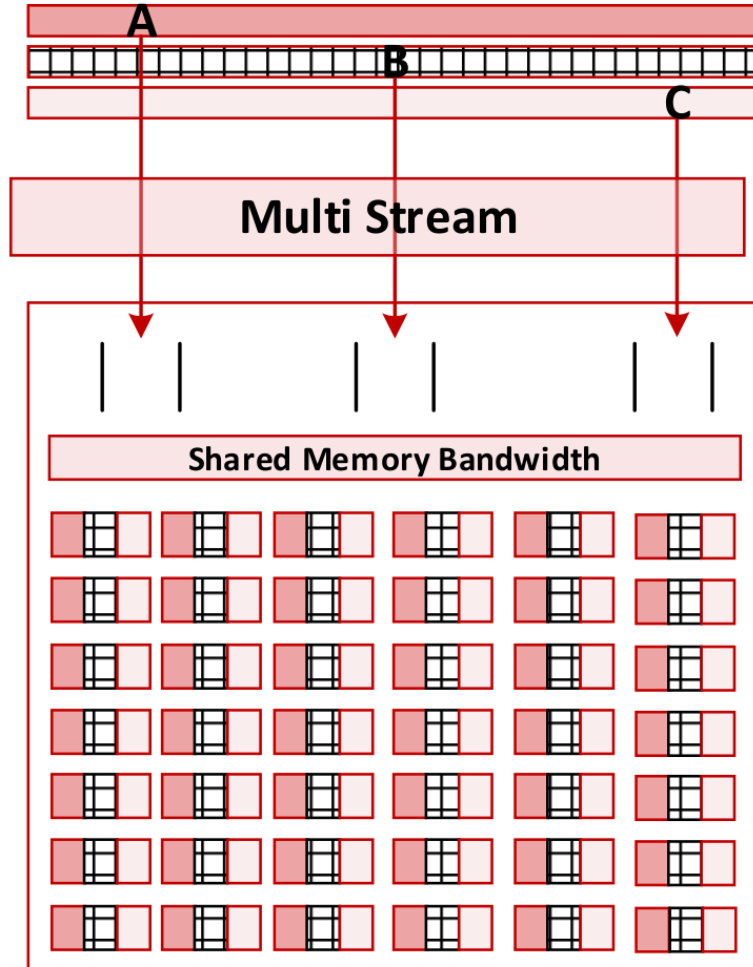
- data & work sharing

[TensorSocket: Shared Data Loading for Deep Learning Training](#)

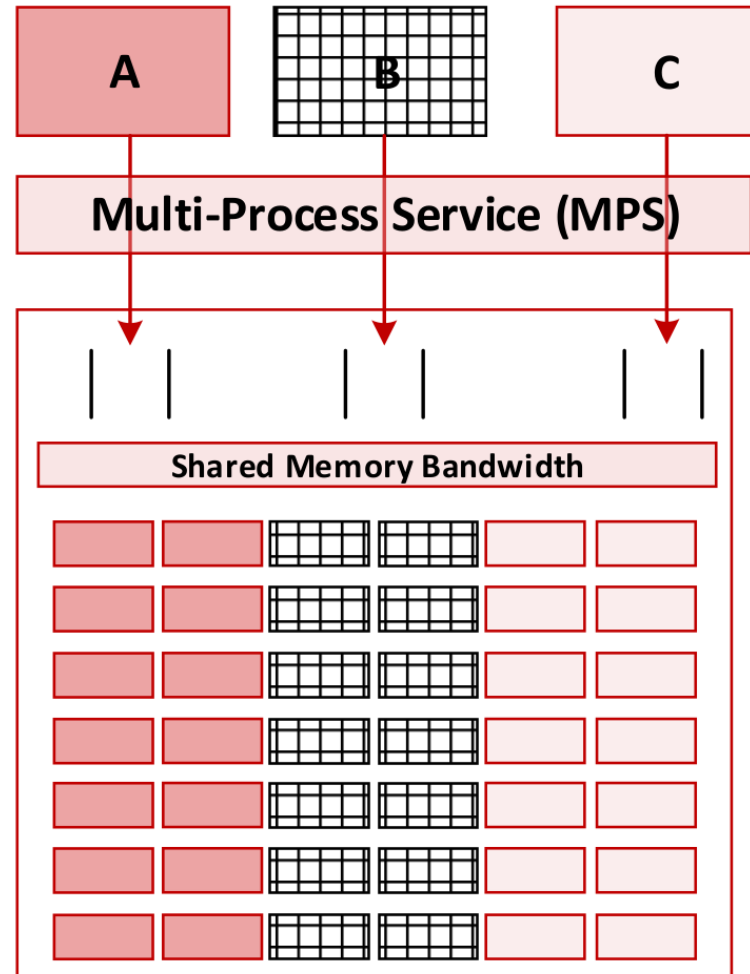
Ties Robroek, Neil Kim Nielsen, Pinar Tözün.



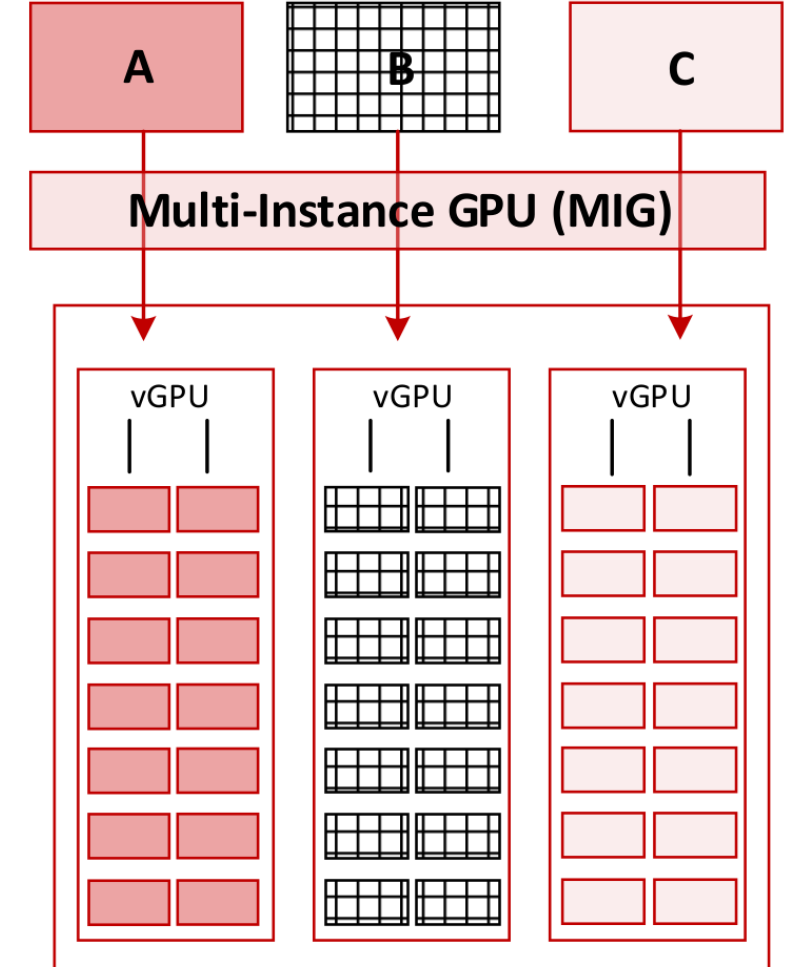
sharing resources on (NVIDIA) GPUs



- most straightforward
- time-multiplexing
- ✗ limited parallelism



- finer-grained sharing
- ✗ single user
(due to safety)



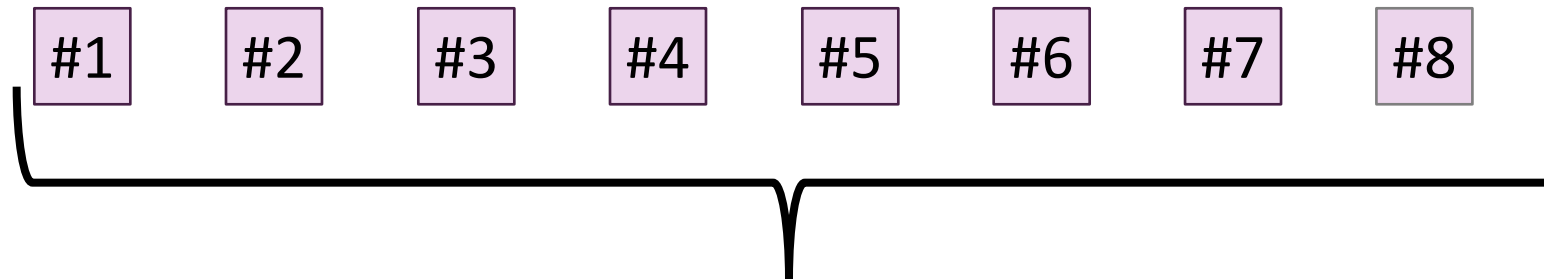
- hardware-support for resource split
- ✗ rigid partitioning

multi-instance GPU

compute:




memory:



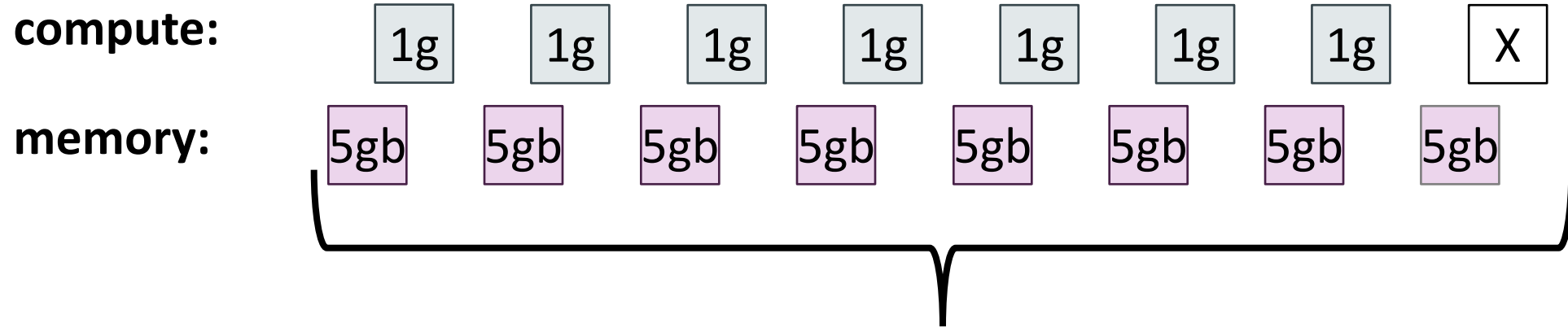
 1 compute unit

 1 memory unit

 unused available (memory/compute) unit

 unavailable compute unit

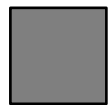
multi-instance GPU on A100 (40GB)



