AI Training Clusters: Challenges @ DC Scale

Opportunities for Open Programmable Infrastructure

Nic Viljoen
Network Systems Engineer

Meta

Thanks to Dheevatsa Mudigere & Whitney Zhao

Software-Hardware Co-design for Fast and Scalable Training of Deep Learning Recommendation Models

Challenges and Opportunities in DC scale AI Cluster Design Meta case study

Agenda

01 Model Growth In AI Training

02 Training Systems Today

03 Upcoming Challenges

04 How HW/SW Co-design can help

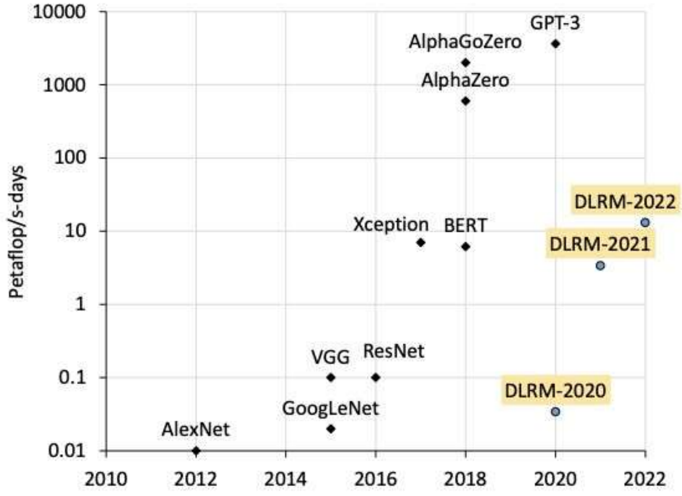
05 OPI & the SW stack

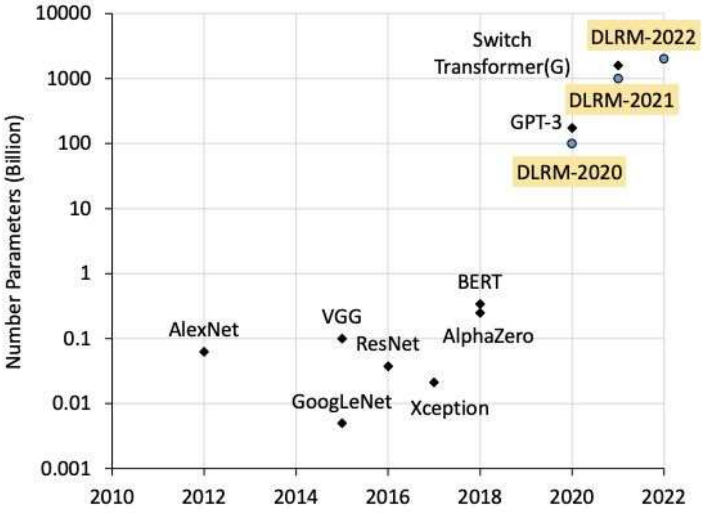
06 Summary

Model Growth

Complexity & Size is Growing exponentially

- Recommendation models pressure HW in different ways to most traditional AI workloads.
- Model parameter growth means memory speed & capacity are important to performance
- Size of models necessitates **synchronous** multi-node processing





