Inside the Cerebras Wafer-Scale Cluster

Cerebras Systems

Sean Lie
Co-founder & Chief Hardware Architect

Cerebras Systems

Building a new class of computer system for the future of AI work

A full AI acceleration solution: chip, system, software, ML

Founded in 2016

350+ Engineers

Offices
Silicon Valley | San Diego | Toronto | Tokyo

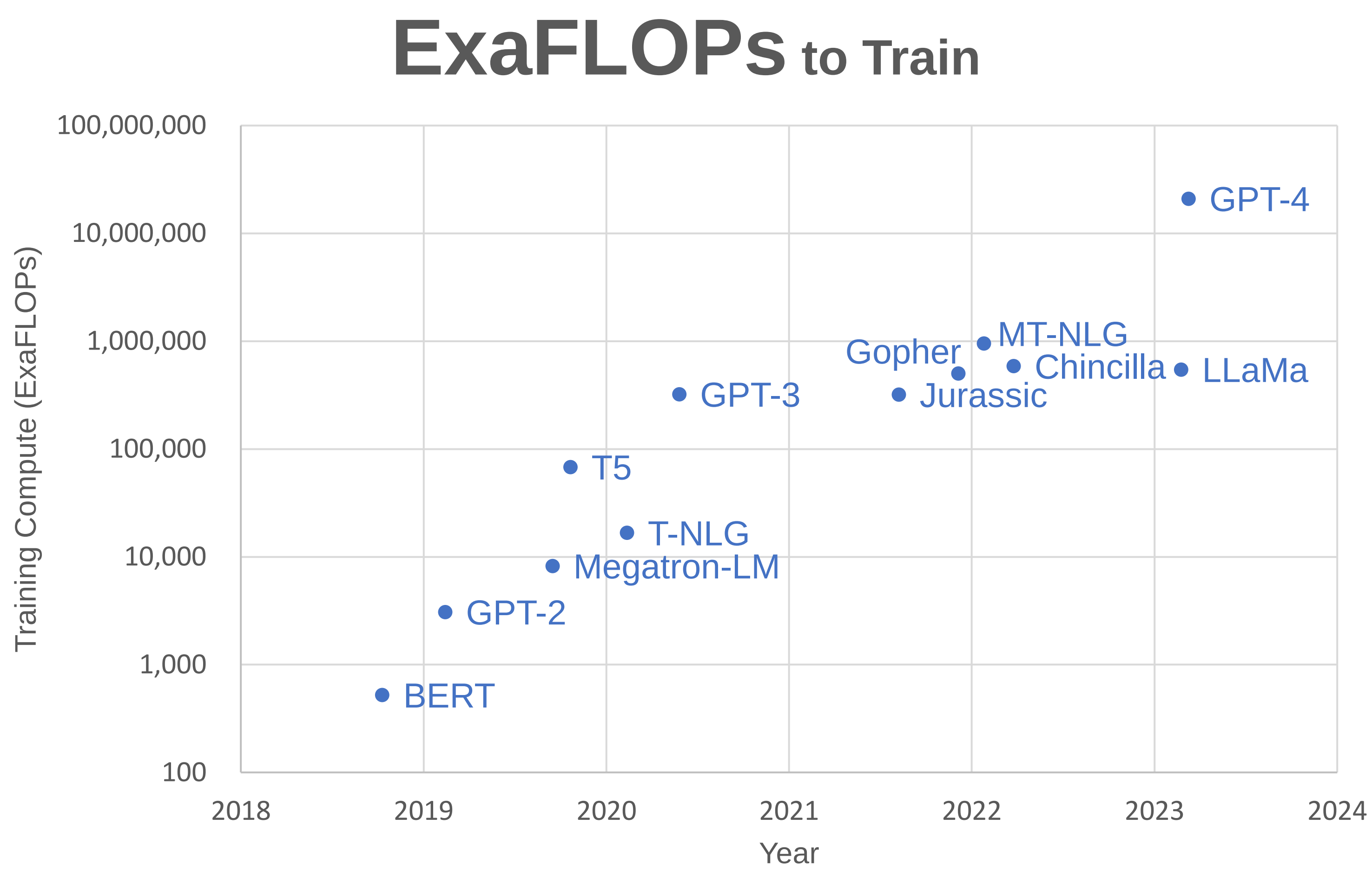
Customers
North America | Asia | Europe

Select Cerebras Customers

Customers: Large Enterprise, HPC, Government

- G42, GlaxoSmithKline, TotalEnergies, AstraZeneca, Bayer, Genentech, Tokyo Electron Devices...
- ANL, LLNL, NETL, PSC, NCSA, EPPC, Leibniz Supercomputing Centre...
- Security, e.g. DARPA, USAF, ARL

Solving the Exponential Growth of Generative AI



Unprecedented demand enabled by HW

40,000x more compute

In just **5 years**

Next 5 years…

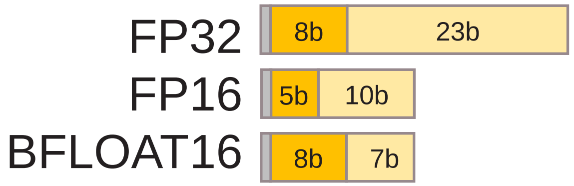
both exciting and daunting

ML Acceleration Hardware Improvements in Industry

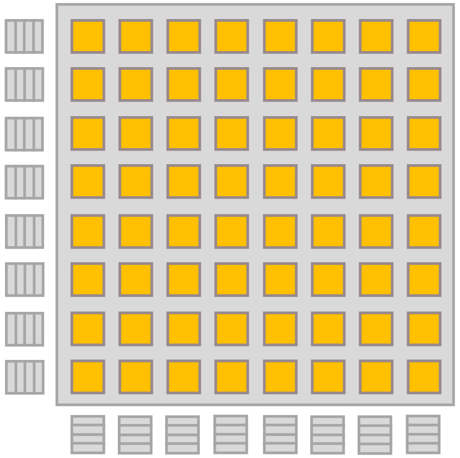
A lot of innovation in last few years

For example…

- Process: 16nm, 12nm, 7nm, 5nm
- Architecture: Low precision, systolic array, etc.



Low precision numerics

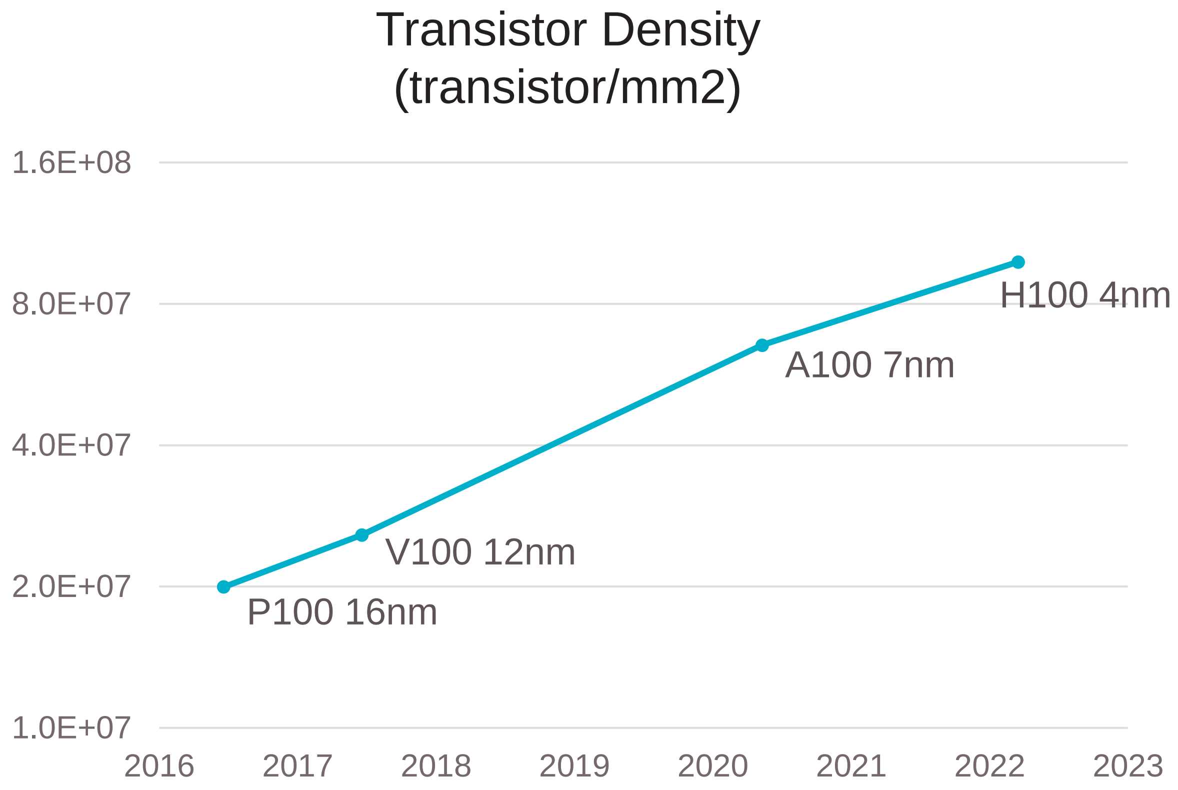


Dense GEMM datapath

Process Technology Gains

For example…

- Process: 16nm, 12nm, 7nm, 5nm
- Architecture: Low precision, systolic array, etc.