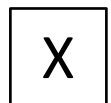
1 compute unit = 1g = 14 SMs



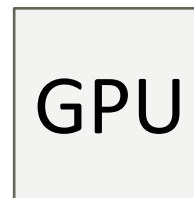
1 memory unit = 5GB



unused available (memory/compute) unit



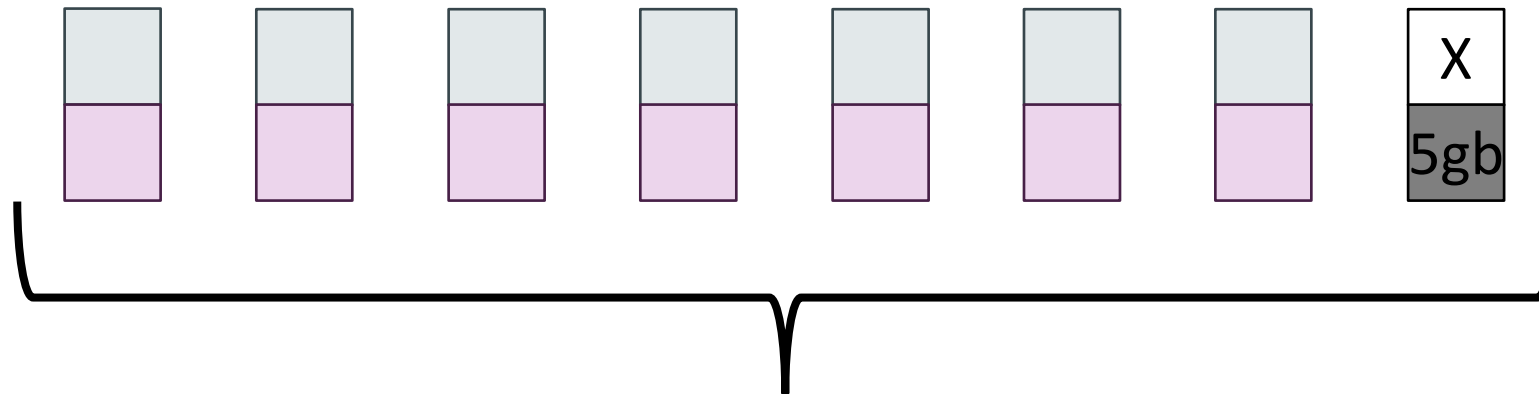
unavailable compute unit = 10 SMs (streaming multiprocessor)



multi-instance GPU on A100 (40GB)

compute:

memory:



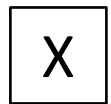
1 compute unit = 1g = 14 SMs



1 memory unit = 5GB



unused available (memory/compute) unit



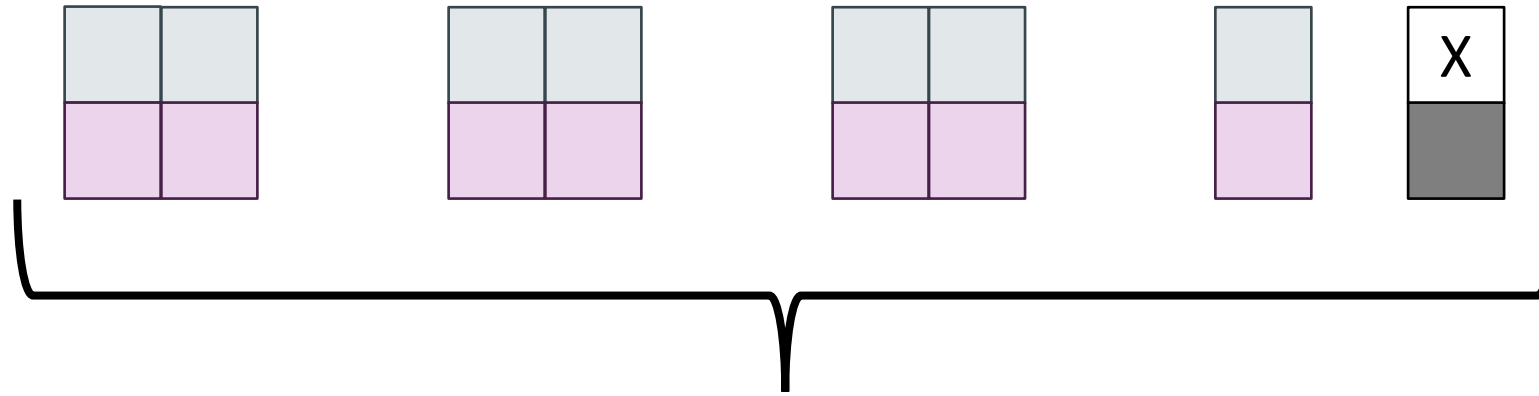
unavailable compute unit = 10 SMs (streaming multiprocessor)

GPU

multi-instance GPU on A100 (40GB)

compute:

memory:



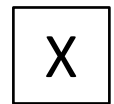
1 compute unit = 1g = 14 SMs



1 memory unit = 5GB



unused available (memory/compute) unit



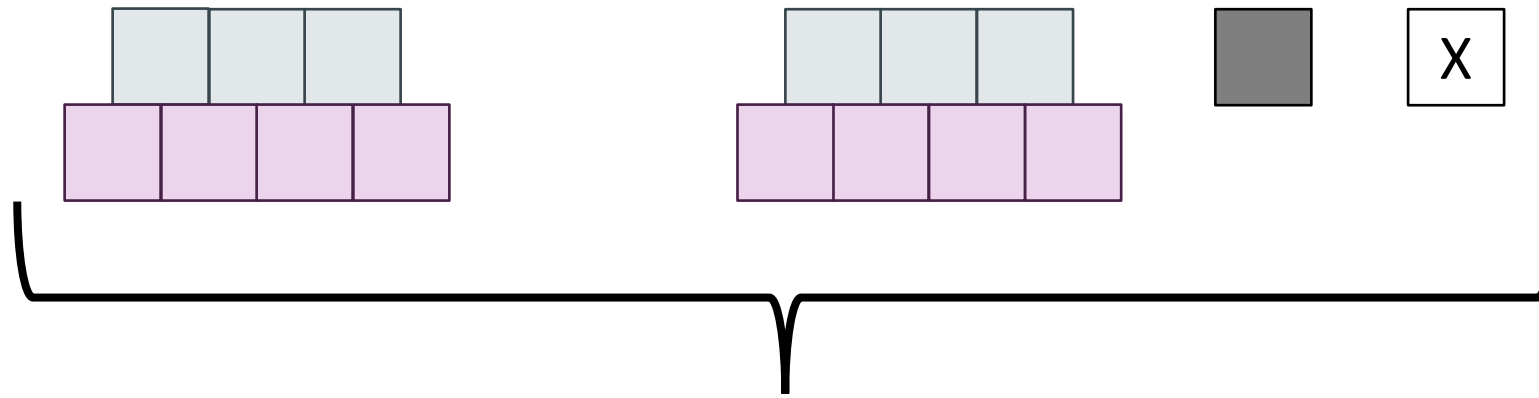
unavailable compute unit = 10 SMs (streaming multiprocessor)

GPU

multi-instance GPU on A100 (40GB)

compute:

memory:



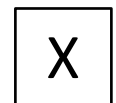
1 compute unit = 1g = 14 SMs



1 memory unit = 5GB



unused available (memory/compute) unit



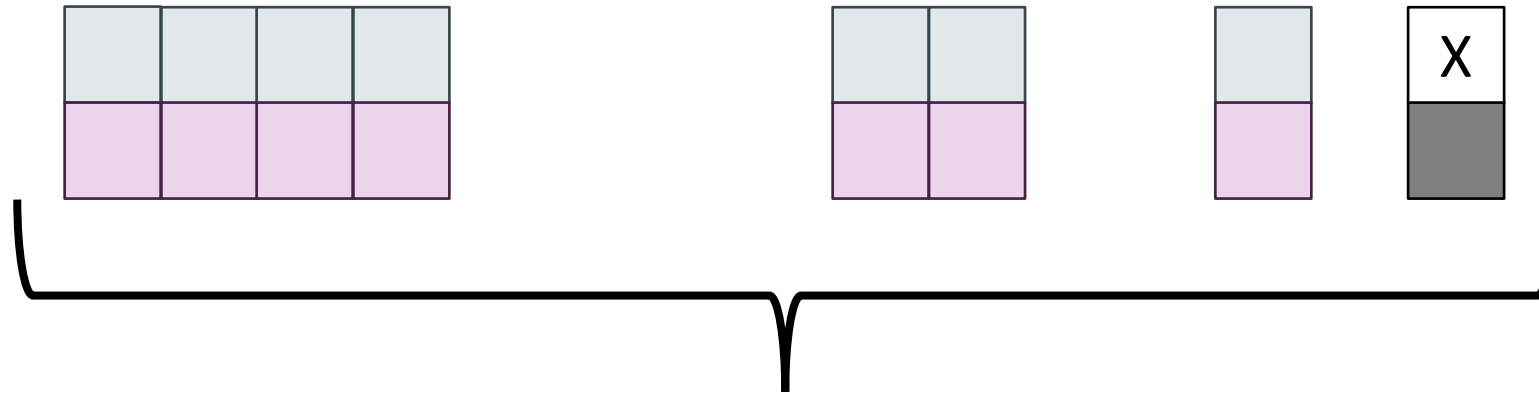
unavailable compute unit = 10 SMs (streaming multiprocessor)

GPU

multi-instance GPU on A100 (40GB)

compute:

memory:



1 compute unit = 1g = 14 SMs

GPU



1 memory unit = 5GB



unused available (memory/compute) unit

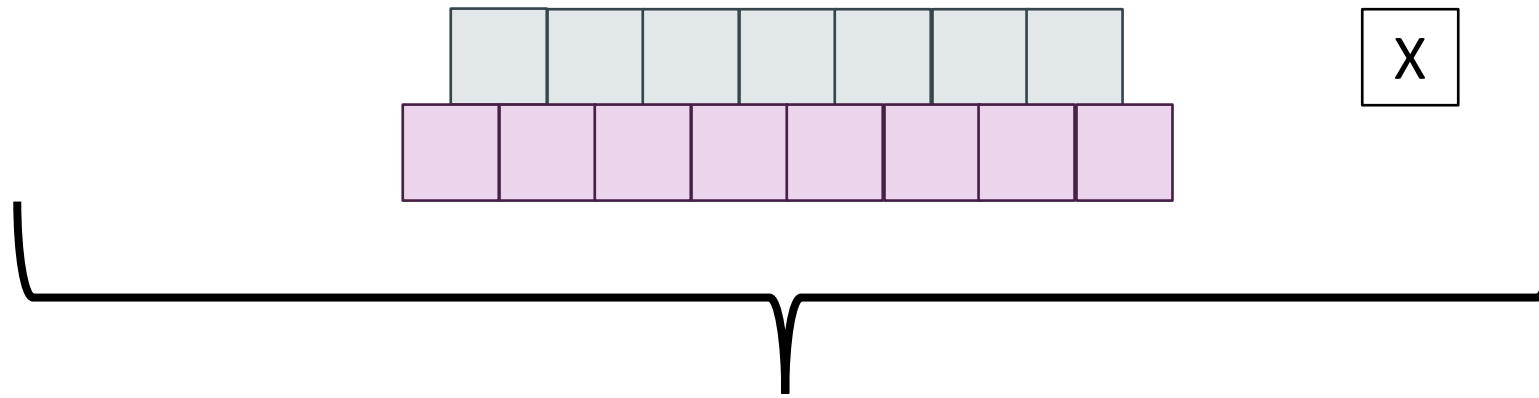


unavailable compute unit = 10 SMs (streaming multiprocessor)

multi-instance GPU on A100 (40GB)

compute:

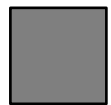
memory:



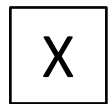
1 compute unit = 1g = 14 SMs



1 memory unit = 5GB

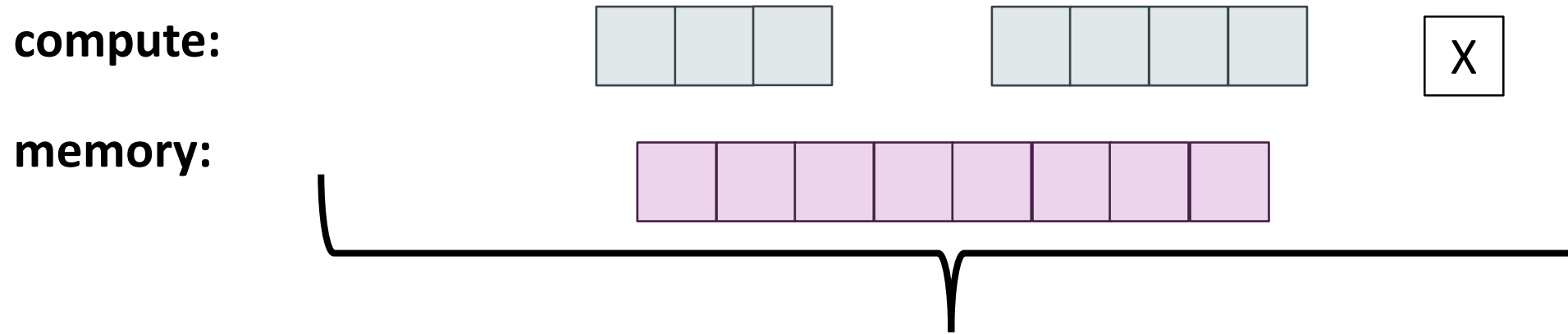


unused available (memory/compute) unit



unavailable compute unit = 10 SMs (streaming multiprocessor)

multi-instance GPU on A100 (40GB)



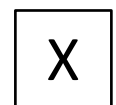
1 compute unit = 1g = 14 SMs



1 memory unit = 5GB



unused available (memory/compute) unit

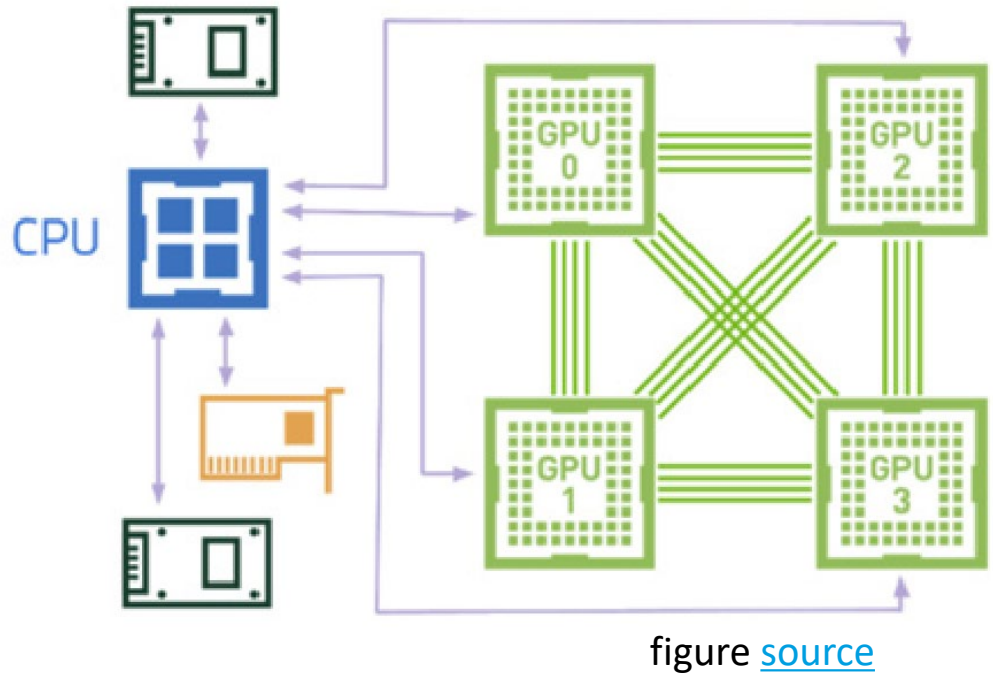


unavailable compute unit = 10 SMs (streaming multiprocessor)

GPU

performance impact of collocation

NVIDIA DGX Station A100



 Display GPU
  NVMe
  PCIe
  NVlink

CPU = AMD 7742 – 512 GB RAM

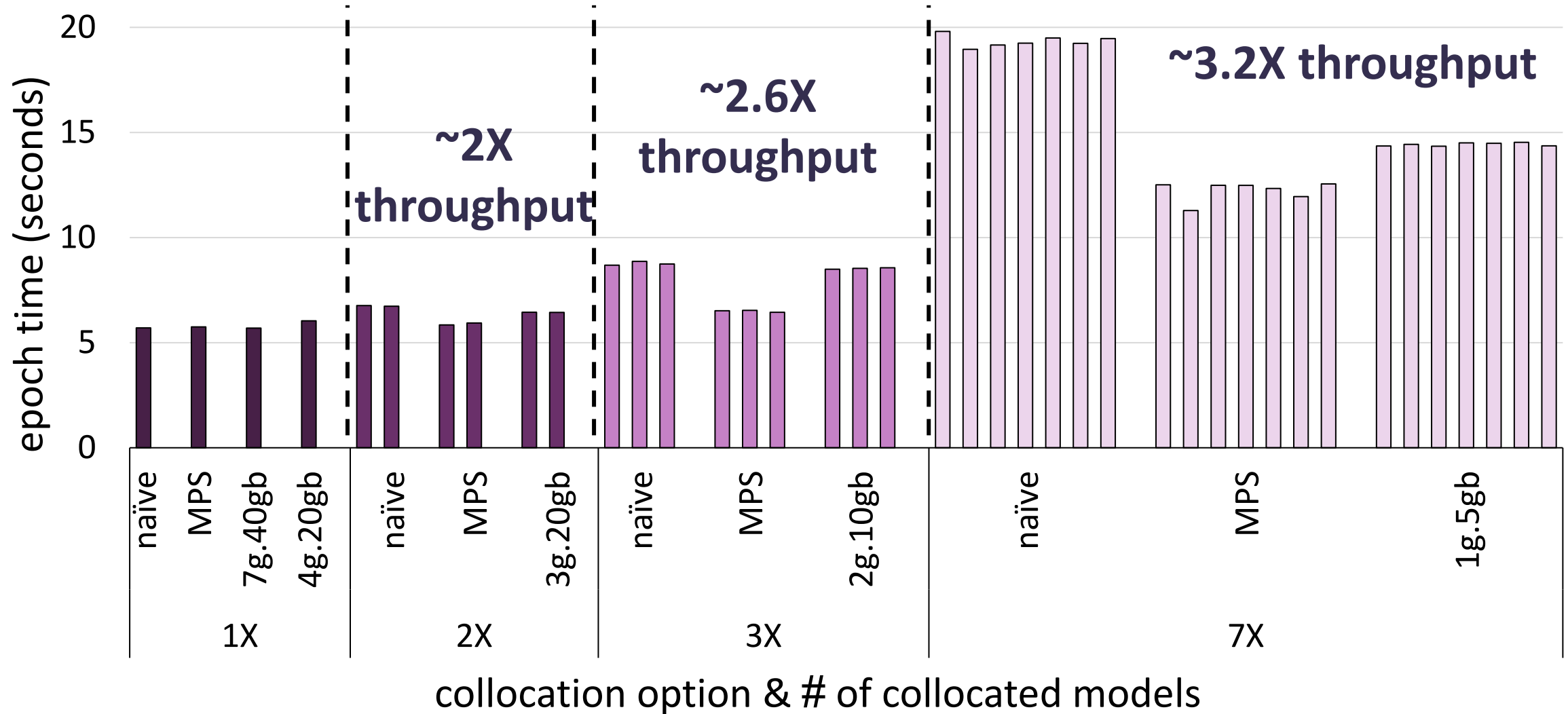
64 physical cores

GPU = NVIDIA A100 – 40 GB RAM

workloads	model	batch size	dataset
small	ResNet26 EfficientNet	128	CIFAR-10
medium	ResNet50 EfficientNet	128	downsampled ImageNet*
large	ResNet152 CaiT	32 128	ImageNet (2012)
xlarge	DLRM	1	Criteo Terabyte

- image models: CNN & transformers recommender model
- on single GPU with PyTorch v2.0
- results reported from 2nd epoch of training