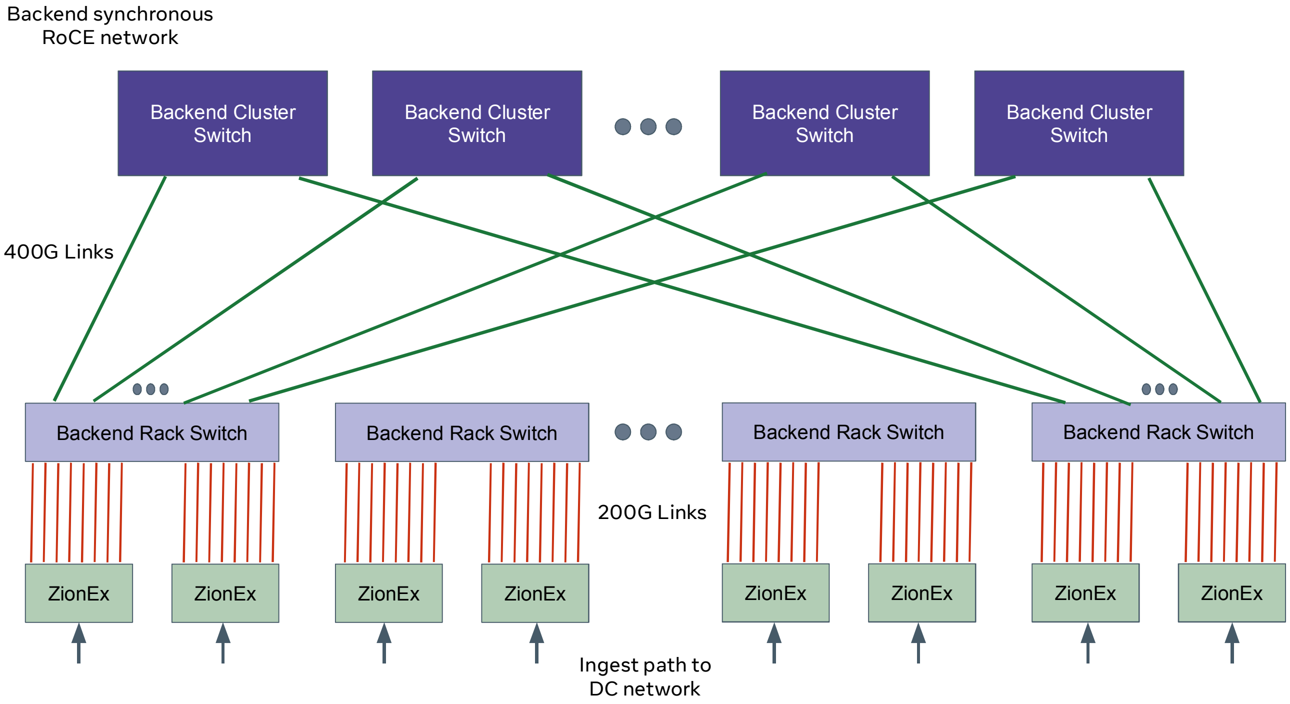
Training Systems Today

Synchronous AI Training Architecture

- Required due to degrading model accuracy when using asynchronous updates across a very large number of workers