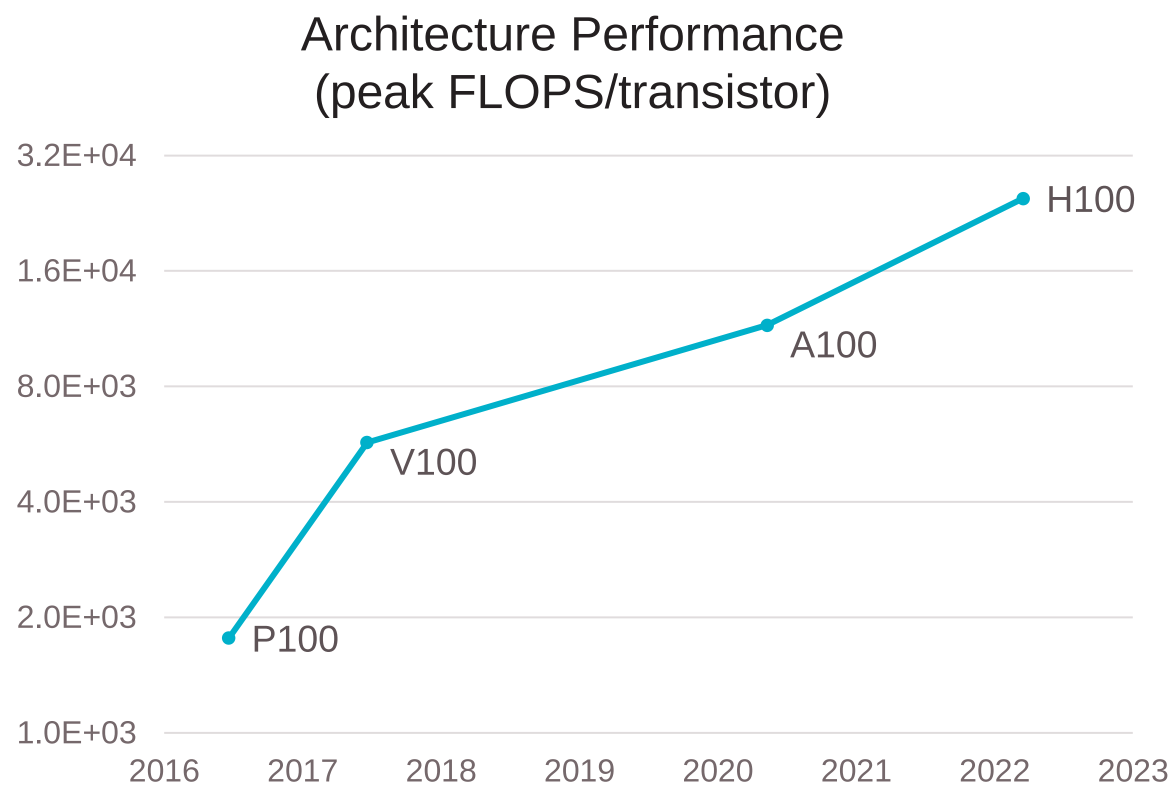


Moore's Law is not dead!

Chip Architecture Gains

For example…

- Process: 16nm, 12nm, 7nm, 5nm
- Architecture: Low precision, systolic array, etc.



Hardware architecture matters!

Meeting the Need

Exciting: in the last 5 years…

What our industry delivered:

- 5x: Process technology
- 14x: Chip architecture
- **600x: Cluster scale-out**

Daunting: what the ML needs:

- BERT → GPT-4
- **40,000x** more compute

Cluster Scale-out Dominated Performance Gains

But Existing Scale-out is Limited

Massive models need massive **memory**, massive **compute**, and massive **communication**.

On giant clusters of thousands of small devices, **all three become intertwined, distributed problems**.

Running a **single problem** requires inefficient, fine-grained partitioning and coordination of memory, compute, and communication

Distribution complexity scales dramatically with cluster size

The Cerebras Approach

Balanced scaling across all dimensions

- Process: **WSE-2 Wafer Scale Integration**
  - Order of magnitude improvement
  - Amplifying Moore's law
  - 46,225 mm2
  - 2.6 trillion transistors
  - 850,000 cores
- Architecture: **Unstructured sparsity acceleration**
  - Order of magnitude improvement
  - Full memory bandwidth for vector-scalar ops
  - Fine-grained dataflow scheduling
- Scale-out: **Wafer Scale Cluster architecture**
  - Inherently scalable to train largest models

cerebras

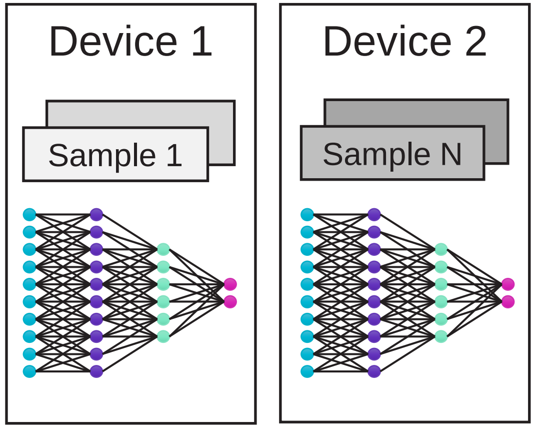
Wafer Scale Cluster

Cluster Level Co-design

Challenges to Existing Scale-out

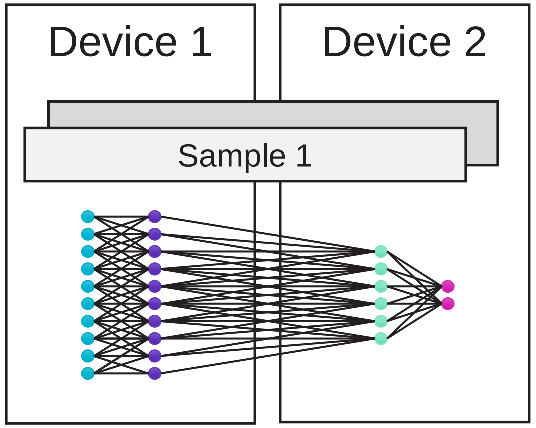
Several existing scale-out techniques

Data Parallel



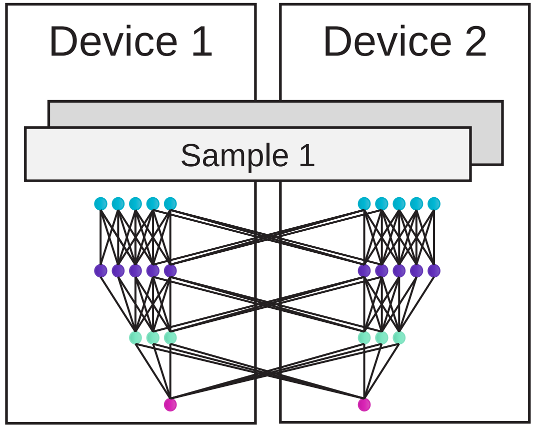
Simple and scales well
Multiple samples at a time
Parameter memory limits