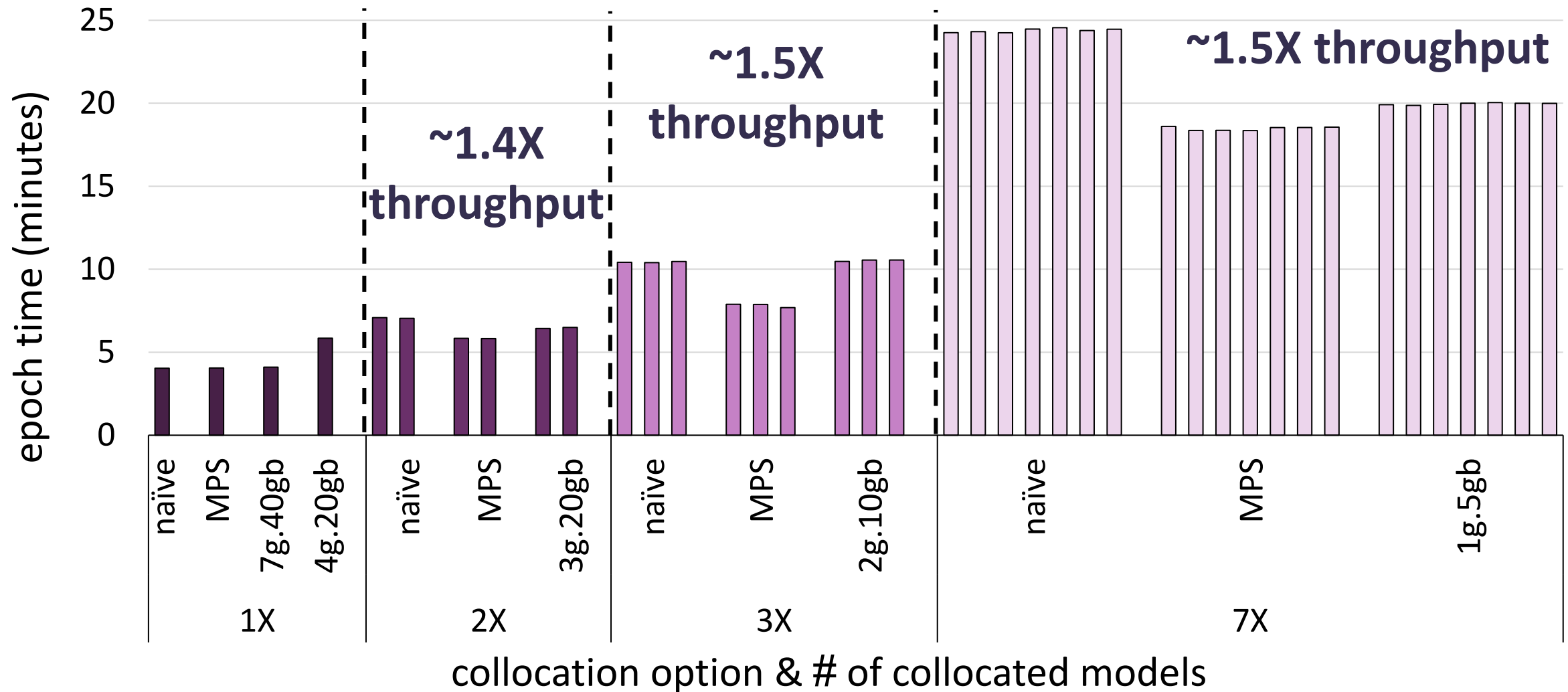
small case – ResNet26



collocation benefits despite increased epoch time

MPS > MIG > naïve

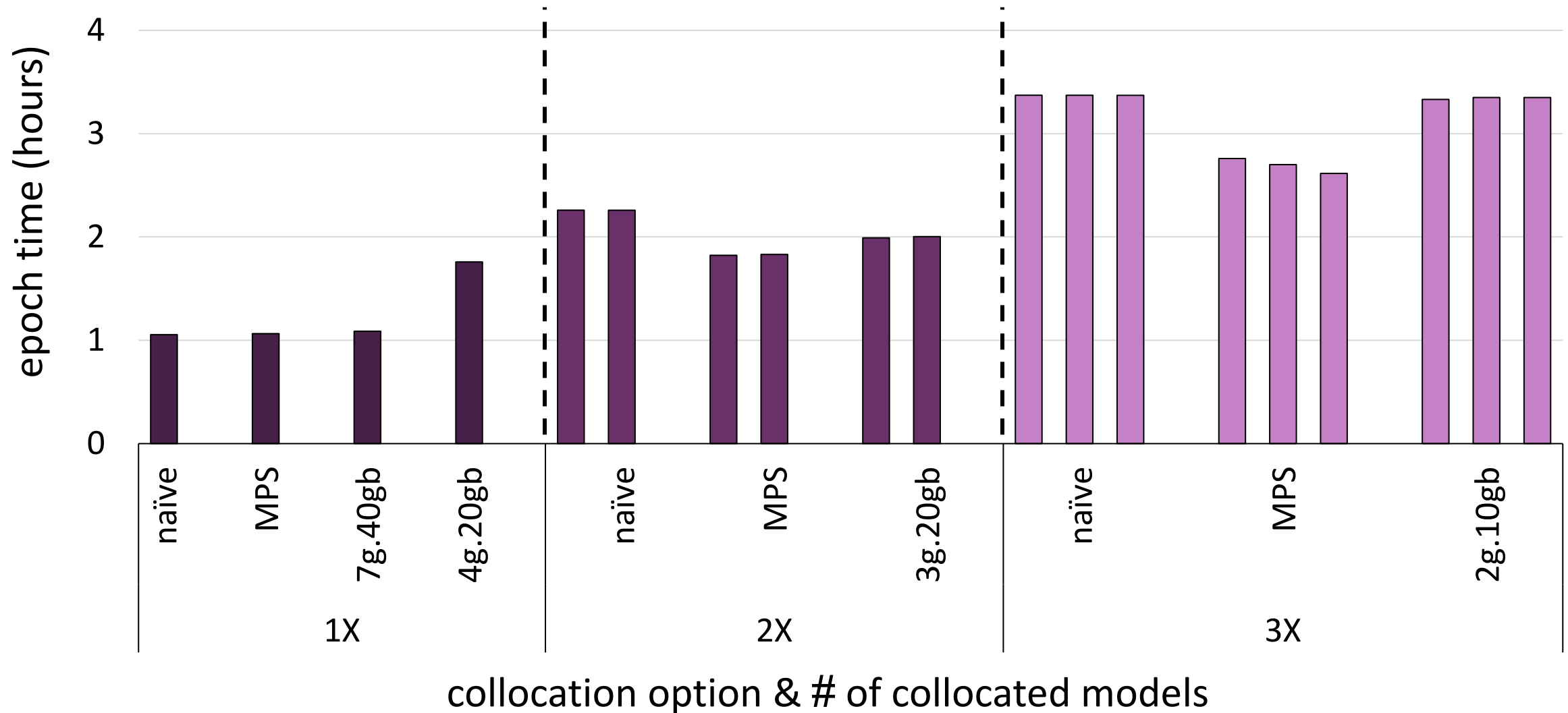
medium case – ResNet50



still some throughput benefits

but diminishing returns for increased collocation

large case – ResNet152



**no more throughput benefits – 80% utilization when training alone
better to collocate with smaller or less compute heavy tasks**

mixed workloads: compute- & memory-heavy

	DLRM – time per training block	ResNet152 – time per epoch	sm activity	memory footprint
DLRM alone			5%	29.14 GB
ResNet152 alone			82%	8.47 GB

mixed workloads: compute- & memory-heavy

	DLRM – time per training block	ResNet152 – time per epoch	sm activity	memory footprint
DLRM alone			5%	29.14 GB
ResNet152 alone			82%	8.47 GB
naïve			81%	37.75 GB
MPS			81%	37.62 GB
MIG:				
3compute – DLRM			39%	37.86 GB
4compute – ResNet				
shared memory				

mixed workloads: compute- & memory-heavy

	DLRM – time per training block	ResNet152 – time per epoch	sm activity	memory footprint
DLRM alone	5.36 h	-	5%	29.14 GB
ResNet152 alone	-	1.05 h	82%	8.47 GB
naïve			81%	37.75 GB
MPS			81%	37.62 GB
MIG:				
3compute – DLRM				
4compute – ResNet			39%	37.86 GB
shared memory				

mixed workloads: compute- & memory-heavy

	DLRM – time per training block		ResNet152 – time per epoch		sm activity	memory footprint
DLRM alone	5.36 h		-		5%	29.14 GB
ResNet152 alone	-		1.05 h		82%	8.47 GB
naïve	6.09 h	(+14%)	1.11 h	(+5%)	81%	37.75 GB
MPS	5.57 h	(+5%)	1.10 h	(+4%)	81%	37.62 GB
MIG:						
3compute – DLRM	5.60 h		1.40 h		39%	37.86 GB
4compute – ResNet						
shared memory						

**collocation can lead to (almost) free lunch
when workloads stress different hardware resources**

sharing for deep learning training

- GPU sharing

[An Analysis of Collocation on GPUs for Deep Learning Training](#)

Ties Robroek, Ehsan Yousefzadeh-Asl-Miandoab, Pinar Tözün.
EuroMLSys 2024

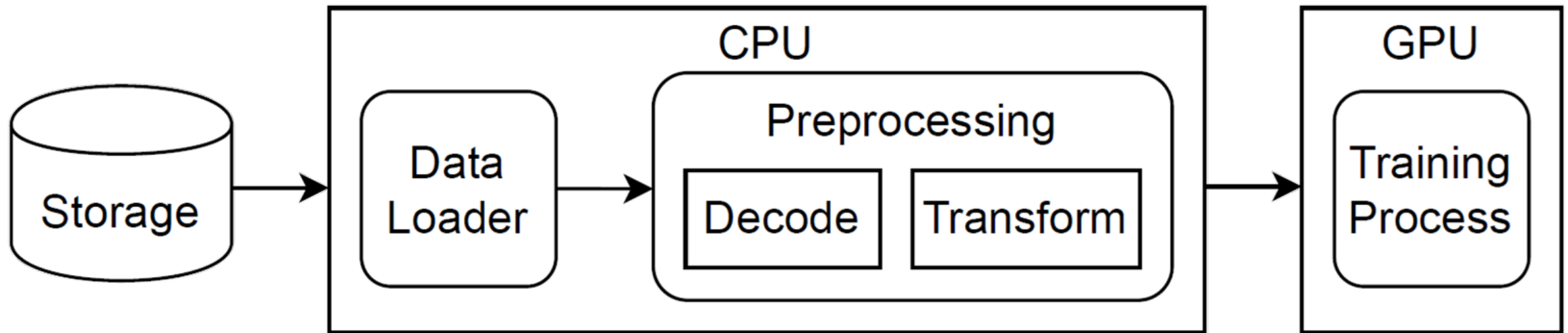
- data & work sharing

[TensorSocket: Shared Data Loading for Deep Learning Training](#)

