HPC I/ODC
I/O in the Data Center
Workshop

AI for Science and Exascale

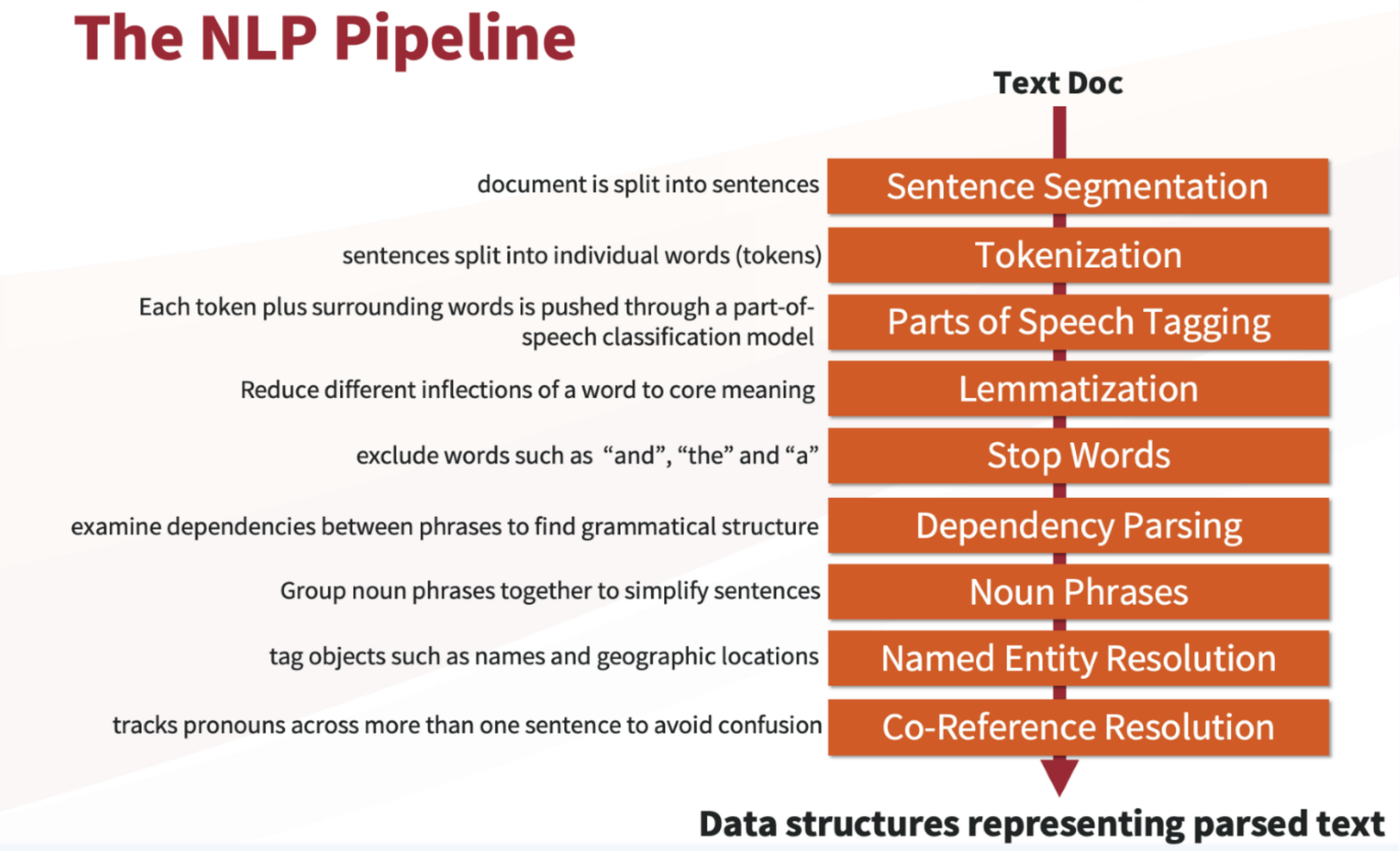
Large Language Models
The Rise of Data

Jean-Thomas Acquaviva
jtacquaviva@ddn.com

Data Challenges usually associated with AI for Science

- **Ingestion Challenge: Evolving applications and acquisition devices generate more and more diverse data**
  - Large volume of data to ingest
  - Heterogenous data types and file sizes (KBs to TBs), challenging data and metadata patterns
  - Diverse and complex data pipelines: AI and DL , IO vs Mem, GPU vs CPU, distributed computing
  - No single protocol for data acquisitions / Transfer
- **Logistic Challenge: Complex data movements stumble on siloed architectures**
  - Significant waste of personnel and instrument time for data management
  - Challenges exacerbated at-scale, data bottlenecks severely cripple AI effort
  - On-prem, cloud and hybrid considerations
- **Legal Challenge: data may have specific requirements**
  - Ethical aspect, responsibility, data bias,
  - Ensure integrity and availability for repeatability, collaboration and innovation
  - Enforce data privacy, ownership, auditable access controls,