Pipelined Model Parallel



Multiple layers at a time
Communication overhead
N^2 activation memory

Tensor Model Parallel



Multiple splits at a time
Communication overhead
Complex partitioning

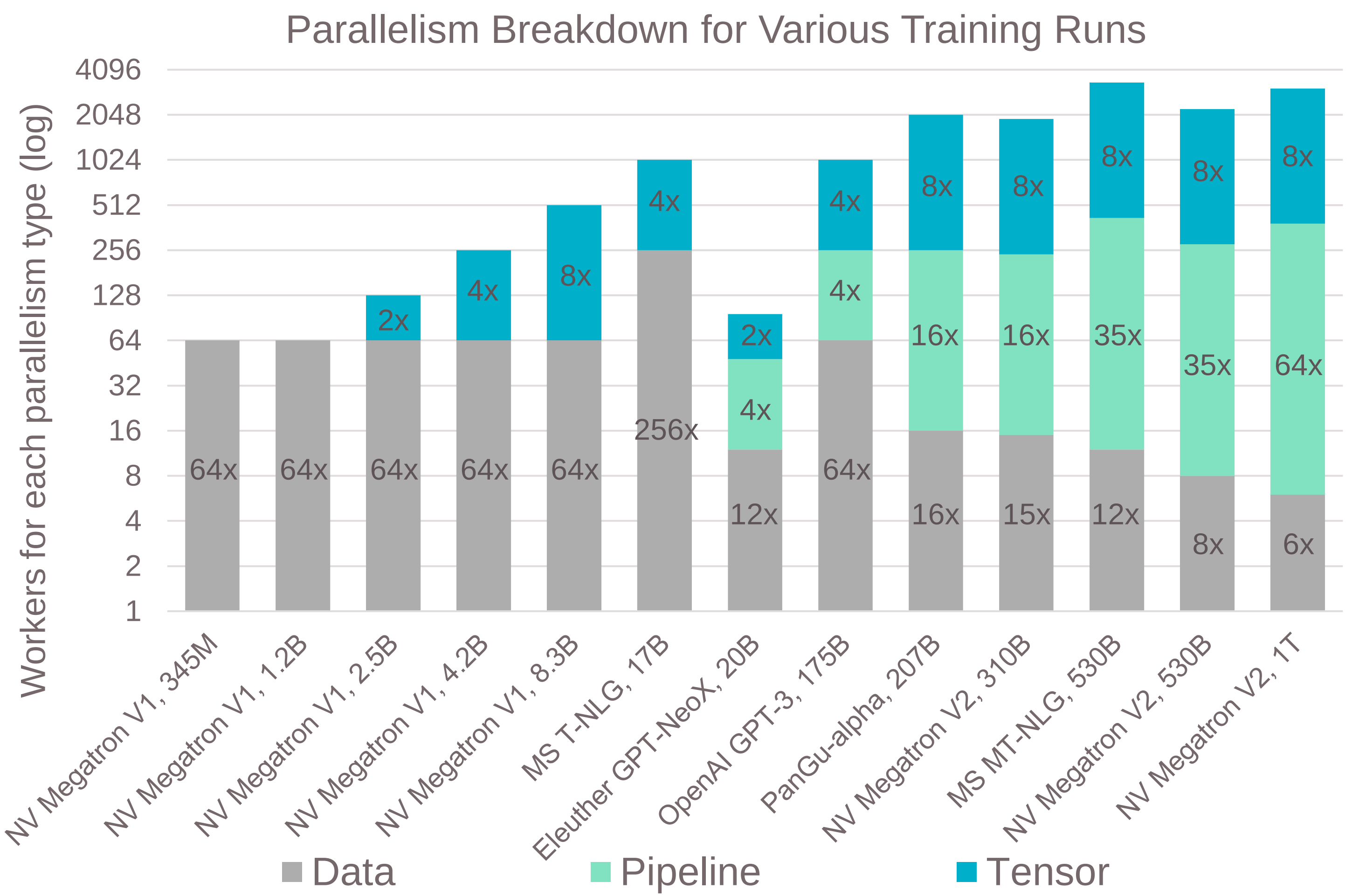
No single solution, traditional scale-out requires hybrid use of all forms of parallelism

Complexity in Practice on GPU Clusters

Traditional scaling complexity

- Extreme-scale models on GPU requires all forms of parallelism simultaneously
- Tensor model parallel limited to within single server
- Pipelined model parallel makes up most of parallelism for largest model, but it's the most complex
- Solution is bespoke distributed system
- Resulting in complexity and often poor scaling

All share same limitation: memory tied to compute

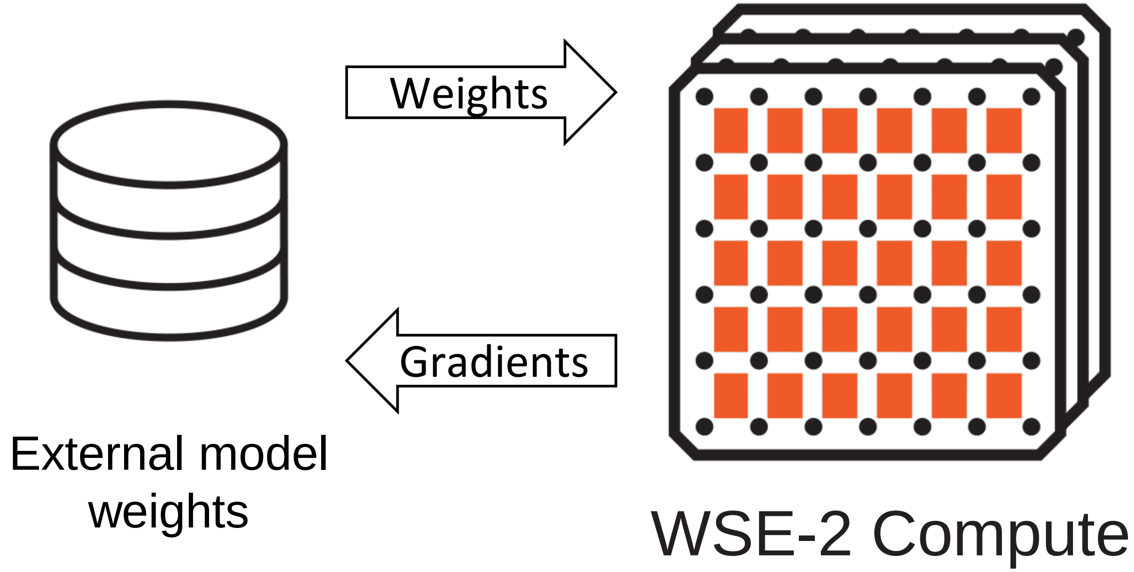


*Parallelism breakdown derived from the model publication:
NV Megatron V1, MS T-NLG, Eleuther GPT-NeoX, OpenAI GPT-3, PanGu-alpha, NV Megatron V2, MS MT-NLG

The Cluster *is* the ML Accelerator

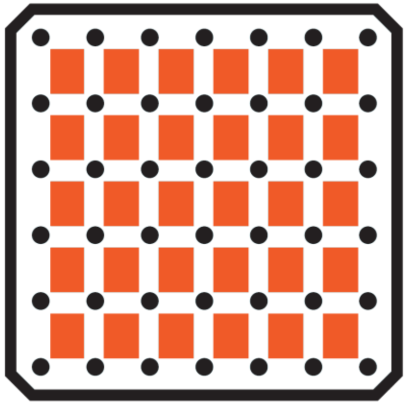
Disaggregation of memory and compute

- Architect cluster-level memory and compute
- Store model weights externally
- Weight Streaming execution model
- Untangle memory and compute dependency



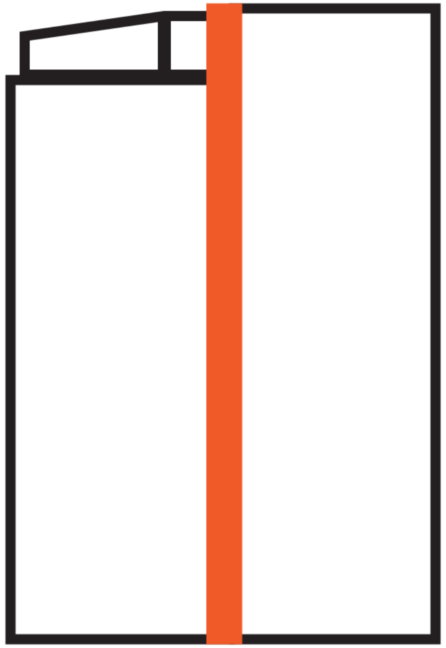
Scale model size and training speed independently