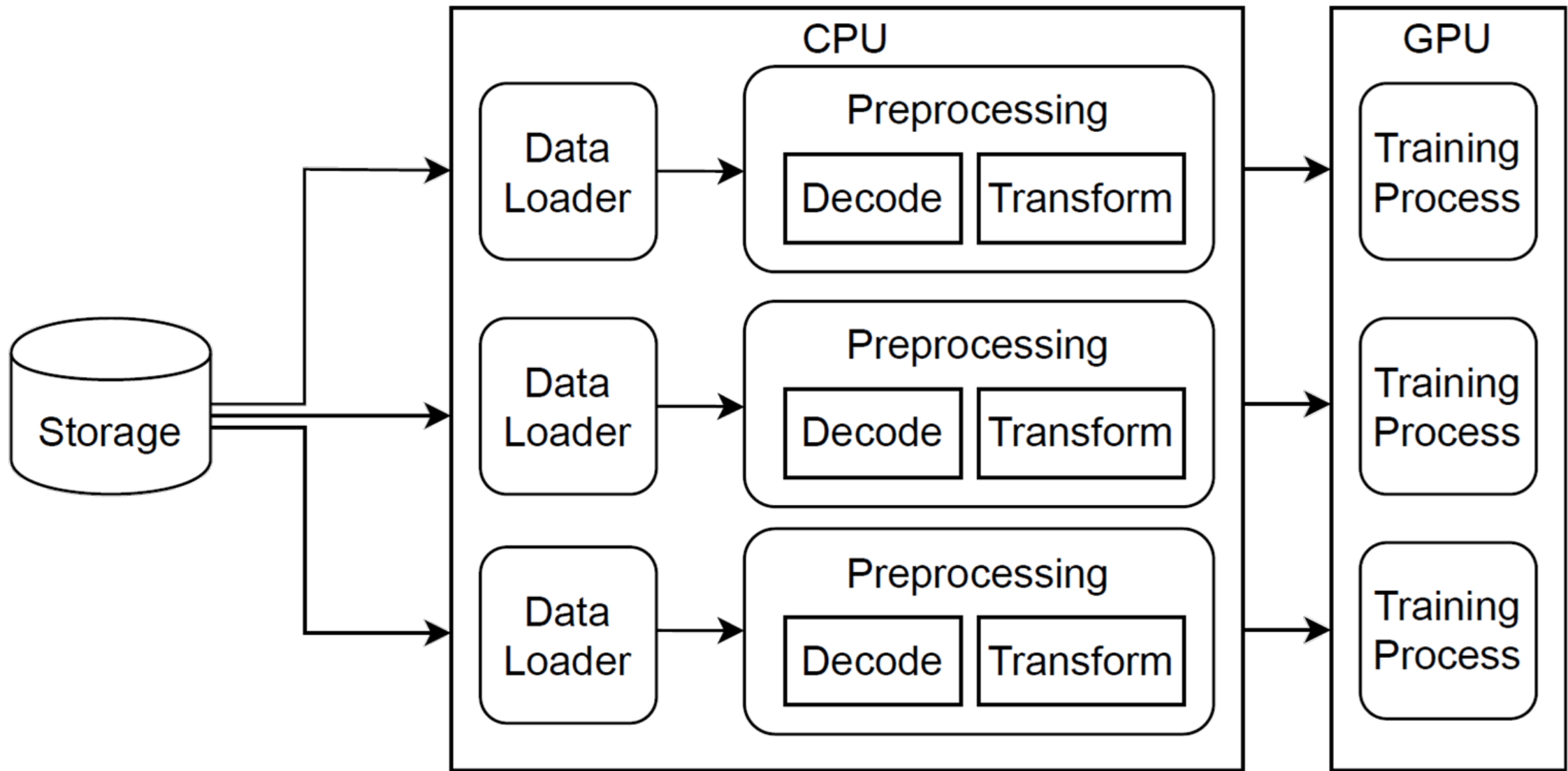
Ties Robroek, Neil Kim Nielsen, Pinar Tözün.



conventional journey of data while training



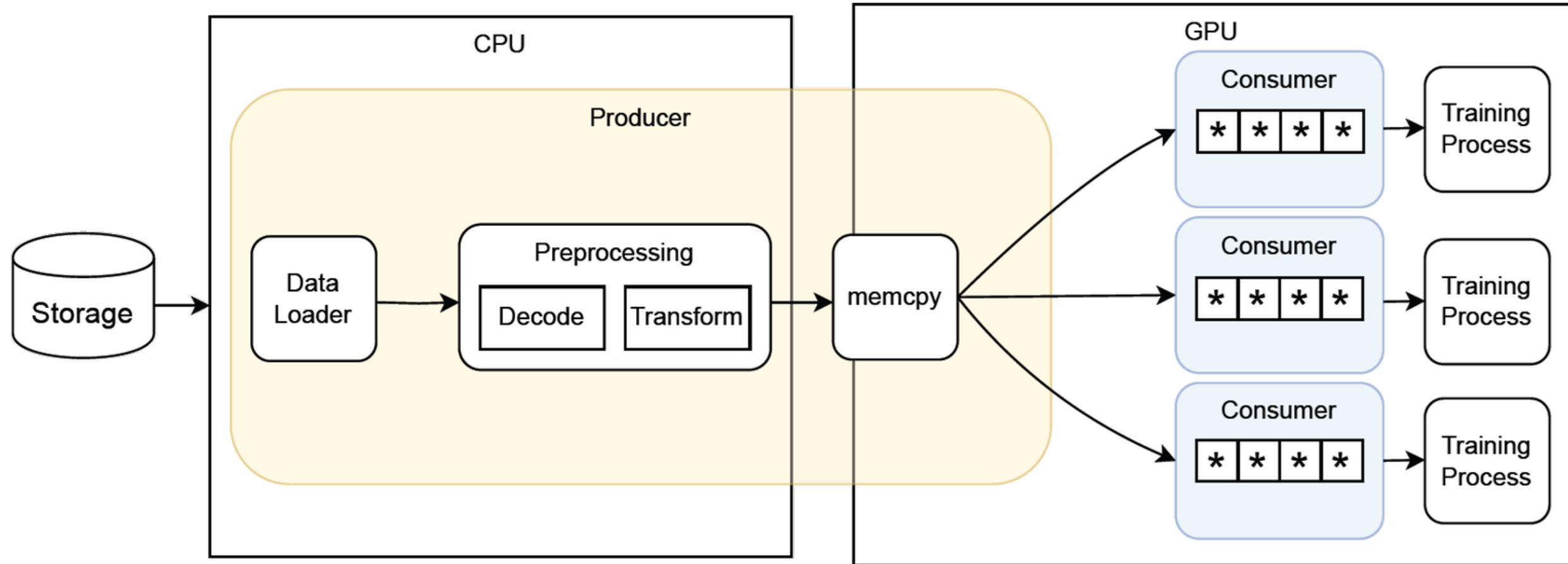
data journey in collocated training



redundant work & memory use!

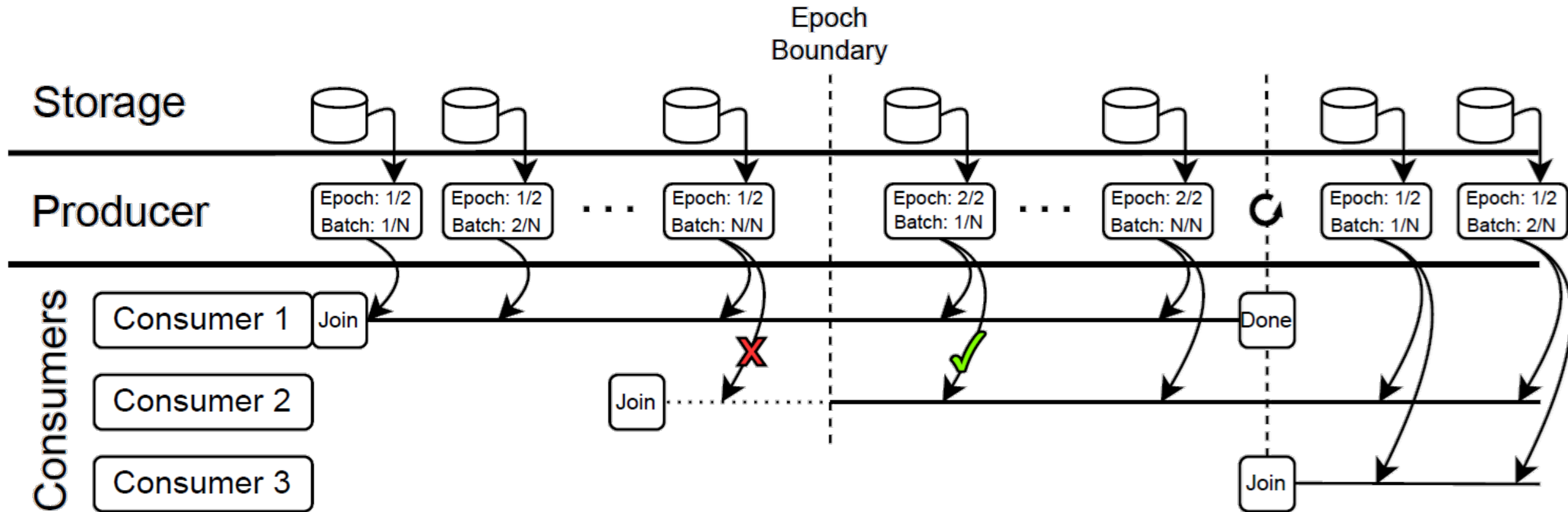
data sharing for collocated training

TensorSocket



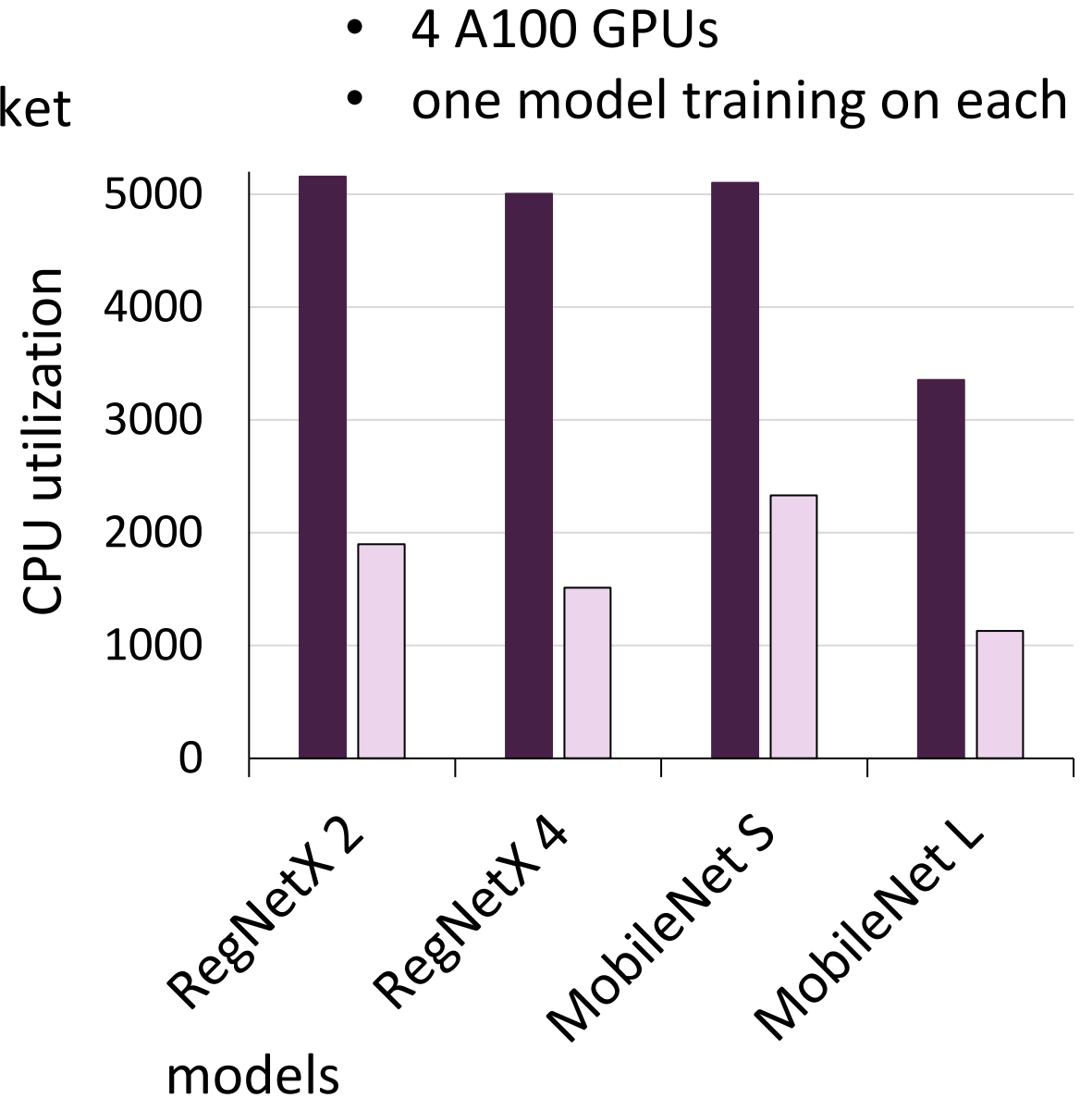
eliminates redundant work on CPUs!

data loading server



consumers don't have to be in perfect sync.