Large Language Models have a specific data consideration



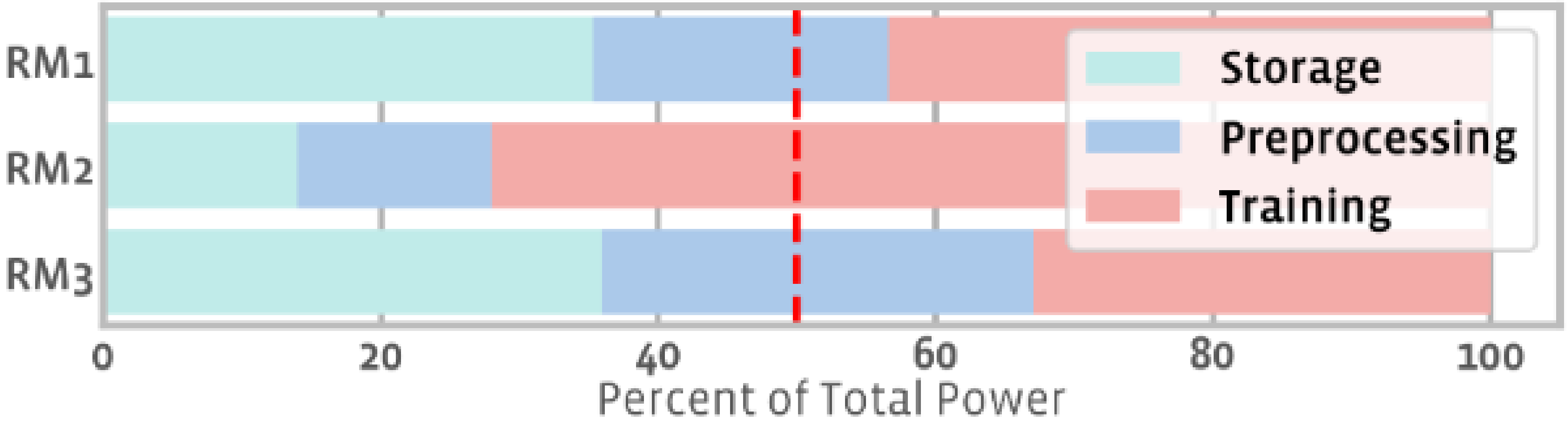
Multiple stages

Different processing / data requirements

- Complex process: parallelization, modifications are costly
- AI is building its legacy codes

LLM and Storage: a looming issue

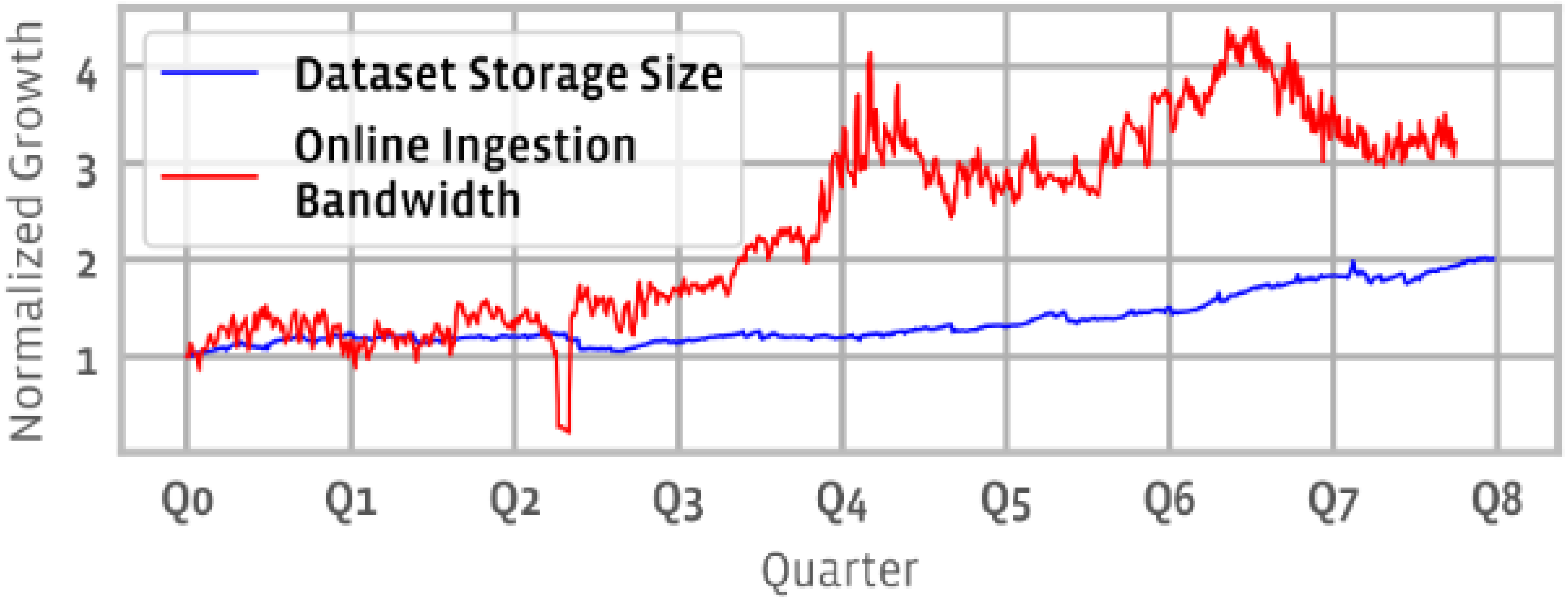
Data management in 3 Facebook use cases:
storage + data ingestion consumes more power than training



LLM and Storage: a looming issue

Bandwidth requirement is growing faster than capacity.