WSE-2

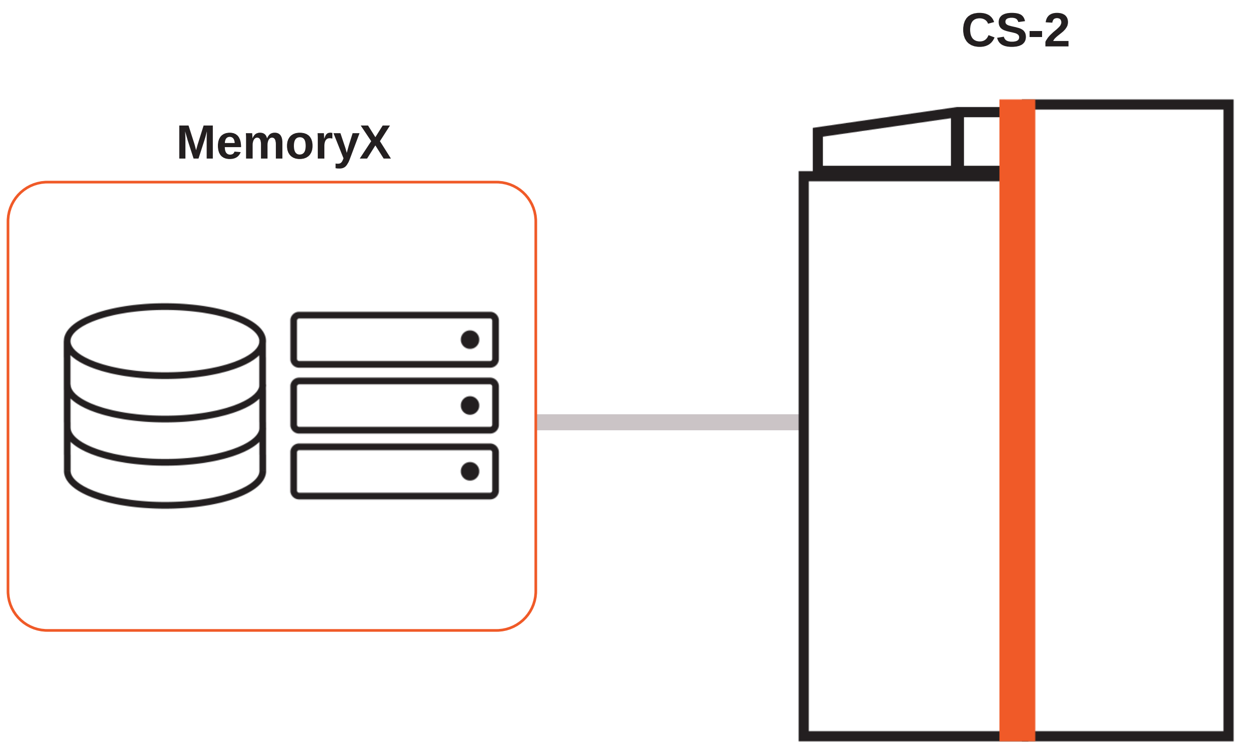


850,000 cores can run models of all sizes

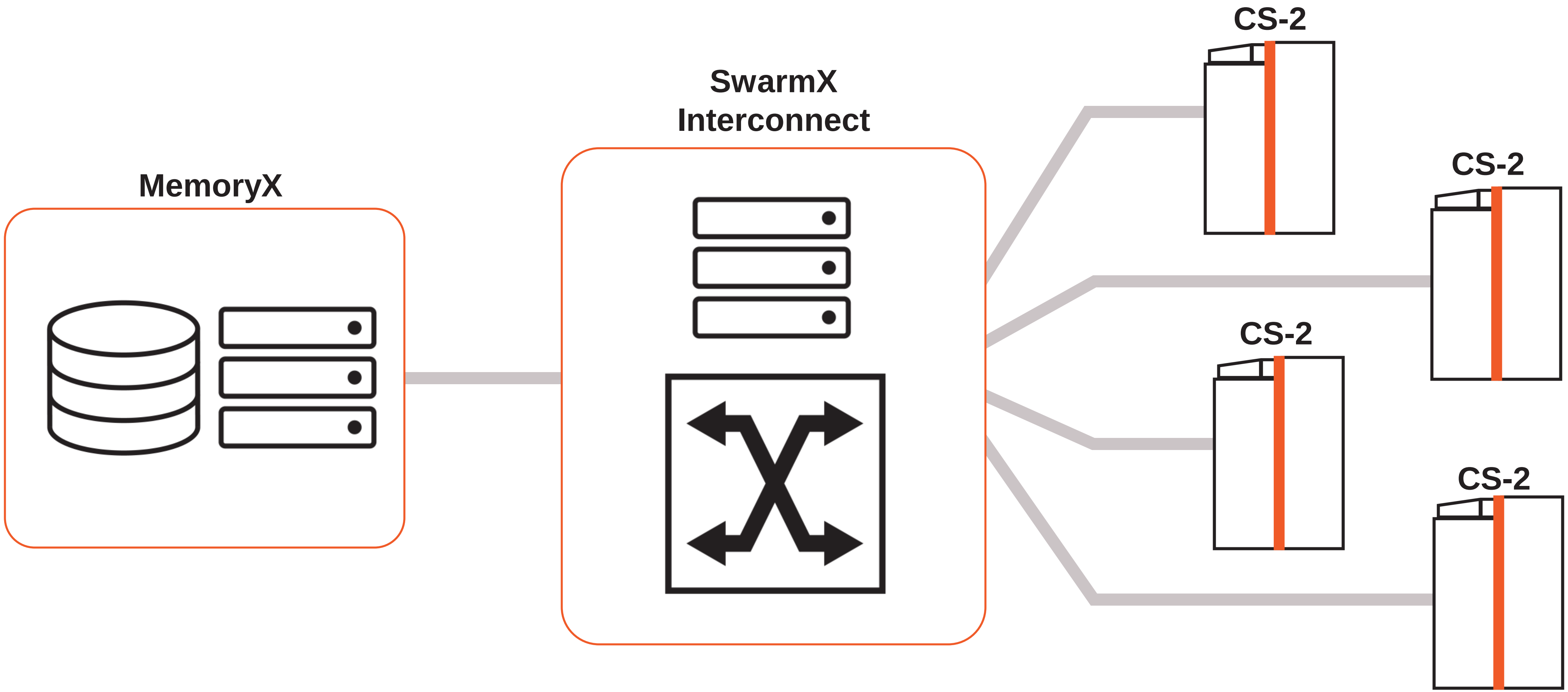
CS-2



850,000 cores can run models of all sizes



Up to 120 trillion parameters on a single CS-2



Near-linear performance scaling up to 192 CS-2s

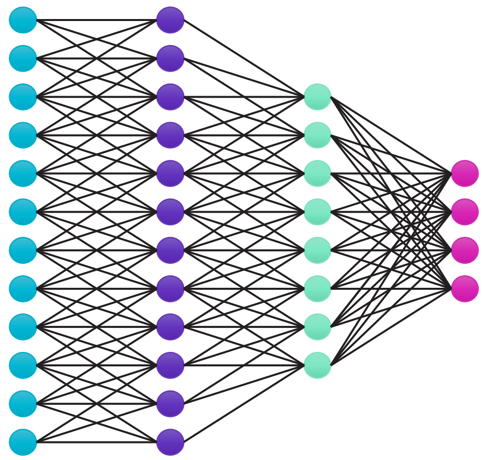
Weight Streaming Execution Model

Built for extreme-scale neural networks:

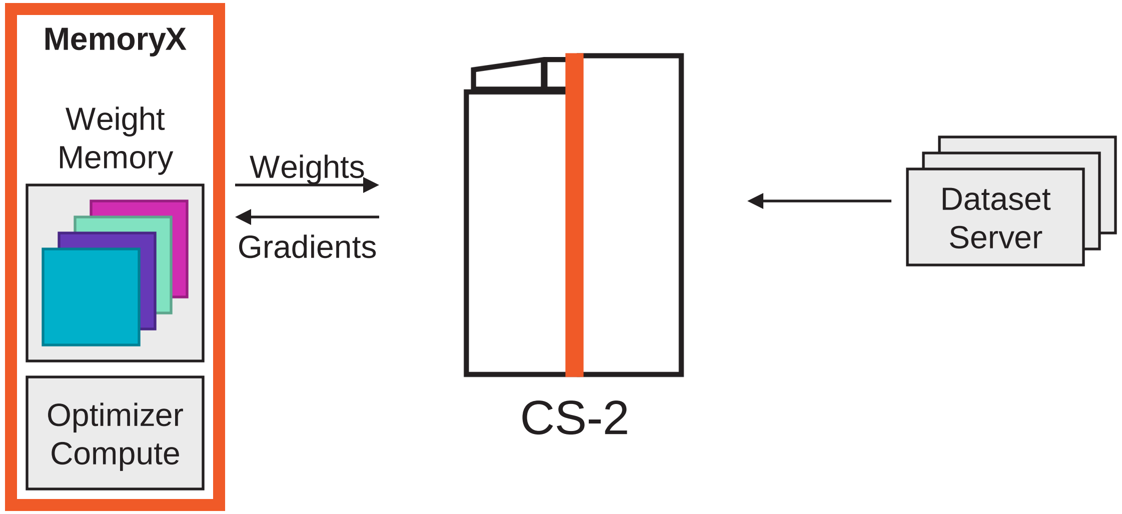
- Weights stored externally off-wafer
- Weights streamed onto wafer to compute layer
- Weight never stored on wafer
- Activations only are resident on wafer

Decoupling weight optimizer compute

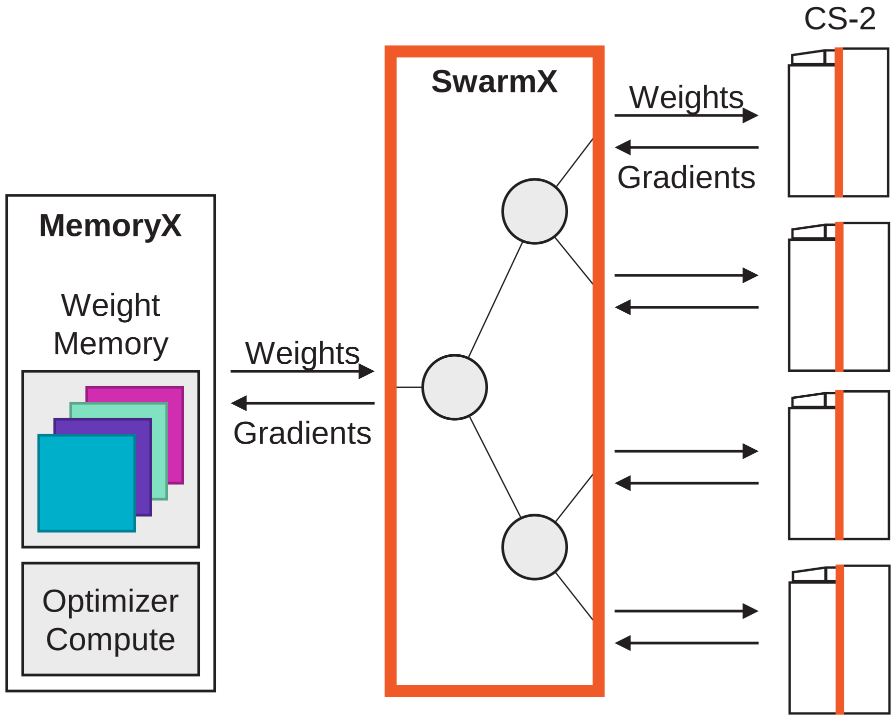
- Gradients streamed out of wafer
- Weight update occurs in MemoryX



Memory and compute hierarchy capable of massive models on single device



SwarmX Fabric Connects Multiple CS-2s



- Data parallel training across CS-2s
- Weights are **broadcast** to all CS-2s
- Gradients are **reduced** on way back
- **Multi-system scaling with the same execution model as single system**
  - Same system architecture
  - Same network execution flow
  - Same software user interface

Scalable to extreme model sizes

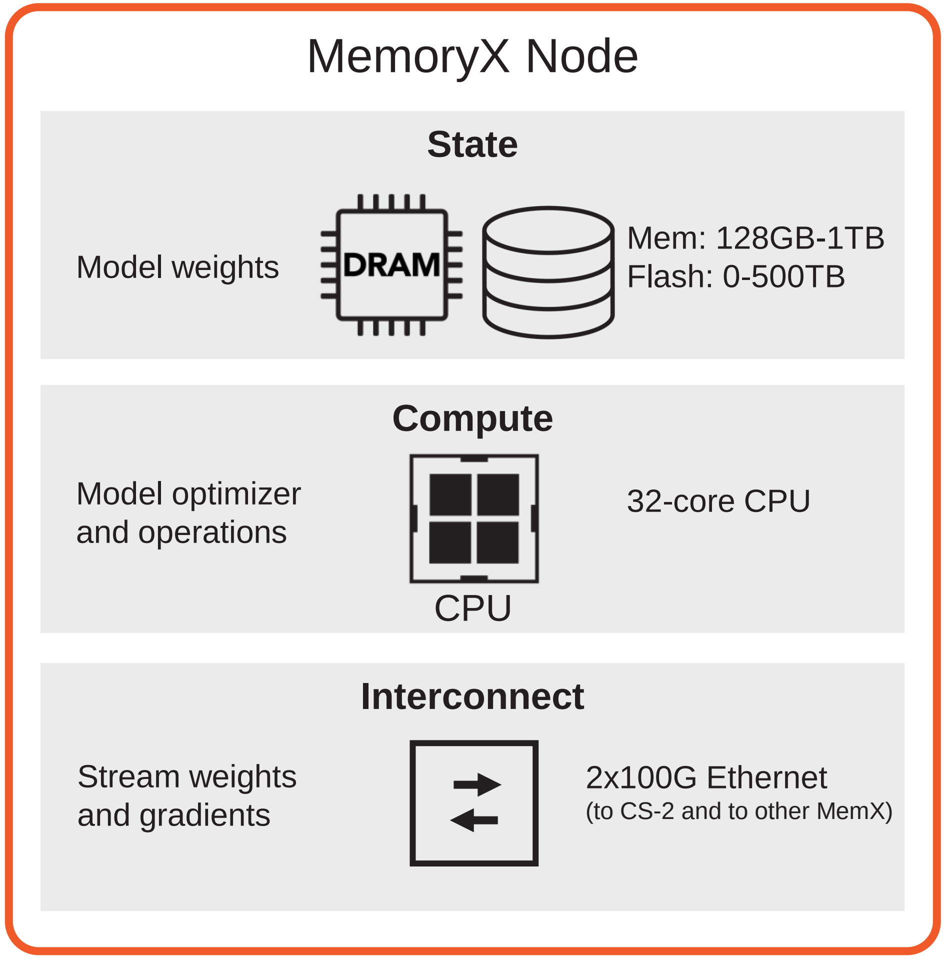
Compute scaling independent from capacity

Cluster Design

MemoryX: Flexible Capacity and Compute

Mapping model weights to hardware

- State
  - Weights stored in DRAM and flash
  - Cost effective and high performance
- Compute
  - Optimizer and other ops run on CPUs
  - General purpose and flexible
  - Support for all common ML operations
- Interconnect
  - Stream weights/gradients over 100G Ethernet
  - Dedicated interfaces to CS-2 and other MemX
- Parallel Operation
  - Tensors sharded to use distributed capacity
  - Multiple nodes for high throughput