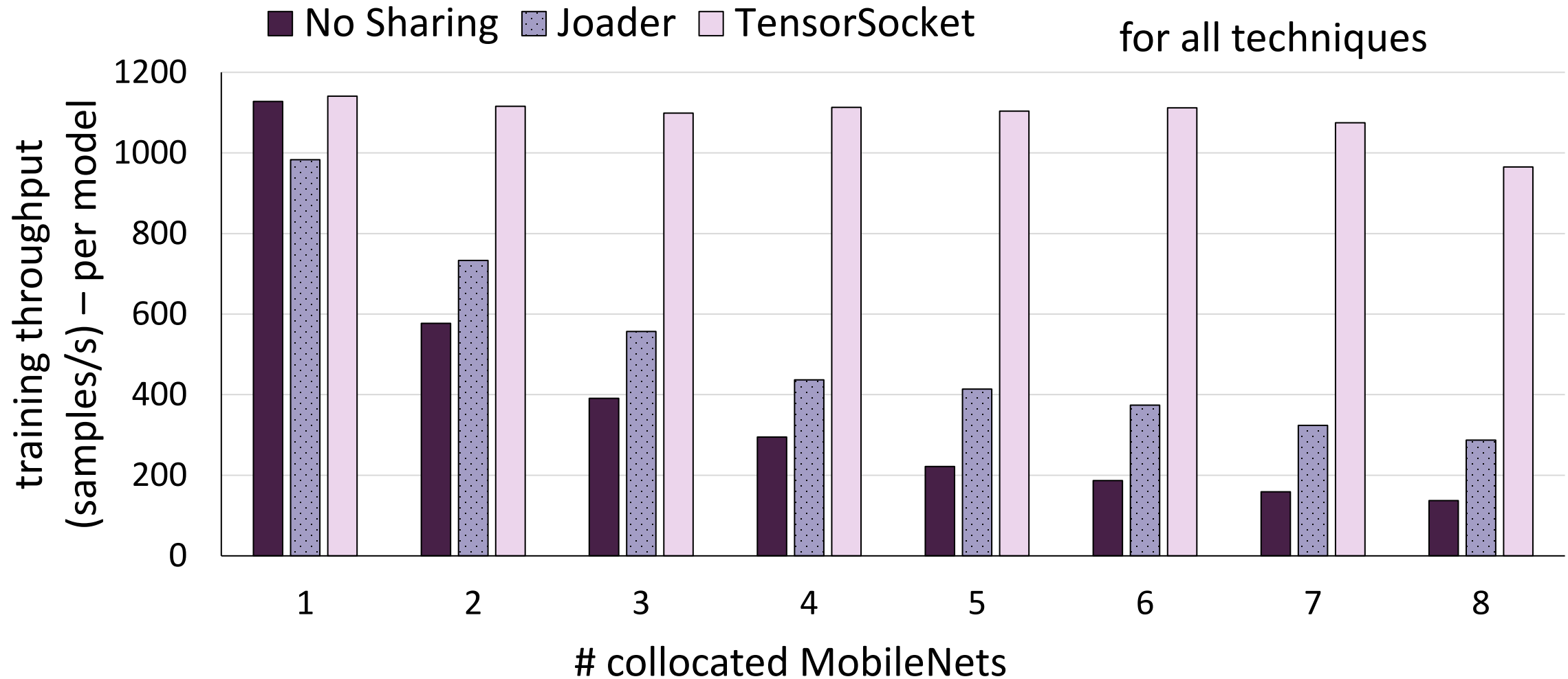
impact of data sharing



higher overall throughput & reduced CPU need!

comparison to other techniques

- on a single H100 GPU (80GB)
- CPU resources are the same for all techniques



TensorSocket maintains throughput even under heavy collocation.

sharing for deep learning training

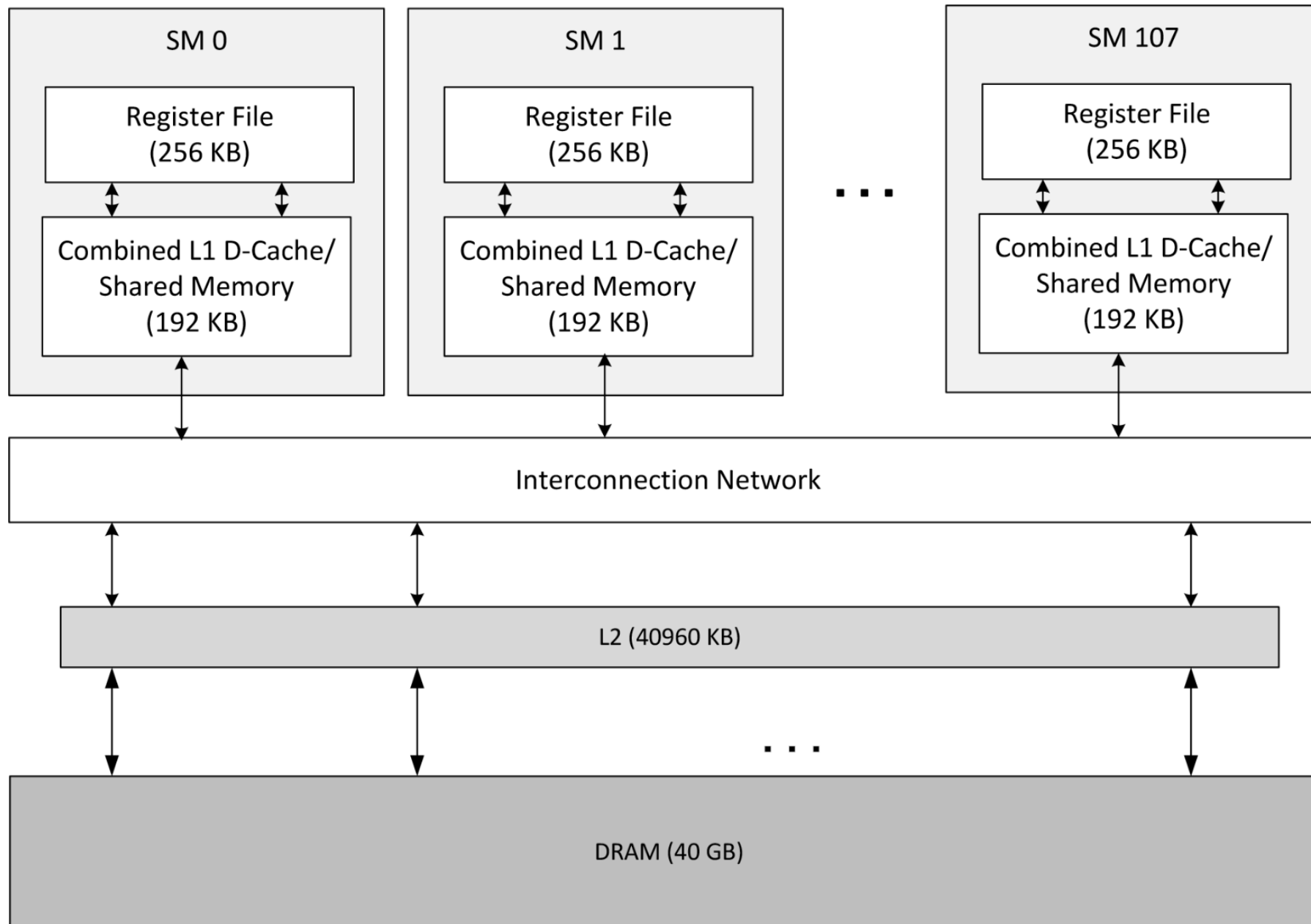
thank you!

- not all training needs all the resources of a single GPU
- collocation on GPUs benefits when the aggregate compute & memory needs of the collocated training runs fit in the GPU
 - MPS performs the best overall
 - MIG is the only option if more strict separation is needed
- data sharing can further reduce hardware resource needs while increasing training throughput

need to build schedulers that incorporate resource & data sharing for deep learning!

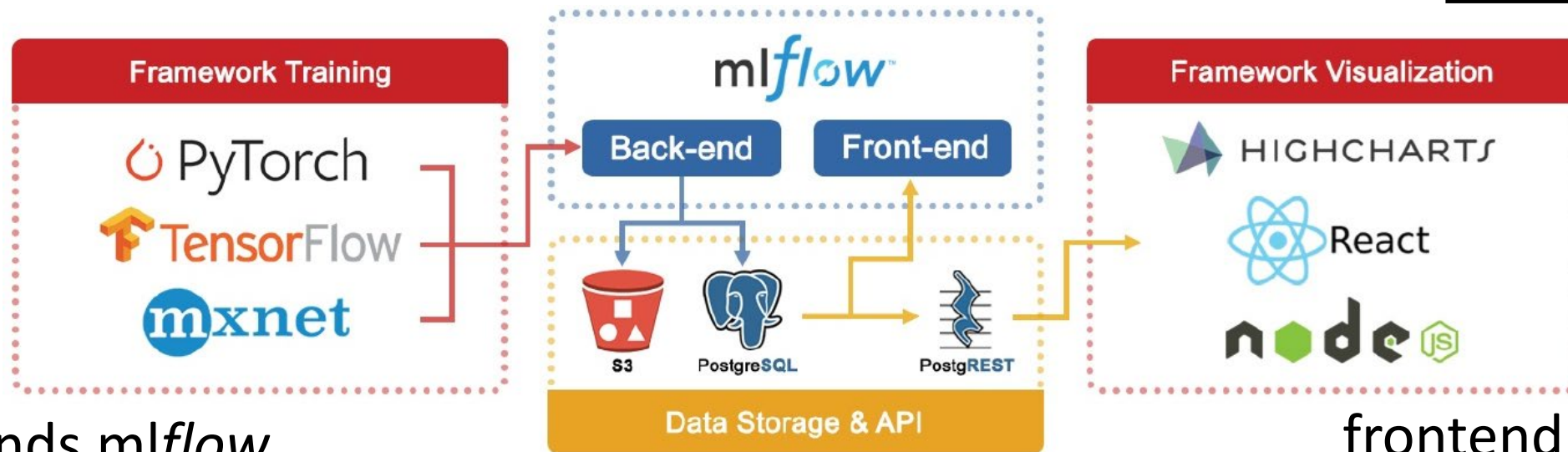
backup

A100



max threads
per SM = 2048

radT



- extends mlflow
- incorporates collocation
- allows easy, extensible, and scalable tracking of hardware metrics on CPUs & GPUs
 - listeners for monitoring (dcm, nvidia-smi, top) & profiling (nsys, ncu, pytorch profiler) tools