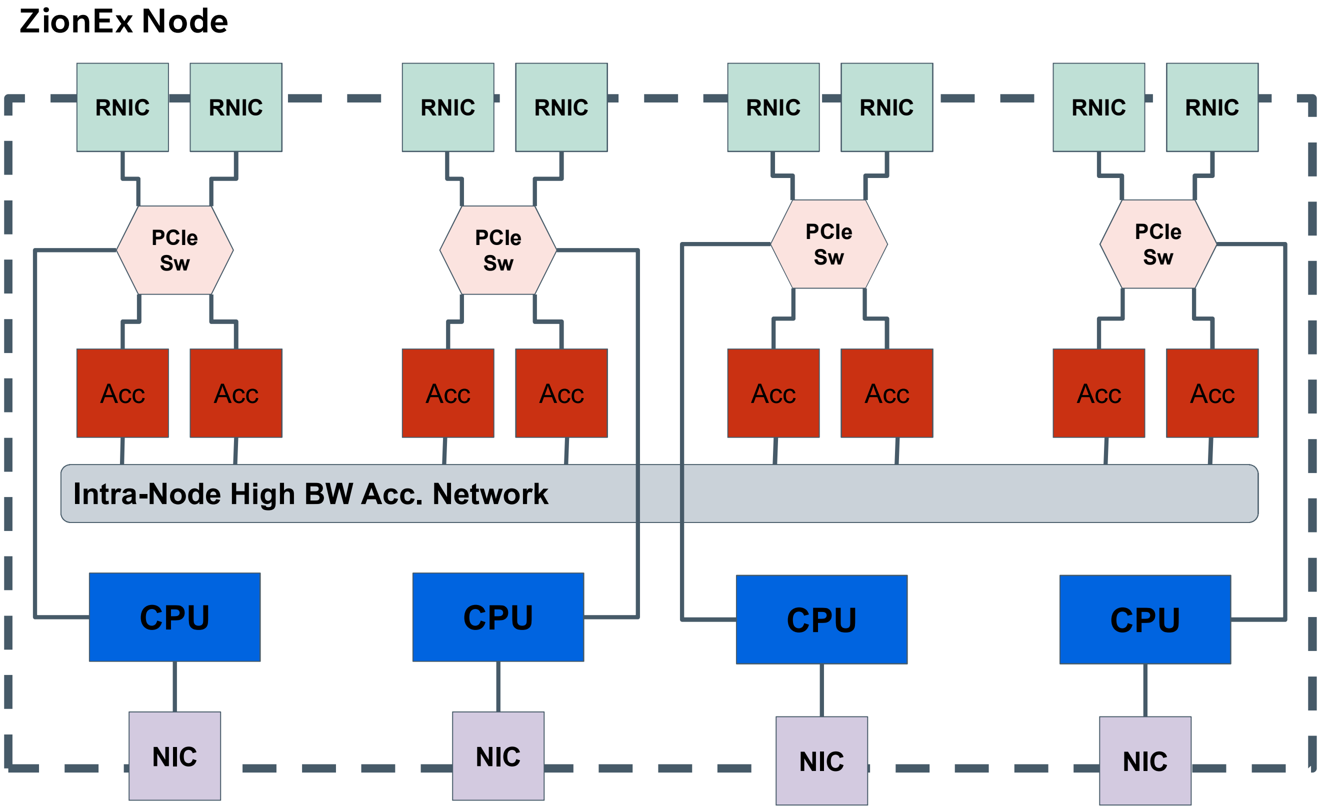
- Utilizes multi-node low latency transport w/ direct data placement (e.g RoCE)
    - Services AlltoAll and AllReduce collectives

- For ease of scaling & management we have a separate backend network



The ZionEx Node

- The ZionEx node ensures ease of data placement by having the backend NICs & accelerators on the same root complex
- Accelerators have a High BW intra-node network
  - Speeds up Allreduce
- CPU is focused on ingest processing and job coordination