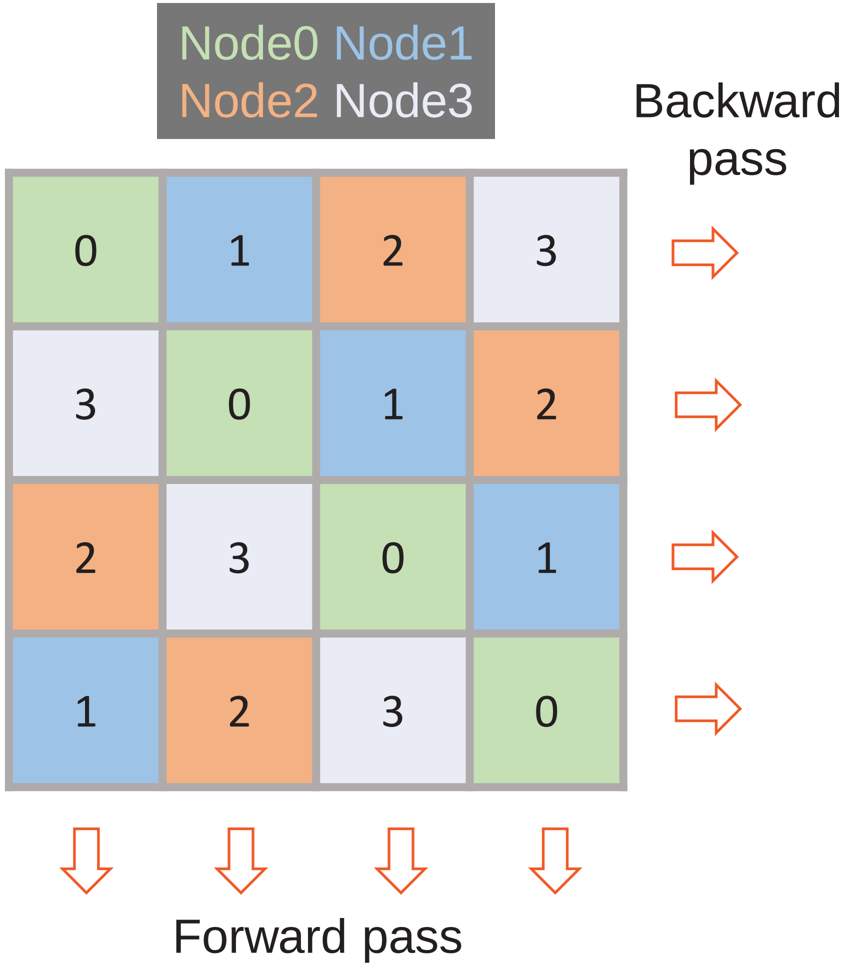
1xCS-2	64xCS-2
12 nodes	Up to 768 nodes
12TB 6PB	**768TB 384PB**
384 cores	24k cores
2.4Tb/s	153Tb/s

MemoryX: Efficient Weight Sharding

Distributed state requires special handling for non-elementwise tensor operations

- Native support for zero communication transpose operation
  - Common weight transpose used in every backward pass of training
  - Data sharded in “checkboard” pattern to enable parallel transpose operation
  - In the forward pass, each row streamed in parallel across nodes into CS-1
  - In the backward pass, each column streamed in parallel across nodes into CS-1
- Support for full collective communication ops
  - Rare but required for some ML operations such as gradient clipping

“Checkerboard” Sharding Pattern



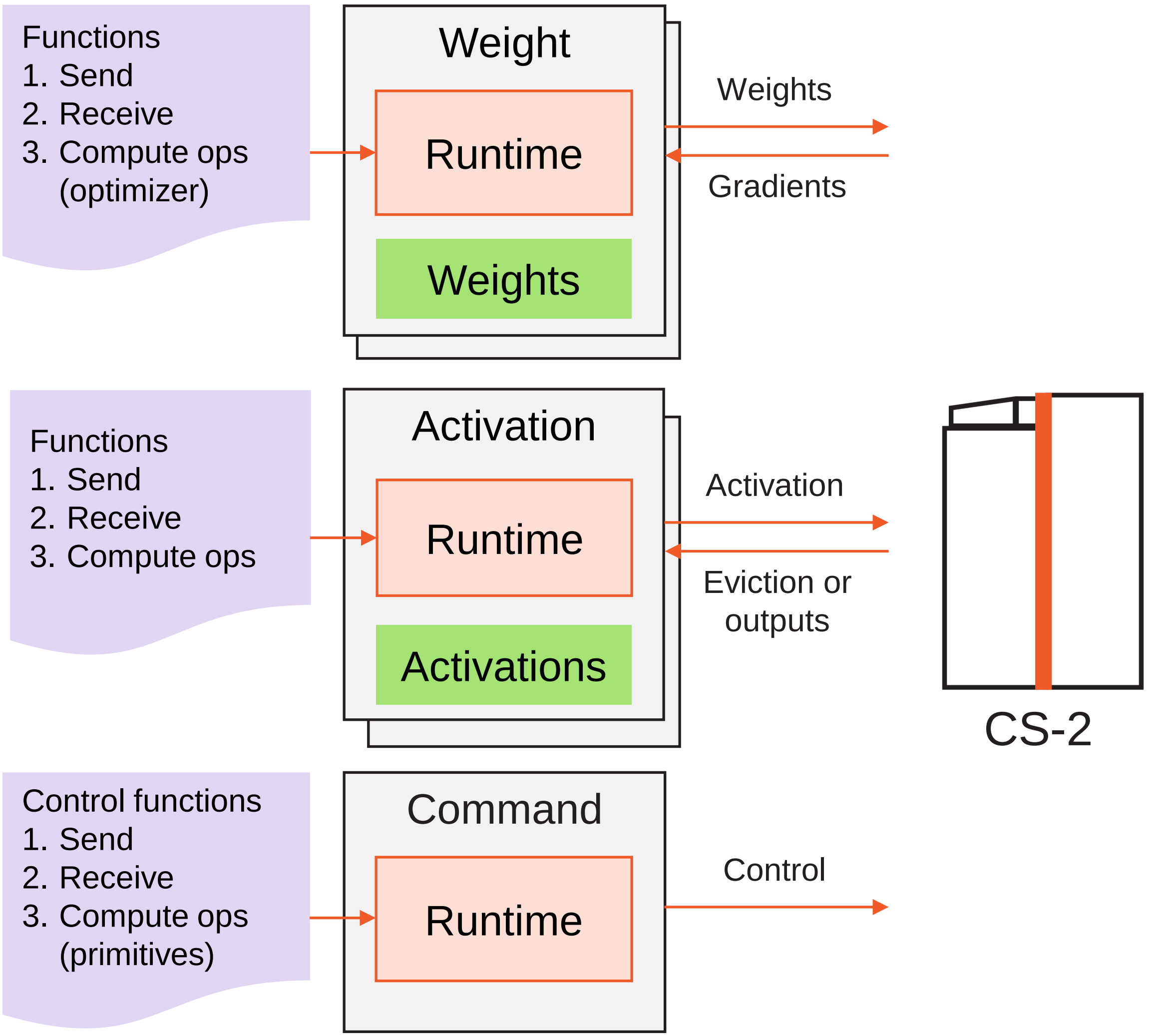
MemoryX: High Performance Runtime

Highly tuned runtime

- Data transfer: send/receive weights/gradients
- Compute: model ops not run on CS-2

Independent runtime functions on MemoryX

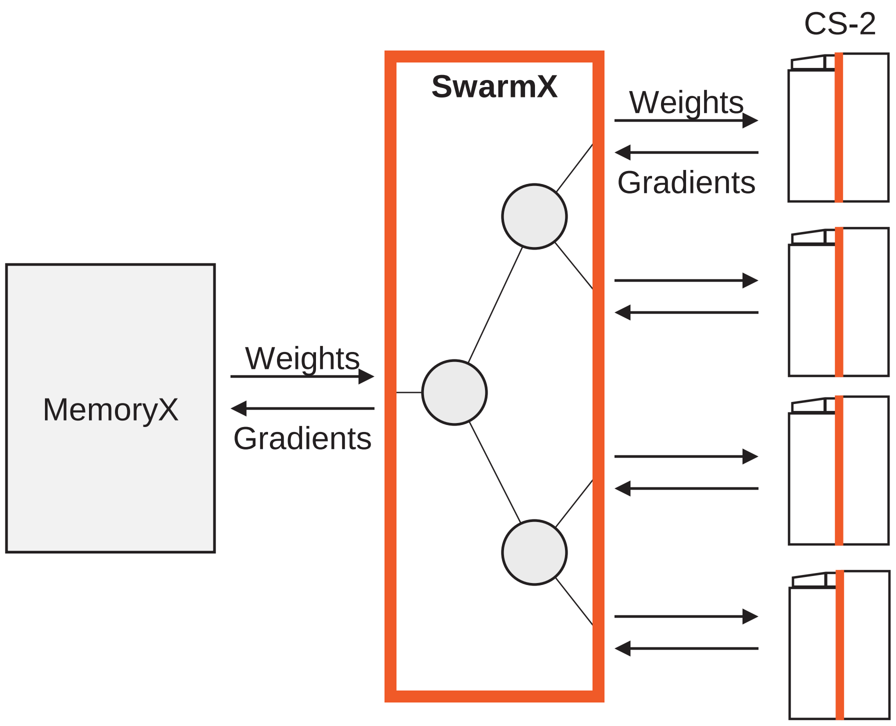
1. Weight runtime
   - Stream weights/gradients to/from CS-2
   - Perform compute ops on weights
     - i.e. optimizer and weight update
2. Activation runtime
   - Stream tensors to/from CS-2
     - i.e. input dataset, activation evict/refill
3. Command runtime
   - Stream control commands to CS-2
     - i.e. instructions to coordinate kernels on CS-2