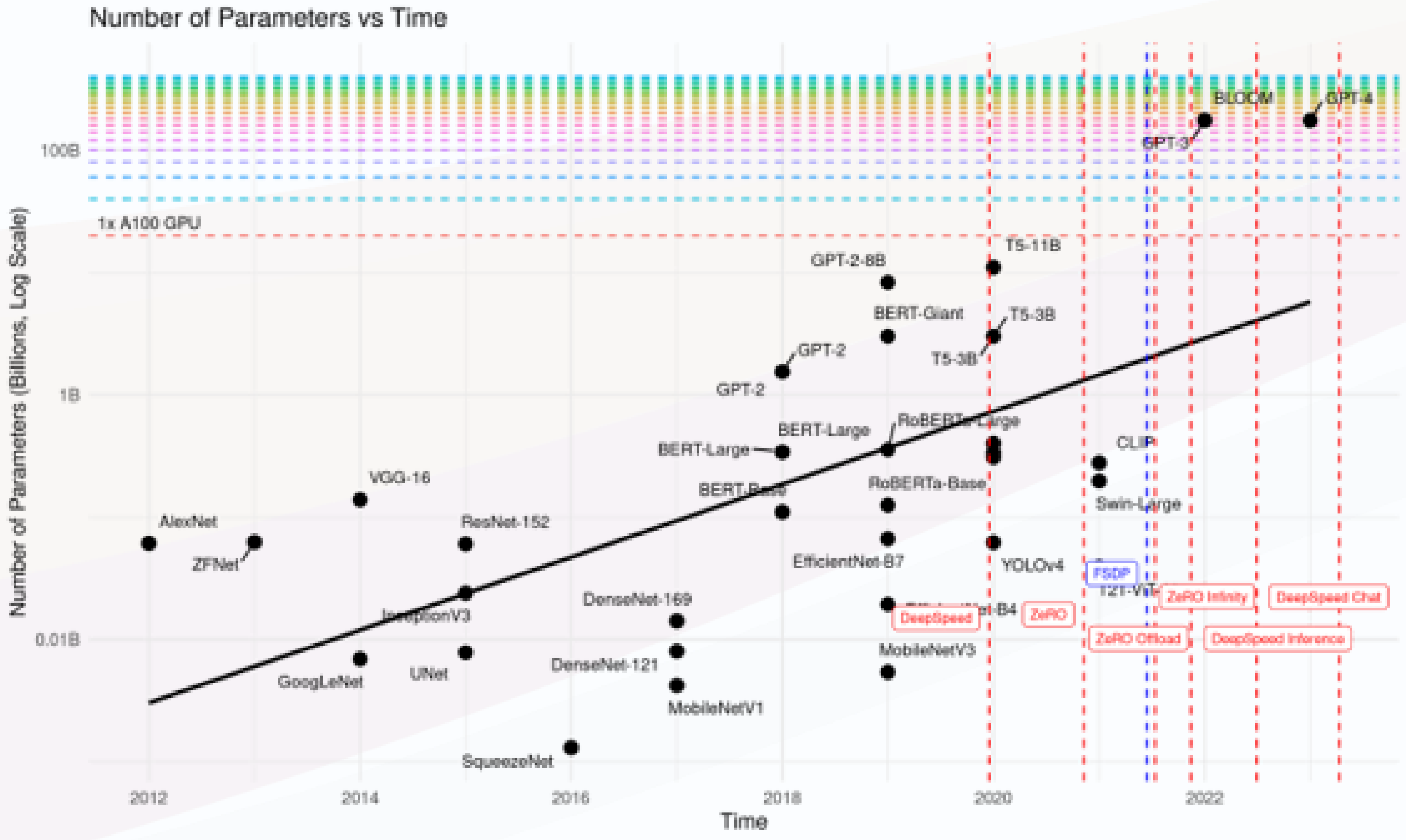
- Bandwidth x4 over 2 years
- Capacity x2 over 2 years



ML Perf has a SIG focused on Storage: https://mlcommons.org/en/groups/research-storage

- Focused for the moment on training
- Key people McGill University

LLM Evolution of the number of parameters over time



Trend correspond to 10 years

Log scale on Y

In 3 years:

- model size x1000
  - 1 order of magnitude per year
- GPU memory x5

LLM Memory Consumption

Memory pressure depends on model size Ψ expressed in number of parameters

- **Parameters, half precision, 2xΨ Byte**
- **Gradient, half precision, 2 x Ψ Byte**
- **Optimizers states, 3 states single precision 12 x Ψ Byte**

The total amount of memory needed:

Byte needed = 16 x number of parameters

A 17B parameters model = 272 GB of memory: Not available on the state-of-the-art H100 GPU

LLM Memory Wall

Within 3 years models will be 100s of trillion of parameters

- To accommodate model growth, GPU will need 100s of TB
- Difficult for a single device
- Achievable for 100s of GPUS

LLM Parallelization Scheme

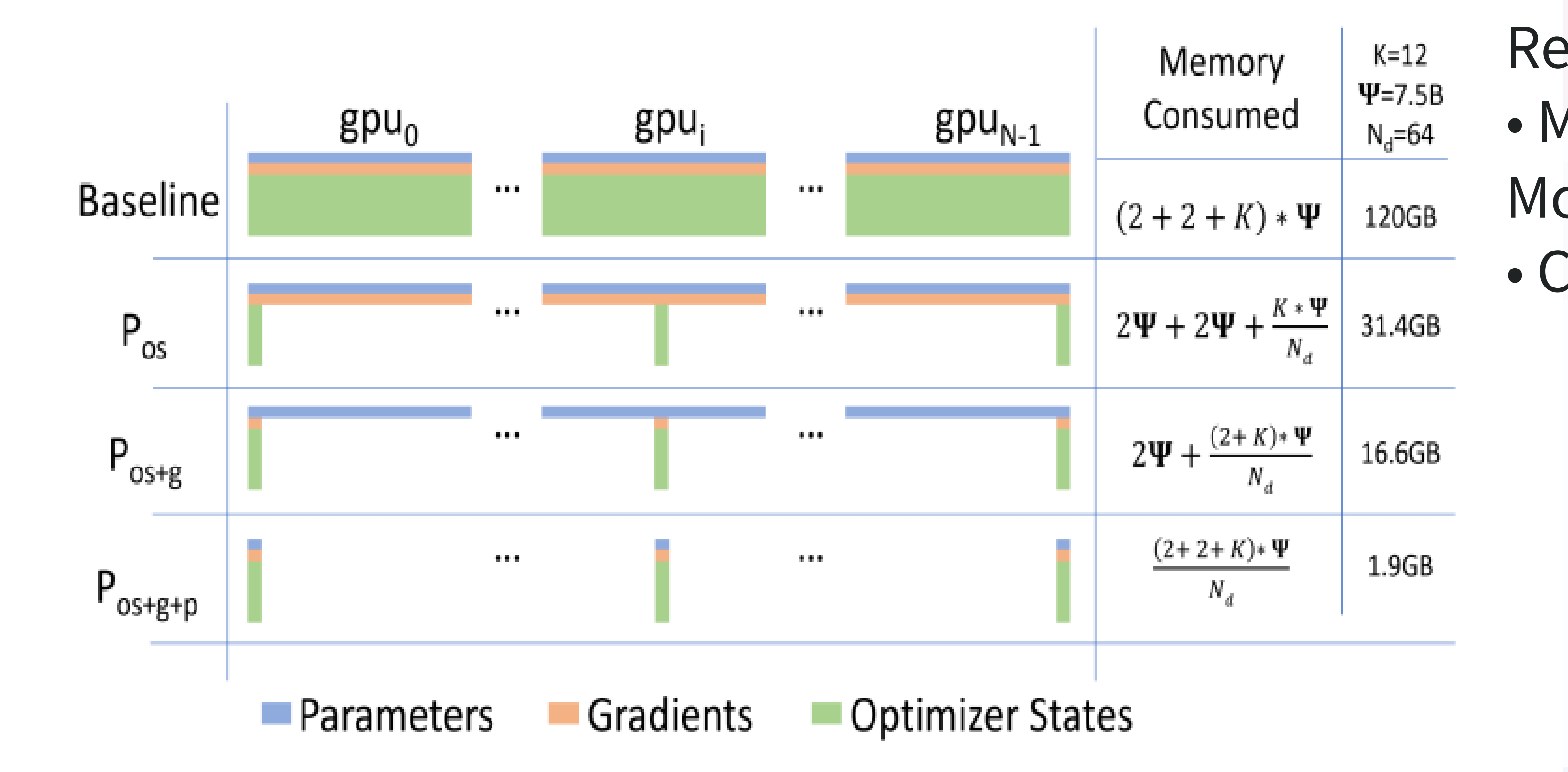
Adding hardware resources to overcome current limitations