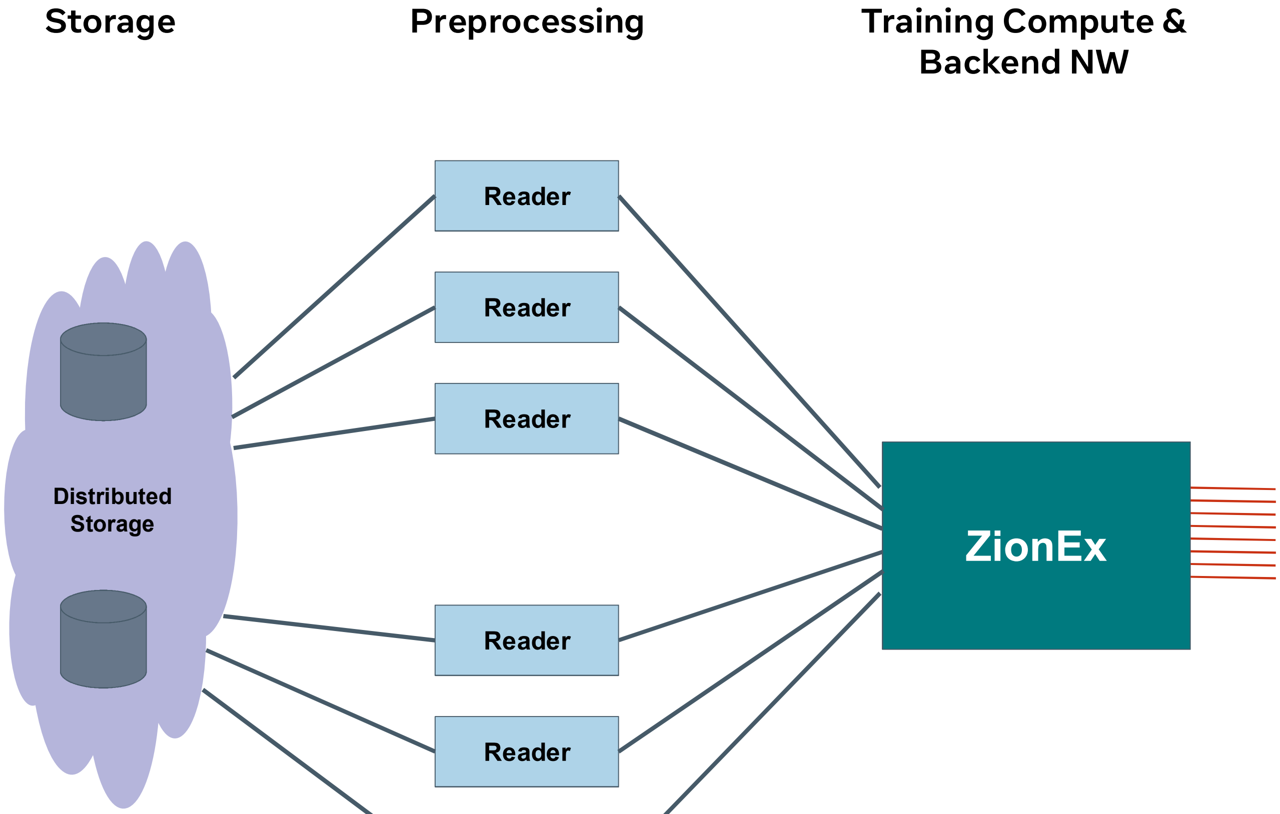


The Ingest Pipeline

- **Storage**: Multi-layer distributed storage contains the model parameters used for training

- **Preprocessing**: The preprocessing stack performs light-weight data pre-processing operations in a distributed fashion

- **Training Compute:** The training compute consists of ZionEx nodes synchronously linked via the backend network



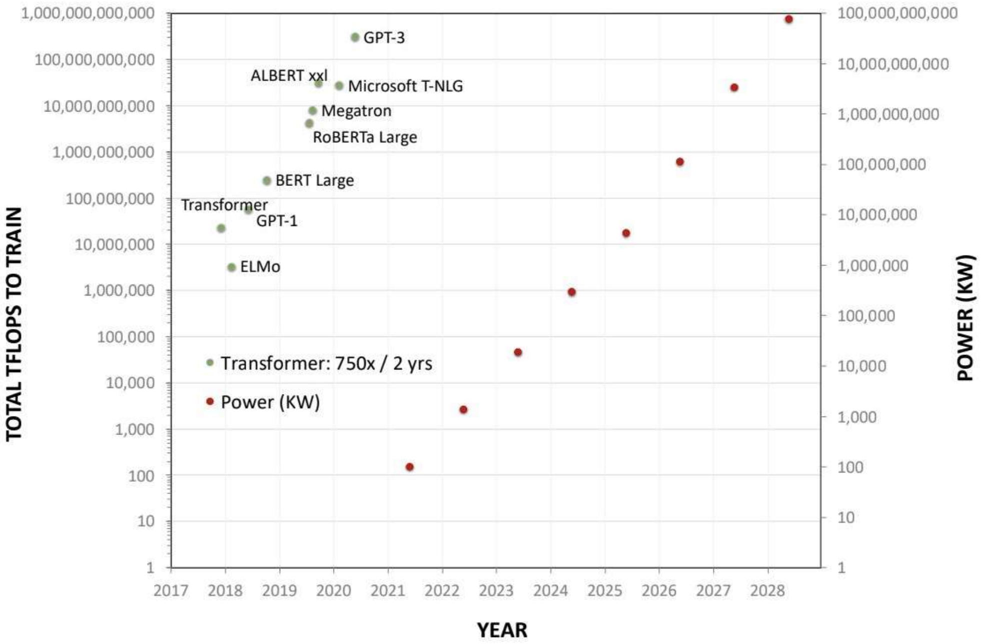
Upcoming Challenges

Current Power Trajectory is not sustainable

- While we expect model optimization and other factors to flatten the curve, HW optimizations will also be needed