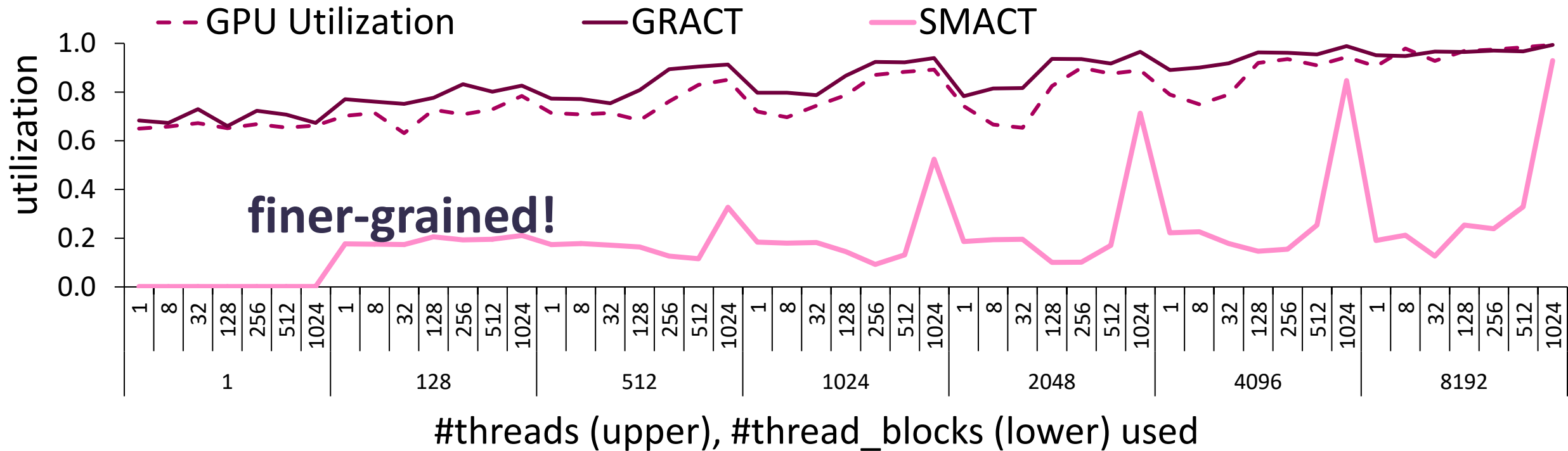
frontend for
data exploration

**used by several members of our group including data scientists
for systematic benchmarking of deep learning training**

Robroek et al. "[Data Management and Visualization for Benchmarking Deep Learning Training Systems](https://arxiv.org/abs/2302.01481)", DEEM 2023
<https://github.com/Resource-Aware-Data-systems-RAD/radt> & <https://www.youtube.com/watch?v=oaGfzYjKJ1Q>

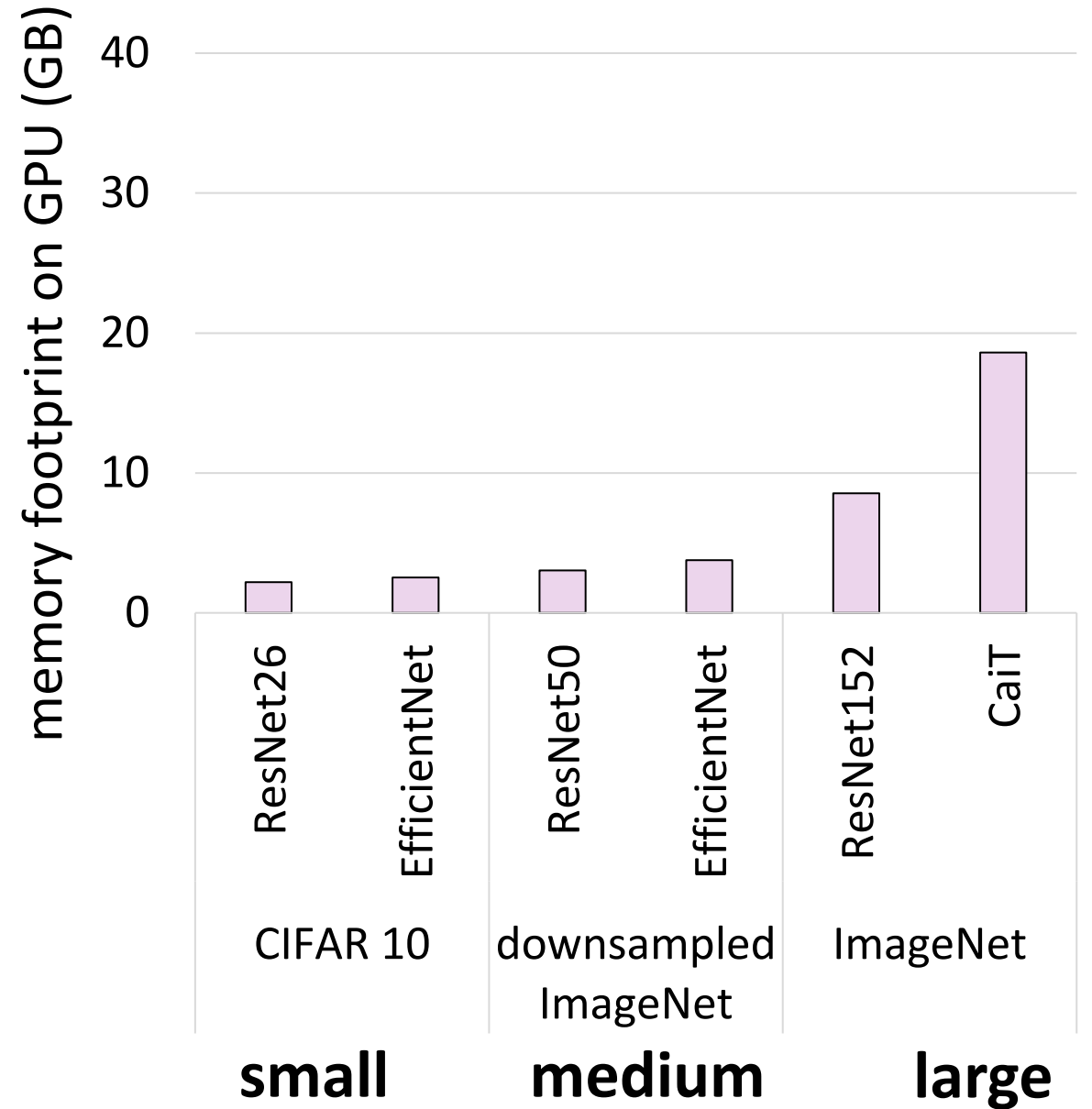
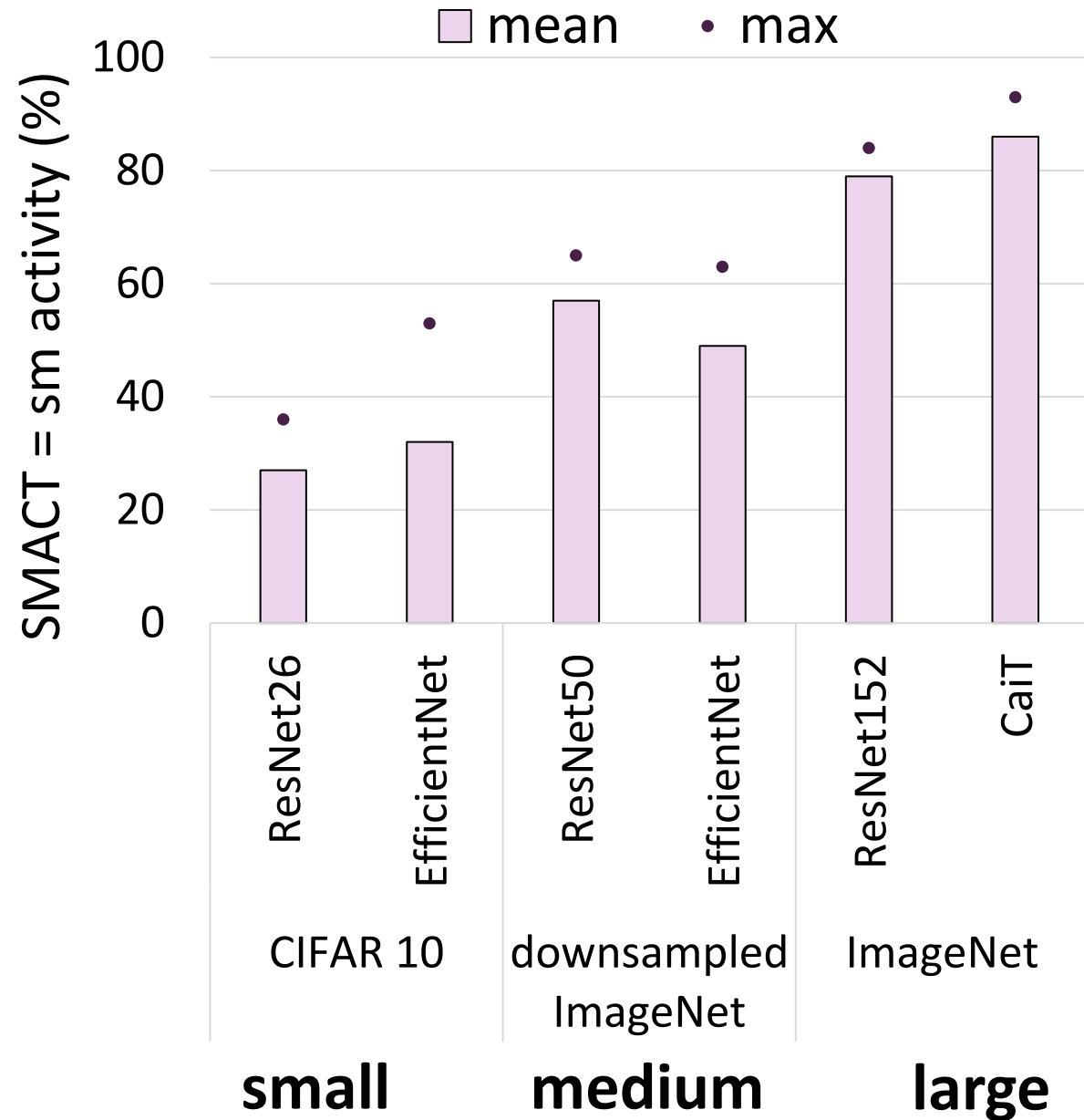
GPU utilization

- **GPU utilization:** % of time one or more kernels were executing on the GPU
- **GRACT:** % of time any portion of the graphics or compute engines were active
- **SMACT:** the fraction of active time on an SM, averaged over all SMs = streaming multiprocessor