More GPUs = More memory

- **Data Parallelism**, duplicate the model with each GPU memory. Does not solve memory issue, accelerate training
- **Model Parallelism**, split the model vertically. Reduce memory footprint by the degree of parallelism, generates lots of communications. Does not scale beyond a DGX (5% of efficient is spanned over multiple DGX)
- **Pipeline Parallelism**, split model horizontally. Complex to implement. Generate synchronization and overhead

LLM Memory Offloading: Zero [2020]

ZeRO: framework from Microsoft interleaving parallelization schemes to minimize memory footprint (at the cost of some communication overhead)



Reduction of memory footprint

- Mixture of Data Parallelism, Model and Pipeline parallelism
- Cap communication overhead

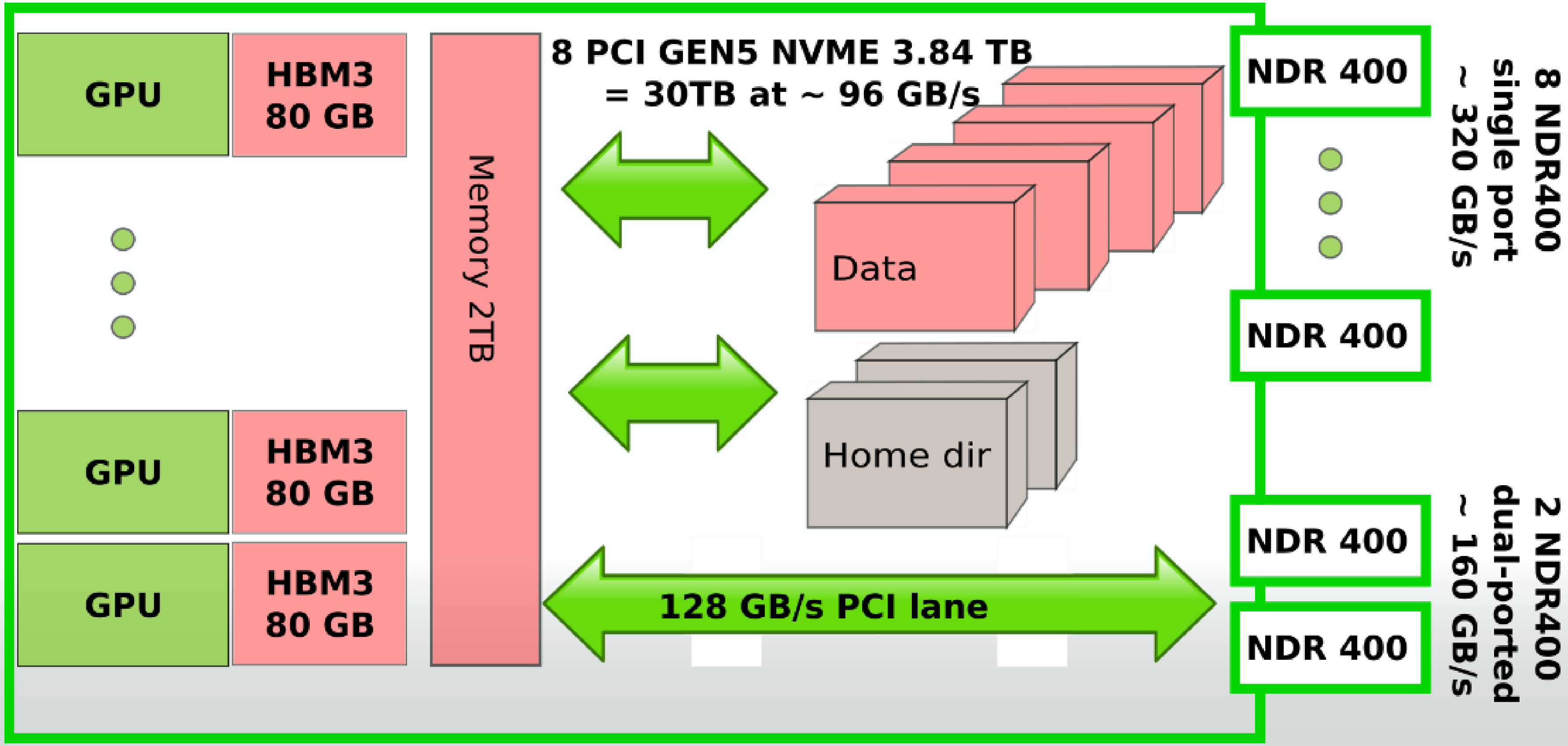
LLM Memory Offloading: ZeRO Infinity [2021]

The resurrection of out-of-core computing

Zero to Infinity, extension of the ZeRO model

Model's parameters, gradient and optimizers states are not offloaded on remote GPUS on but on CPU memory, local storage and remote storage