- This optimization phase will require tight optimization of infrastructure **while** maintaining flexibility to accommodate changes in workload pattern

- To enable this will require us to leverage HW/SW co-design



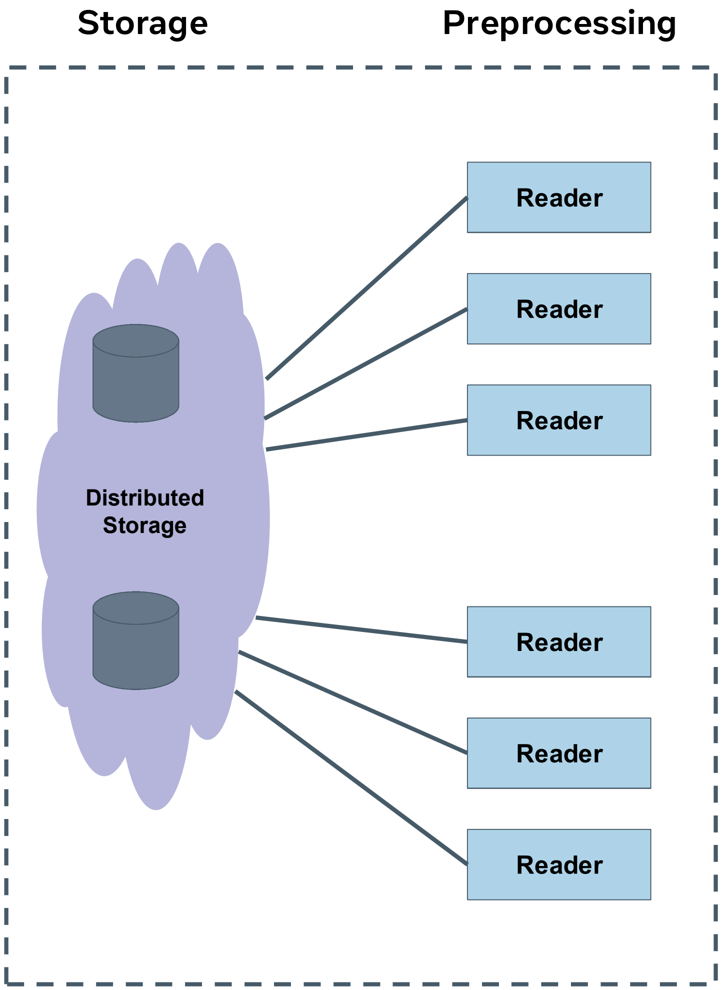
Opportunities for HW/SW Co-design

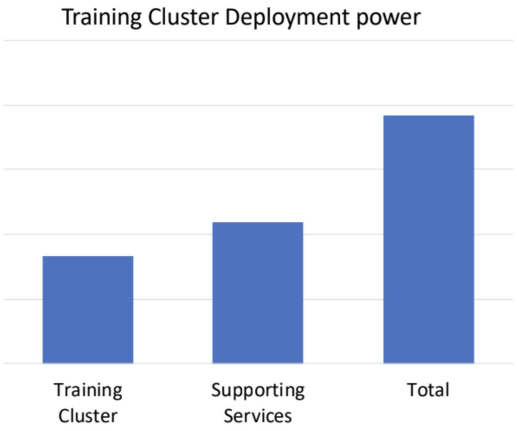
The Ingest Pipeline

- Supporting services are projected to in some cases take more than half of the total cluster deployment power in the near future

- By optimizing the storage architecture to leverage new technology such as network attached storage we can push this trend down