Cluster Scale-Out

SwarmX fabric to scale out CS-2 cluster

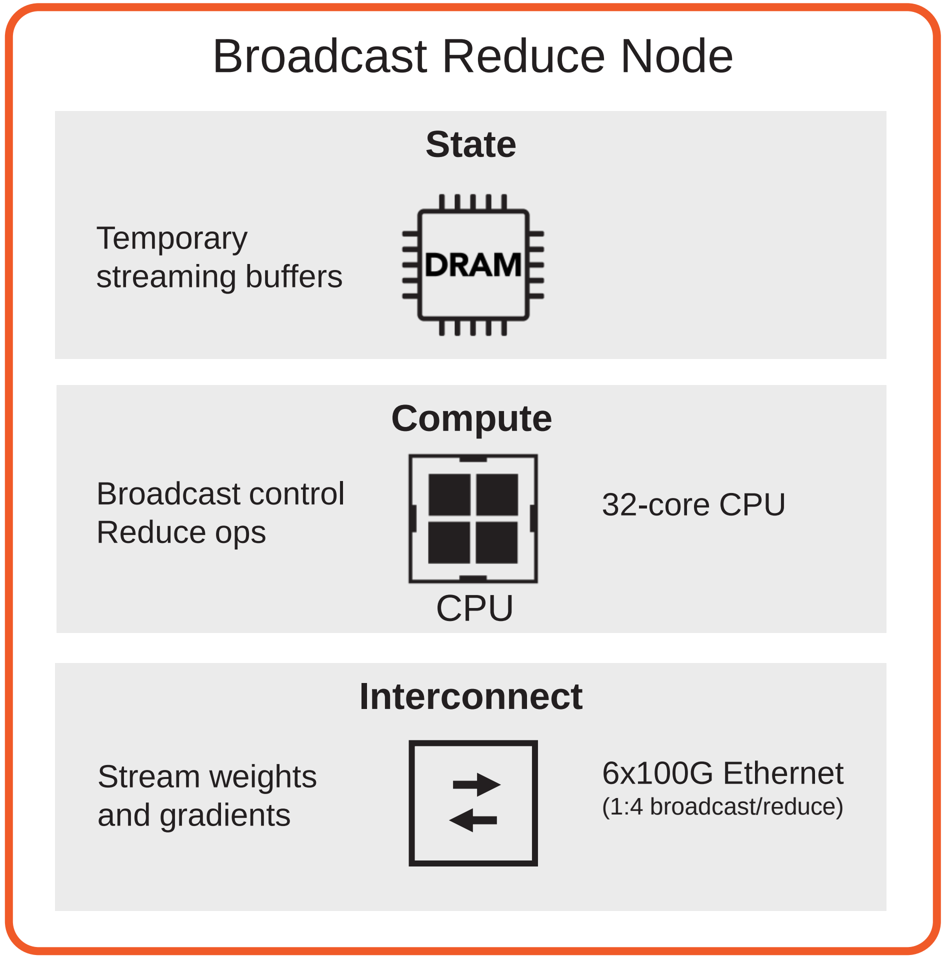
- Connectivity
  - Physical interconnect between all cluster components using high speed 100G Ethernet
  - Cost effective and high performance
  - RoCE RDMA for low overhead and latency
- Broadcast Reduce (BR)
  - Replication and reduction functions performed on flexible CPUs
  - General purpose and high performance
  - Enables efficient data parallel only training



SwarmX: Flexible Broadcast and Reduce Bandwidth

Mapping data parallel training to hardware

- Buffers
  - Temporary streaming buffers in DRAM
  - Low capacity and high performance
- Compute
  - Broadcast and reduction ops run on CPUs
  - General purpose and flexible
- Interconnect
  - Stream over 100G Ethernet RoCE RDMA
  - 6 interfaces for up to 1:4 broadcast/reduce
  - 1 redundant interface
- Parallel Operation
  - Multiple nodes for high throughput

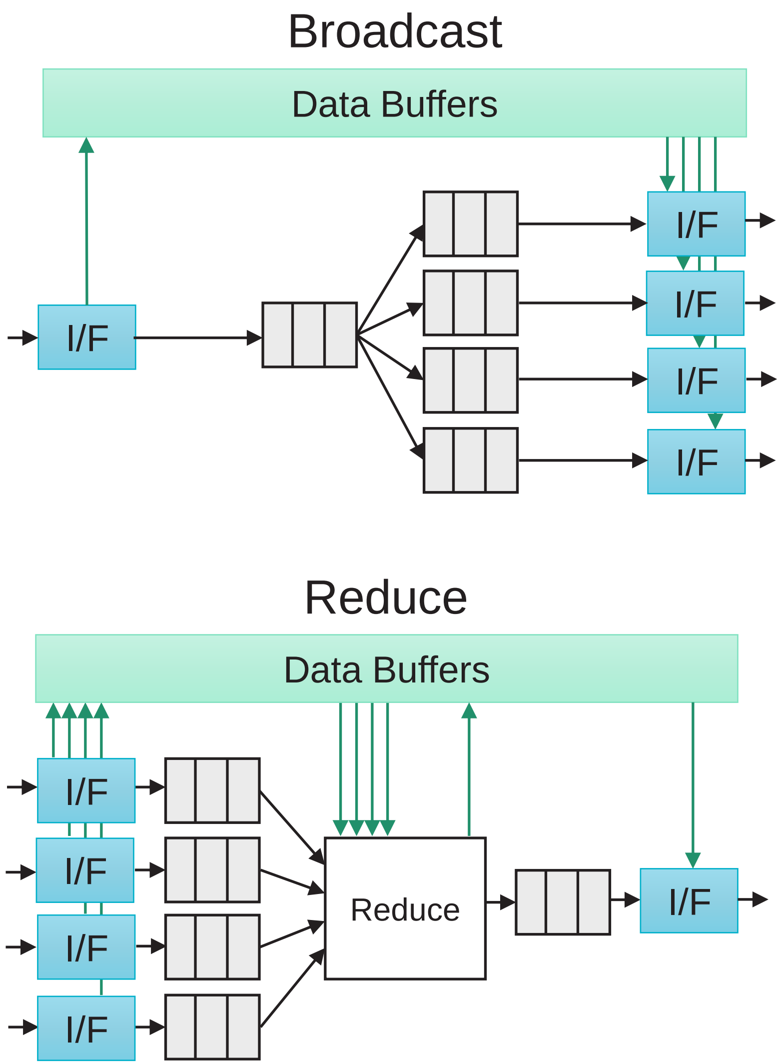


4xCS-2	64xCS-2
12 nodes	252 nodes (3 level tree)
384 cores	8k cores
7.2Tb/s	**151Tb/s**