DGX Memory Hierarchy



Two memory levels

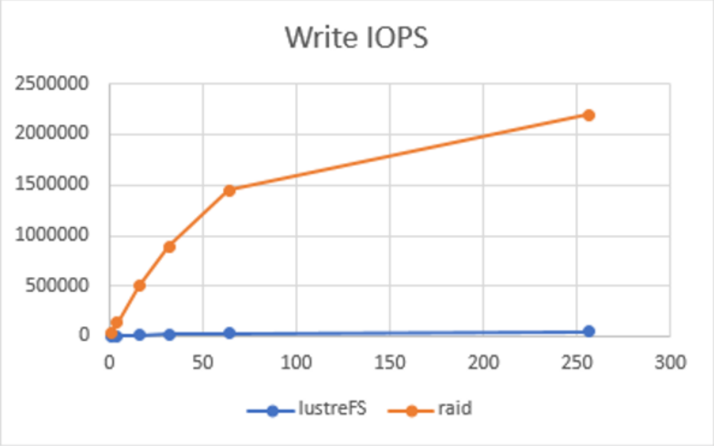
- *80 GB per GPU*
- *2TB shared with CPU*

Two storage levels

- *PCI Gen 5 local NVMes*
- *2 NDR400 IB slots for network attached storage.*

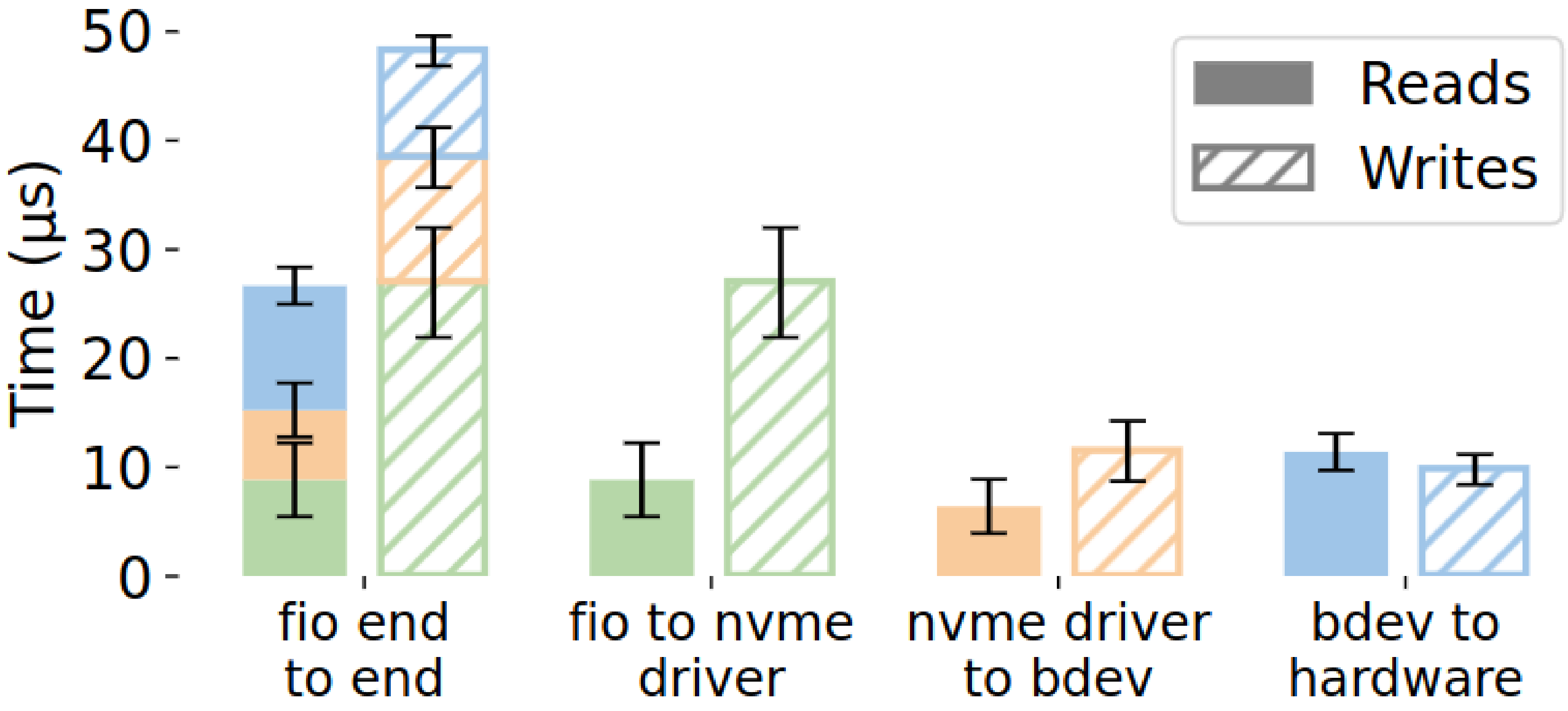
Storage micro-benchmarking

Comparative bandwidth measurements on a DGX platform. Using FIO with threads number ranging from 1 to 256 and large payload.
The Lustre delivers x5 the read performance and x2 the write performance of the local storage.



Comparative latency measurements on a DGX platform. Using FIO with a threads number ranging from 1 to 256 with a small payload. Local storage delivers x5 the IOPS (IO operations per second) than Lustre and x100 the IOPS of Lustre for write operations. Lustre version 2.12 used in this experiment does not support the most recent IOPS write optimizations

Latency is mostly a software issue



28 µsec latency for a read request

- 9 µsec for NVMe driver
- 6 µsec for NVMe driver to the block device driver
- 13µsec for the block device

Software overhead (drivers) **is dominating hardware latency**.