- Another key area to focus on is the offload of data preprocessing

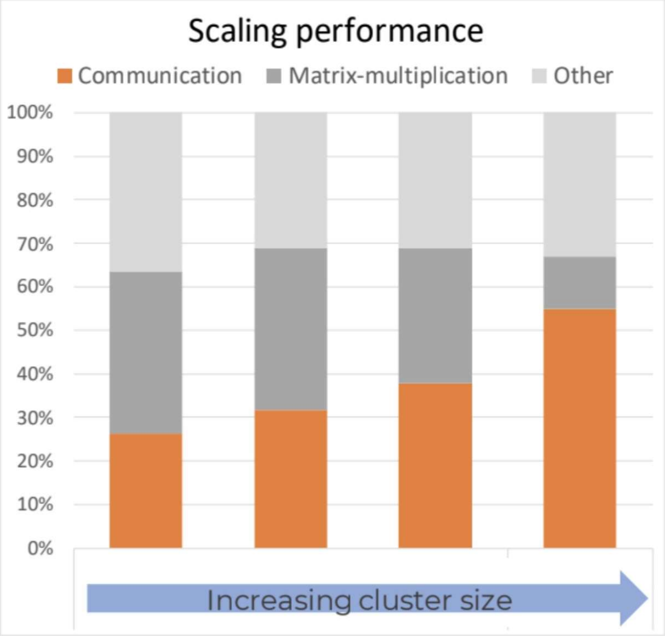


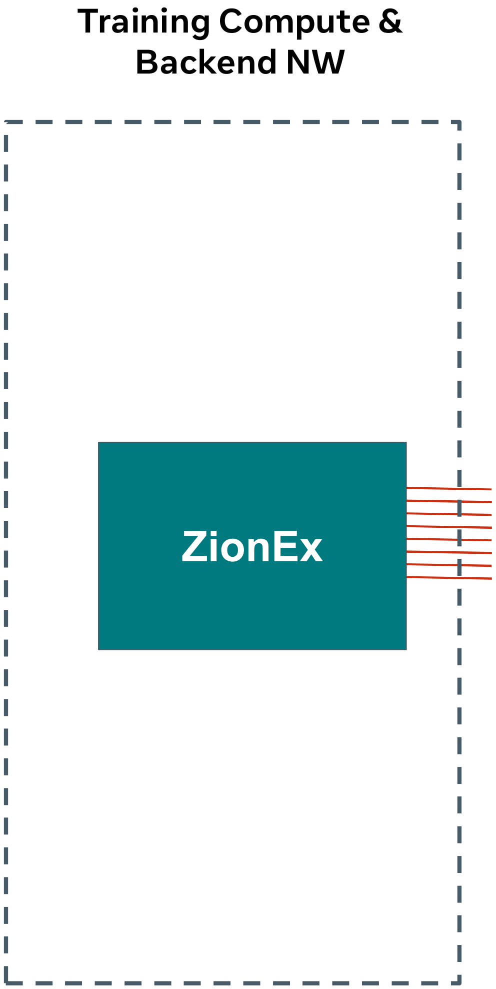


Backend NW Communication

- As cluster size increases, more and more of the overall scaling performance is driven by communication
- This means that focusing on:
  - low latency methods of loss-resiliency or losslessness
  - link utilization/entropy
  - Direct data placement

can provide long term benefits





OPI & The Stack

OPI & The Stack

By ensuring ease of integration w/ existing OSS, adoption will be eased

- **Toolchain:** Integrating with existing compilation & debug frameworks when building programmable devices will make integration with tooling significantly smoother
- **Drivers/kernel/offload integrations:** Tying new open programmable devices to upstream Linux kernel drivers and offload integrations (e.g BPF, NVMeoTCP) makes the value validation phase of exploring new technology much smoother
- **Monitoring:** Visibility into programable devices will be key for their long term success, providing clear monitoring through standardized counters where appropriate and tracepoints will add clear value

Summary

01

We are focused on Synchronous Training

Synchronous training has been shown to scale to recommendation models with trillions of parameters. This requires low latency transport with direct data placement to be effective.

02

HW/SW Co-design is needed to meet our needs

To be able to meet our scaling requirements with the flexibility to meet unpredictable future needs will require the HW/SW co-design of NW endpoint systems. This will be needed to meet needs across storage, preprocessing & the backend network.

03

Essential to integrate new HW w/ standard OSS tools

Ensuring that Open programmable HW is integrated w/ standard OSS tools will not only significantly decrease the friction to adoption, but also ensure that it is easy for developers to innovate and build upon standard use cases.

Meta