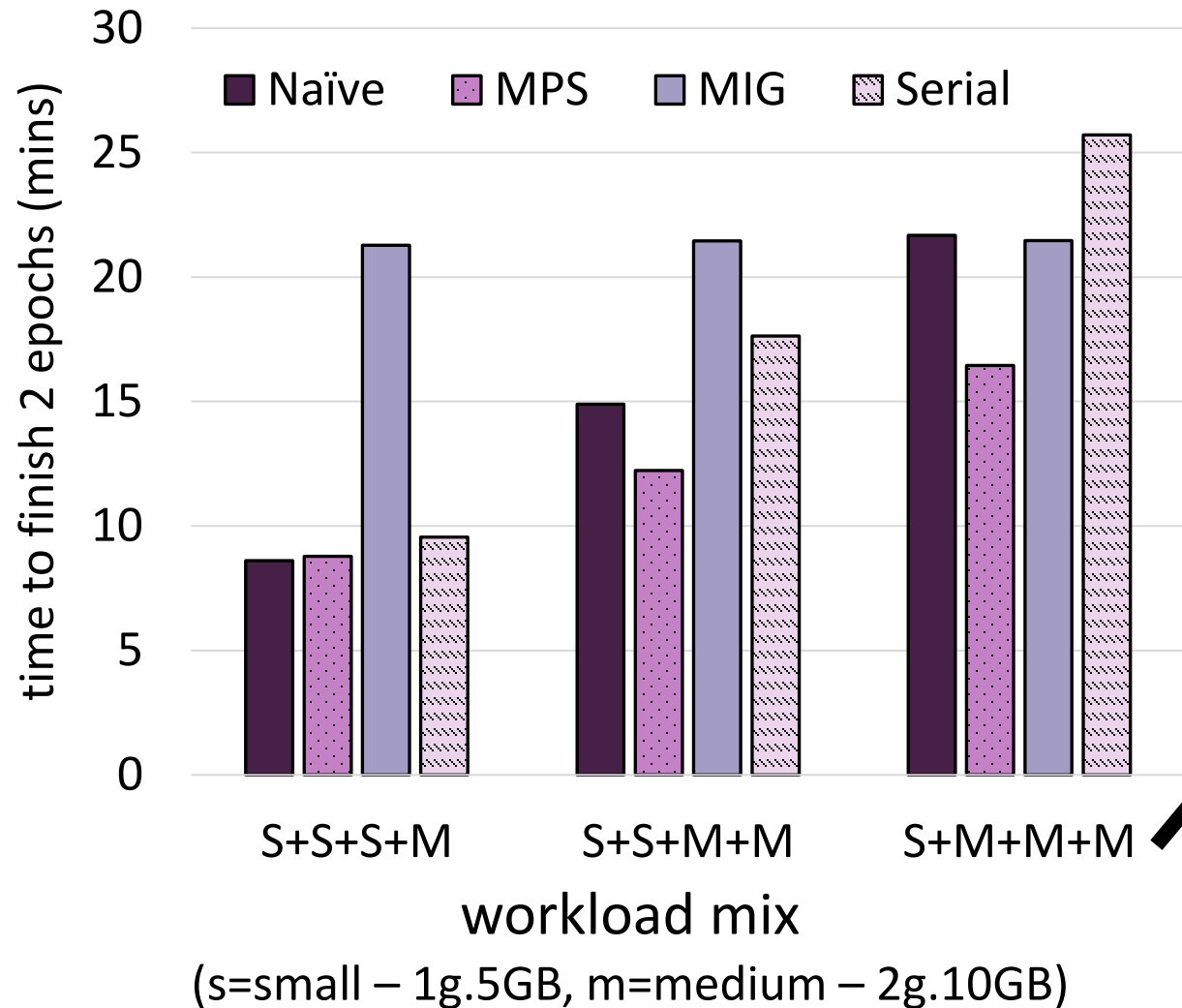


coarse-grained GPU utilization metrics could be misleading!

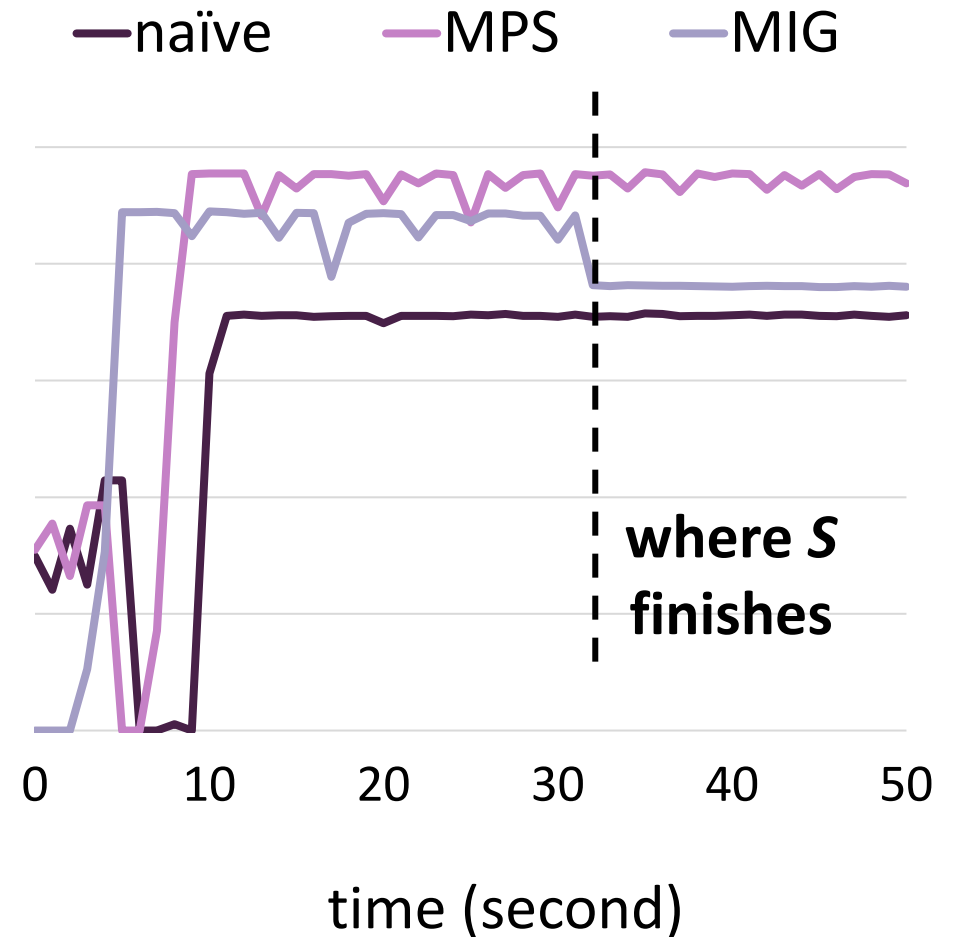
hardware utilization without collocation



mixed workloads: all compute-heavy



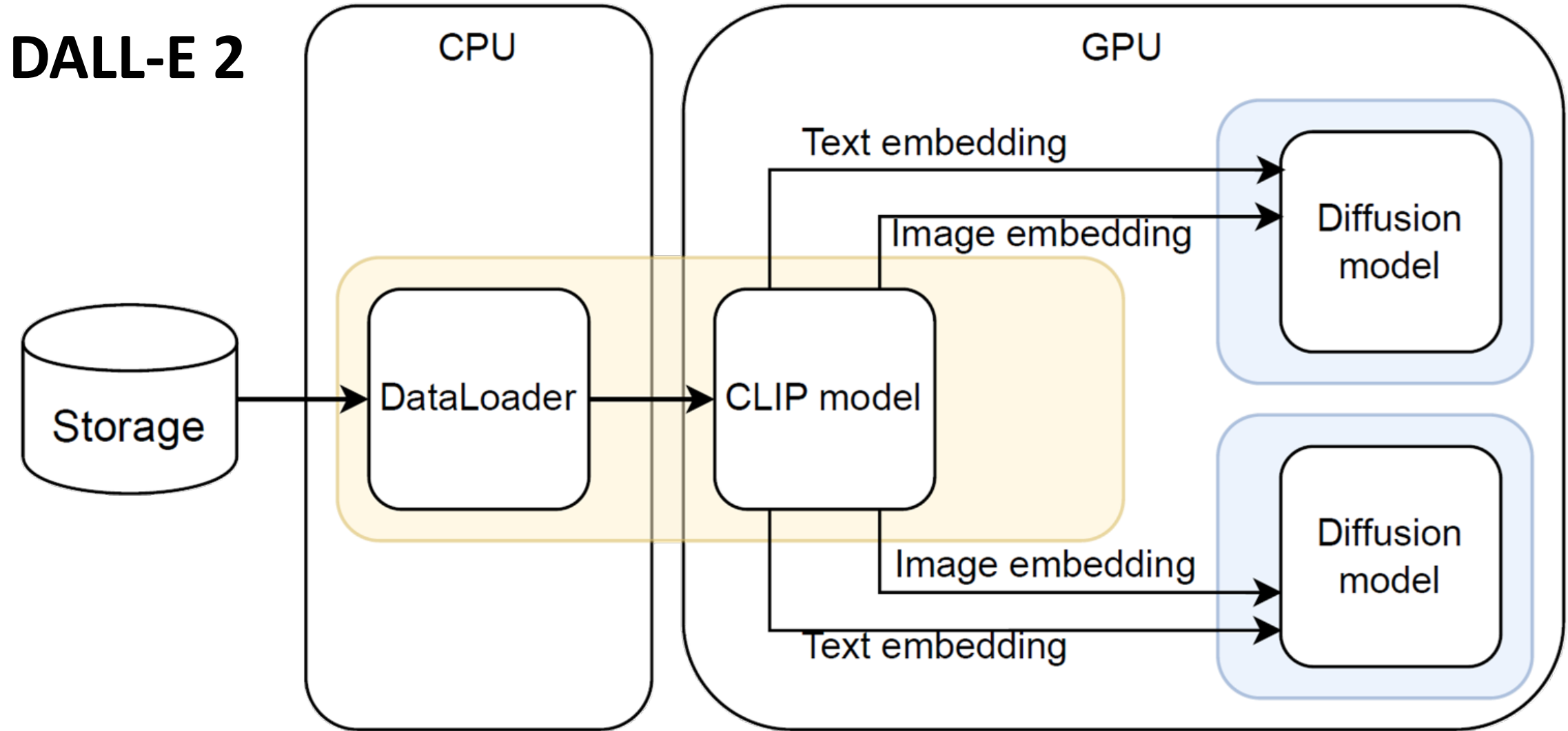
GPU util. (sm activity)



with MPS → the small training is for free near the medium one

with MIG → isolation at the cost of inflexible resource distribution

data sharing for collocated training



can also reduce work on GPUs!

team **RAD** - resource-aware data systems



Ties
Robroek



Ehsan
Yousefzadeh-Asl-Miandoab



Robert
Bayer