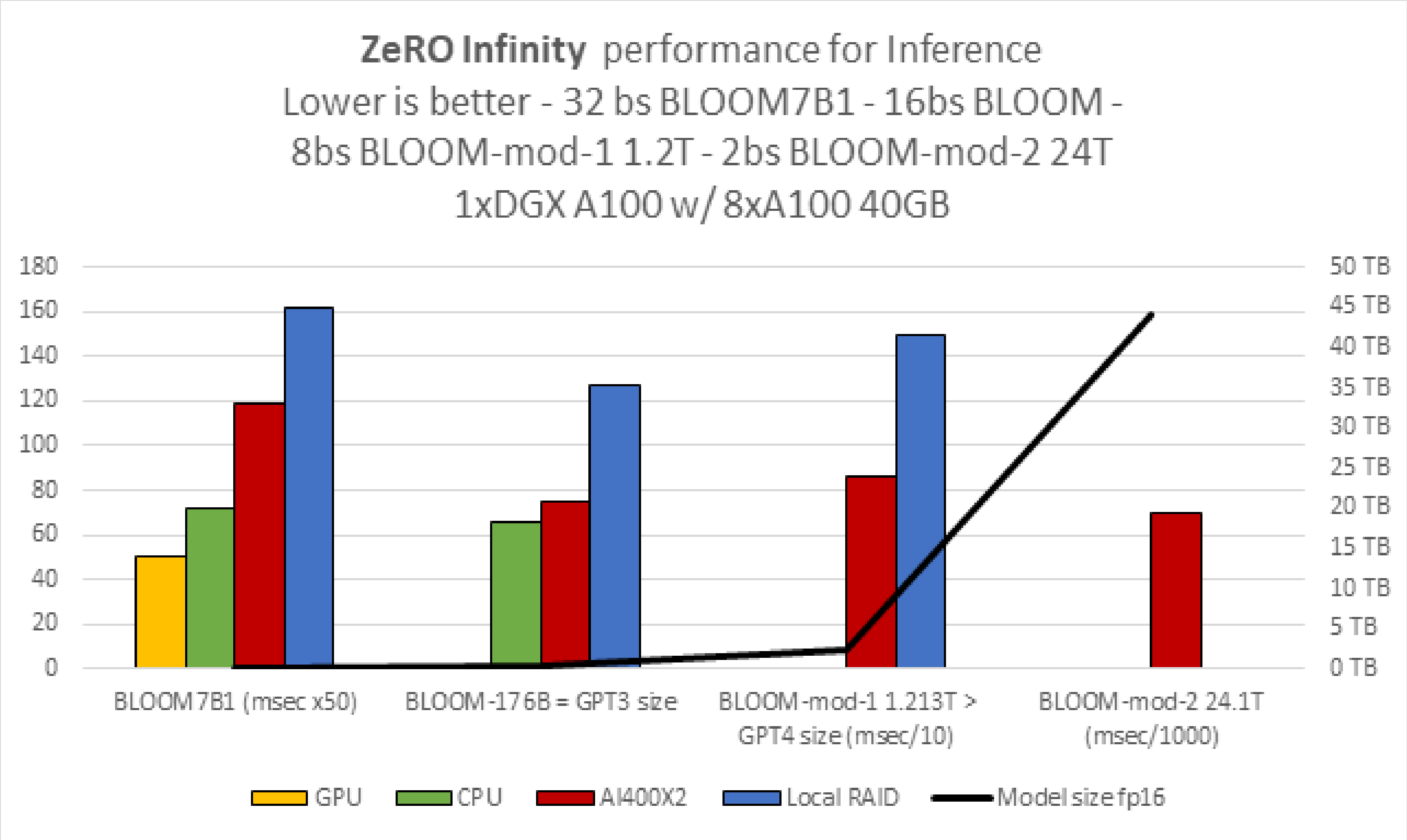
LLM Experimental Results

Using BLOOM A 176B-Parameter Open-Access Multilingual Language Model under Open-Source

Name	BLOOM 7B1	BLOOM (176B params)	BLOOM-mod-1 (1.213T params)	BLOOM-mod-2 (24.17T params)
# hidden layers	30	70	960	4800
memory hierarchy levels to host the model	GPU	GPU + CPU	GPU + CPU + local NVMe	GPU + CPU + Local NVMe + Exascaler
hidden-dim	4096	14336	10240	20480
Storage used for offloading	TODO	350 GigaBytes	2.3 TeraBytes	44 TeraBytes
Batch-size used	32	16	8	1

LLM Experimental Results (WIP)

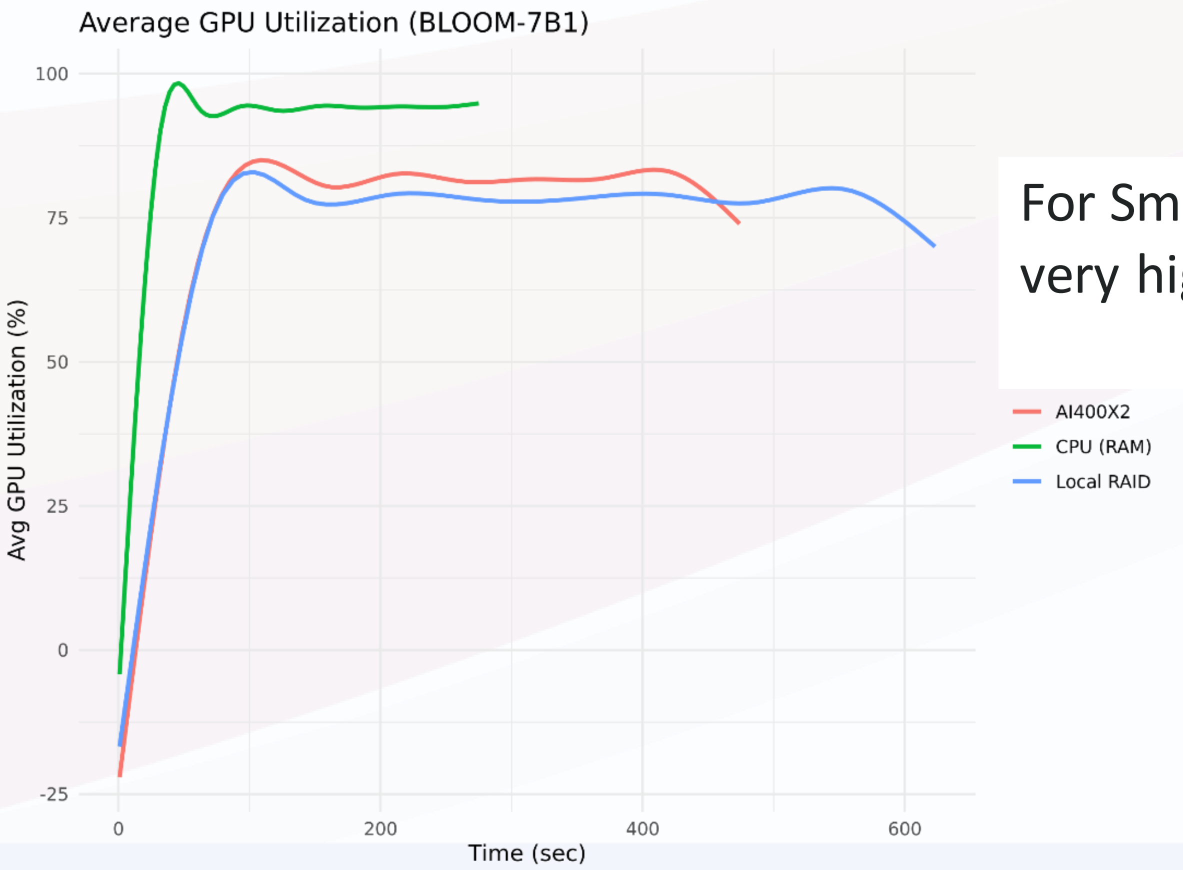
Using BLOOM A 176B-Parameter Open-Access Multilingual Language Model under Open-Source



For large models ExaScaler is competitive with CPU OffLoading Outperforming consistently Local Storage

LLM Experimental Results (WIP)

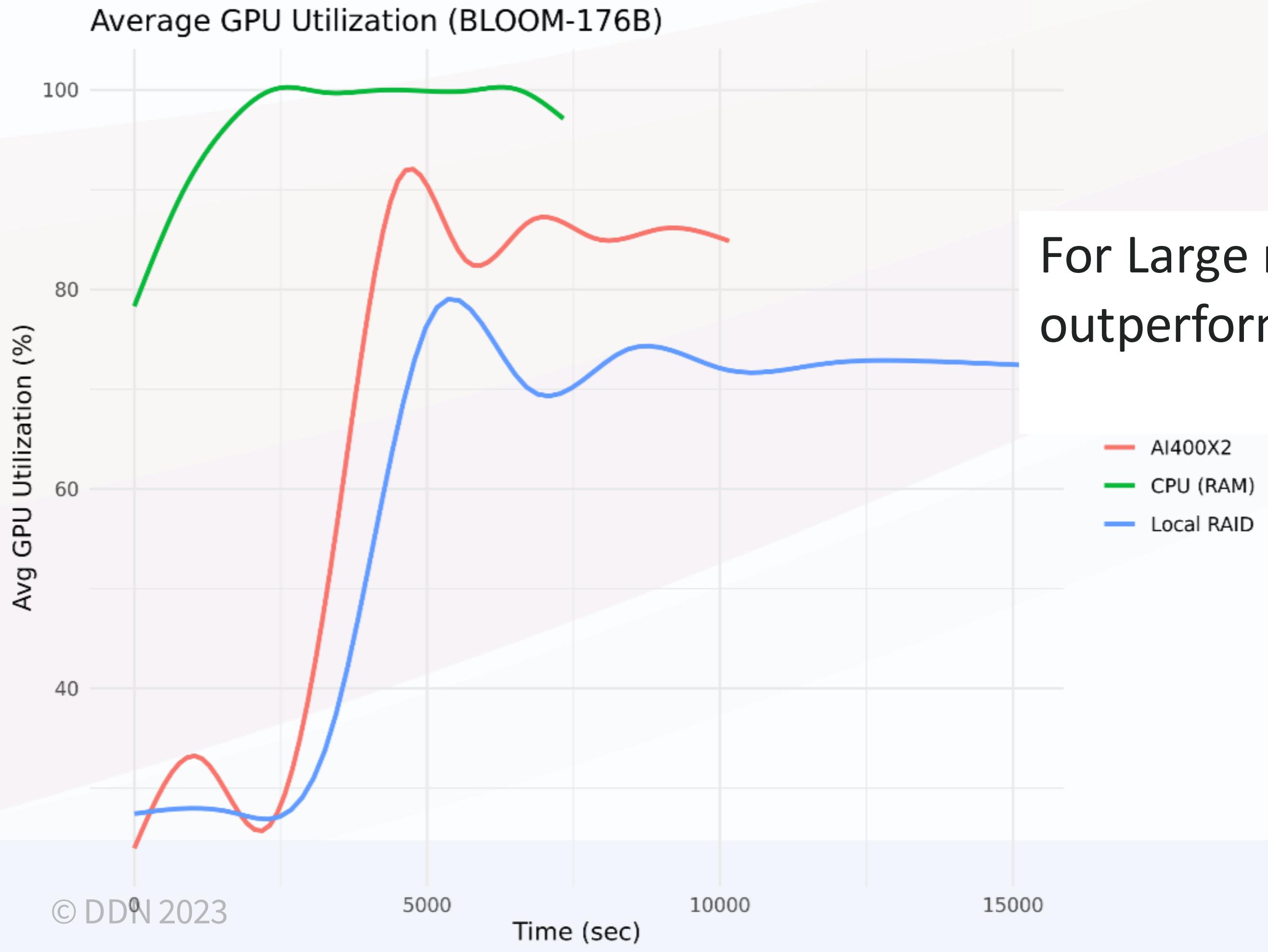
Using BLOOM A 176B-Parameter Open-Access Multilingual Language Model under Open-Source



For Small models fitting in GPU memory, GPU efficiency is very high. Exascaler outperforms local RAID

LLM Experimental Results (WIP)

Using BLOOM A 176B-Parameter Open-Access Multilingual Language Model under Open-Source

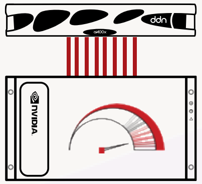


For Large models not fitting in GPU, Exascaler outperforms local RAID

What's next: Big Models vs Big Data

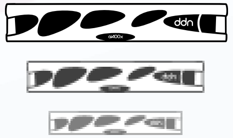
- **Offloading of models' data to the ExaScaler alleviates complexity and delivers a constant 85% GPU efficiency**
- ExaScaler scales seamlessly to hundreds of PetaByte, thus removing memory issue from the design consideration and complexity equation.
- Optimal model accuracy is reached by a balance between model size, volume of data available, amount of processing power devoted to training
  - Accuracy converged faster on the model size axis
  - Current race to bring to market the highest-accuracy models has led to overlooking the data size aspect
  - We expect the competition to displace in the field of data set size, thus increasing the need for data management solution, life cycle orchestration

DDN versatile software

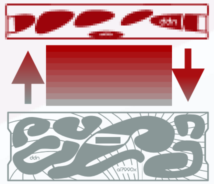


MAX PERFORMANCE

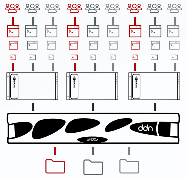
COMPLETE WORKFLOWS



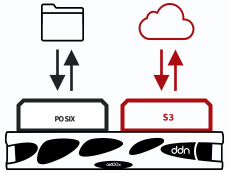
LIMITLESS SCALING



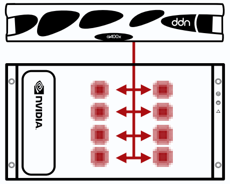
DATA MANAGEMENT



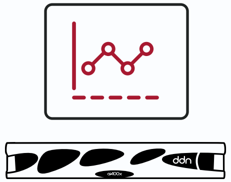
MULTI-TENANCY



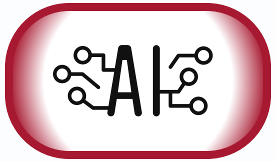
DATA SERVICES



GPUDIRECT TO STORAGE



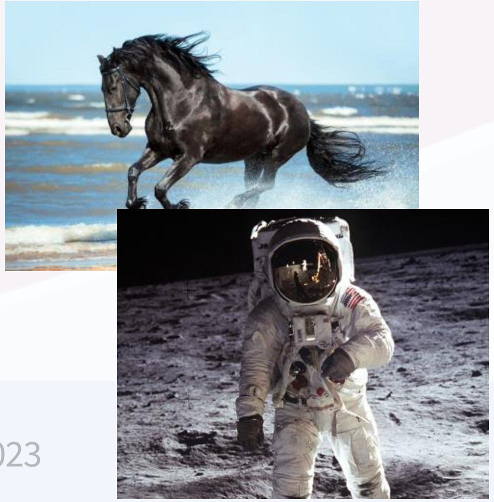
REAL-TIME ANALYTICS



AI OPS INTEGRATION

Future Proof AI solution

High-speed network



GPU training
On cached dataset

GPU inference
On streamed dataset

References

- [BRO20] BROWN, Tom, MANN, Benjamin, RYDER, Nick, *et al.* Language models are few-shot learners. *Advances in neural information processing systems*, 2020, vol. 33, p. 1877-1901. https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf
- [SCA22] SCAO, Teven Le, FAN, Angela, AKIKI, Christopher, *et al.* Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2022. https://arxiv.org/pdf/2211.05100
- [RAJ20] RAJBHANDARI, Samyam, RASLEY, Jeff, RUWASE, Olatunji, *et al.* Zero: Memory optimizations toward training trillion parameter models. In : *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*. IEEE, 2020. p. 1-16. https://arxiv.org/pdf/1910.02054.pdf%3E
- [ZHA22] ZHAO, Mark, AGARWAL, Niket, BASANT, Aarti, *et al.* Understanding data storage and ingestion for large-scale deep recommendation model training: Industrial product. In : *Proceedings of the 49th Annual International Symposium on Computer Architecture*. 2022. p. 1042-1057. https://arxiv.org/pdf/2108.09373.pdf
- [MLP22] ML Perf Storage, https://mlcommons.org/en/groups/research-storage/
- [BAL23] Characterizing I/O Patterns in Machine Learning ACM Workshop on Challenges and Opportunities of Efficient and Performant Storage Systems. O. Balmau. https://sigmodrecord.org/publications/sigmodRecord/2209/pdfs/09_Dbrainstorming_Blamau.pdf
- [UMO23] UM, Taegeon, OH, Byungsoo, SEO, Byeongchan, *et al.* FastFlow: Accelerating Deep Learning Model Training with Smart Offloading of Input Data Pipeline. *Proceedings of the VLDB Endowment*, 2023, vol. 16, no 5, p. 1086-1099. https://www.vldb.org/pvldb/vol16/p1086-um.pdf
- [RAJ21] RAJBHANDARI, Samyam, RUWASE, Olatunji, RASLEY, Jeff, *et al.* Zero-infinity: Breaking the gpu memory wall for extreme scale deep learning. In : *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. 2021. p. 1-14.
- [RAJ22] RAJBHANDARI, Samyam, LI, Conglong, YAO, Zhewei, *et al.* Deepspeed-moe: Advancing mixture-of-experts inference and training to power next-generation ai scale. In : *International Conference on Machine Learning*. PMLR, 2022. p. 18332-18346. https://proceedings.mlr.press/v162/rajbhandari22a/rajbhandari22a.pdf
- [KAP20] KAPLAN, Jared, MCCANDLISH, Sam, HENIGHAN, Tom, *et al.* Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. https://arxiv.org/pdf/2001.08361.pdf
- [DGX23] NVIDIA DGX SuperPOD: DDN AI400X2 Appliance, Reference Architecture https://www.ddn.com/wp-content/uploads/2023/01/DDN-A3I-AI400X2-NVIDIA-DGX-A100-SuperPOD-Reference-Architecture.pdf
- [HAZ18] HAZELWOOD, Kim, BIRD, Sarah, BROOKS, David, *et al.* Applied machine learning at facebook: A datacenter infrastructure perspective. In : *2018 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. IEEE, 2018. p. 620-629. https://research.facebook.com/file/904032783795098/hpca-2018-facebook.pdf
- [ROL23] Is Bare-metal I/O Performance with User-defined Storage Drives Inside VMs Possible? ACM Workshop on Challenges and Opportunities of Efficient and Performant Storage Systems. S. Rolon, O. Balmau https://drive.google.com/file/d/1rnd76S0bttLBc6fWs6NUvR2dpHUVZ-zT/view?usp=share_link

Thank You!

Questions?