SwarmX: Efficient Broadcast Reduce

High performance datapath

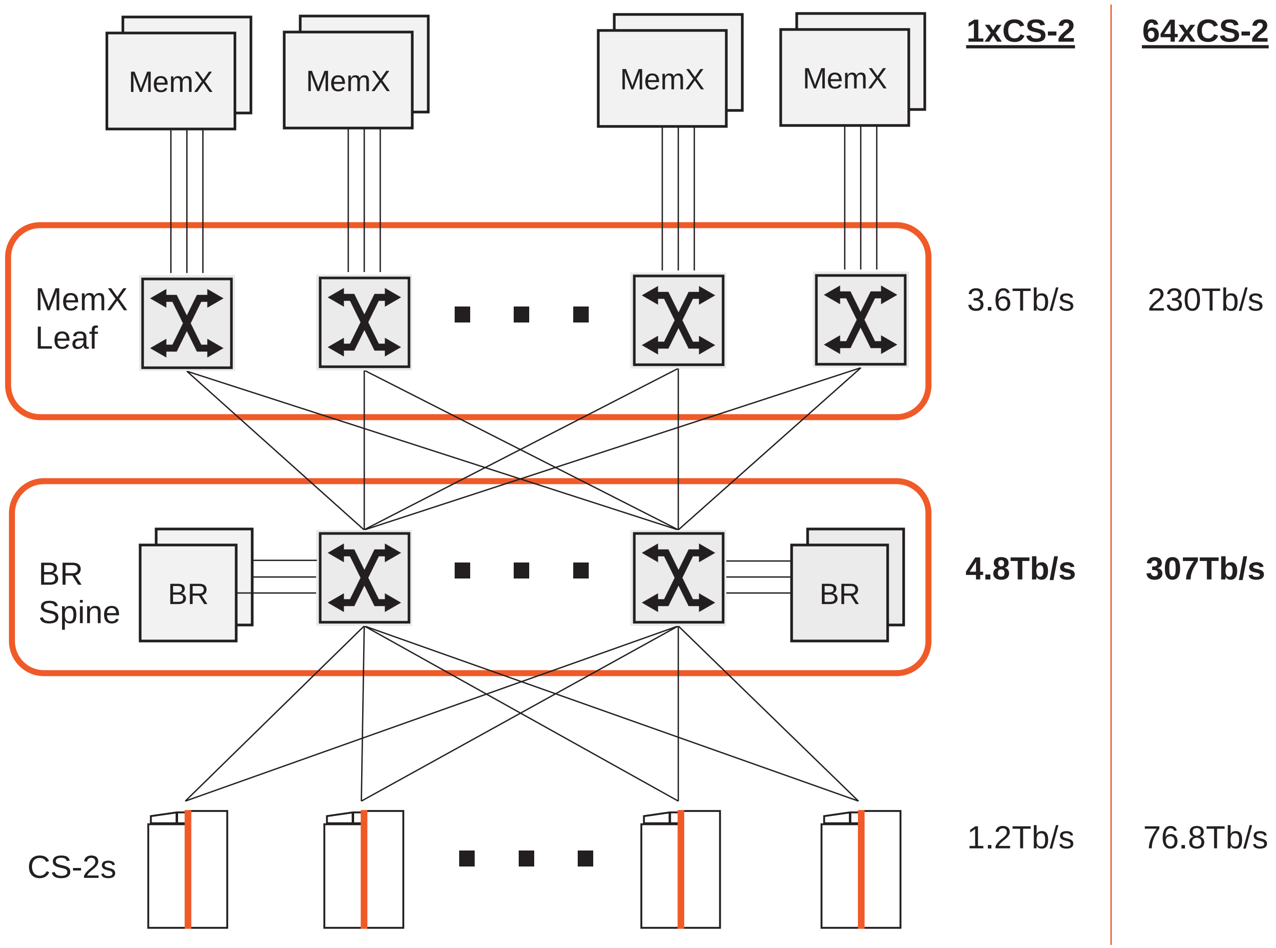
- Each BR node performs up to 1:4 broadcast/reduction
- 1:4 broadcast
  - Input data queued and steamed to output
  - Zero copy with light weight control processing
- 4:1 reduce function
  - Input data queued and aggregated to output
  - Flexible set of reduction operations
    - Sum, Min/Max, Argmin/Argmax
  - Support range of ML usage cases
    - Data parallel gradient accumulation, contrastive loss, tensor summaries, etc.



SwarmX: Scalable and Flexible Topology

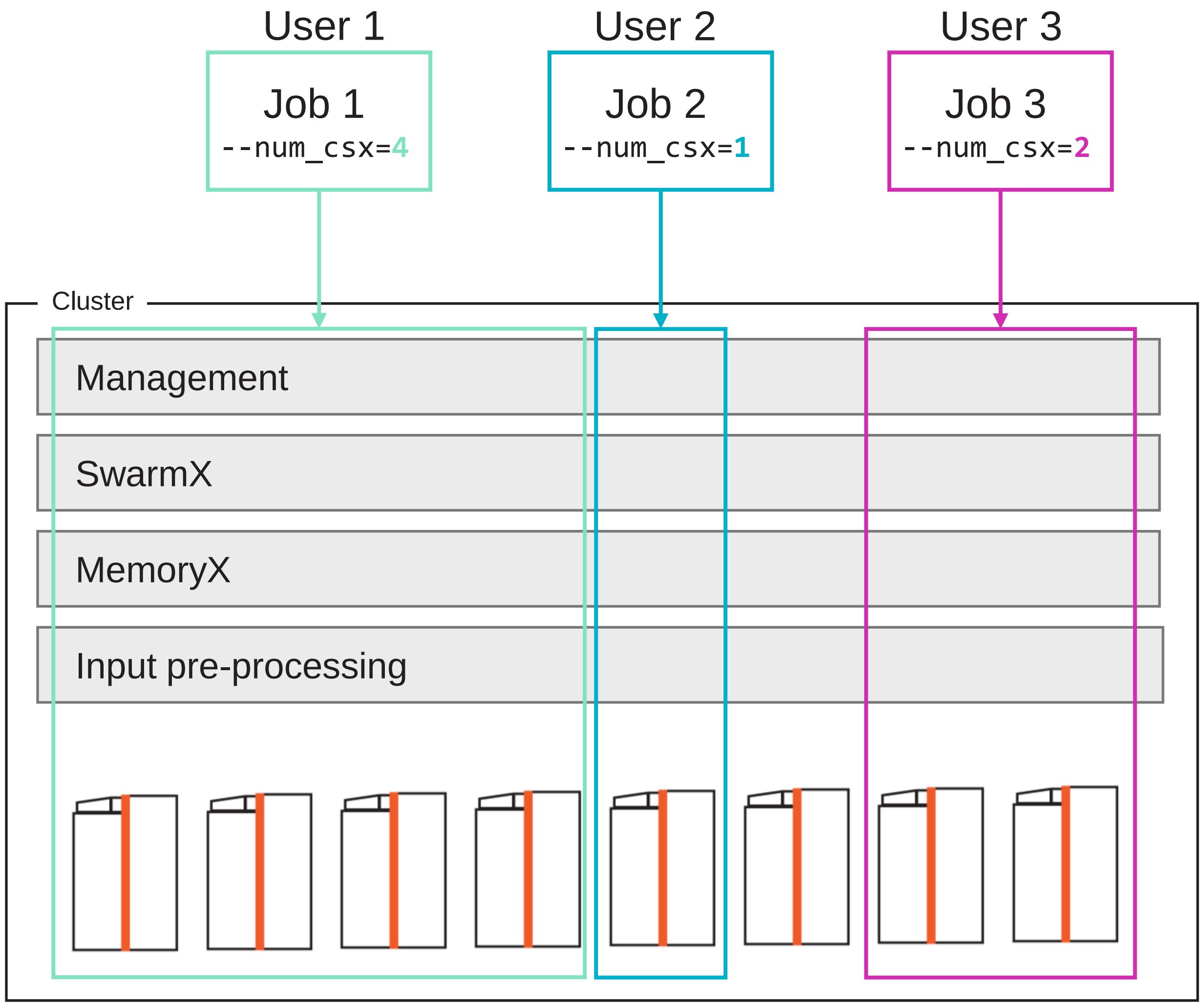
Scalable 2-layer spine-leaf topology

- MemoryX leaf switching
  - Connecting MemoryX globally to BR and CS-2
    - Primary traffic flow for weight streaming
  - Connecting MemoryX nodes locally
    - Secondary traffic flow for collective communication between weight shards
  - Connecting to data processing servers and storage uplink (not shown)
- Broadcast Reduce spine switching
  - Connecting all-to-all globally
    - High aggregate bandwidth and flexible
  - Connecting Broadcast Reduce (BR) nodes
    - Processing in transit between MemX and CS-2
    - Enable logical tree topology flexibly

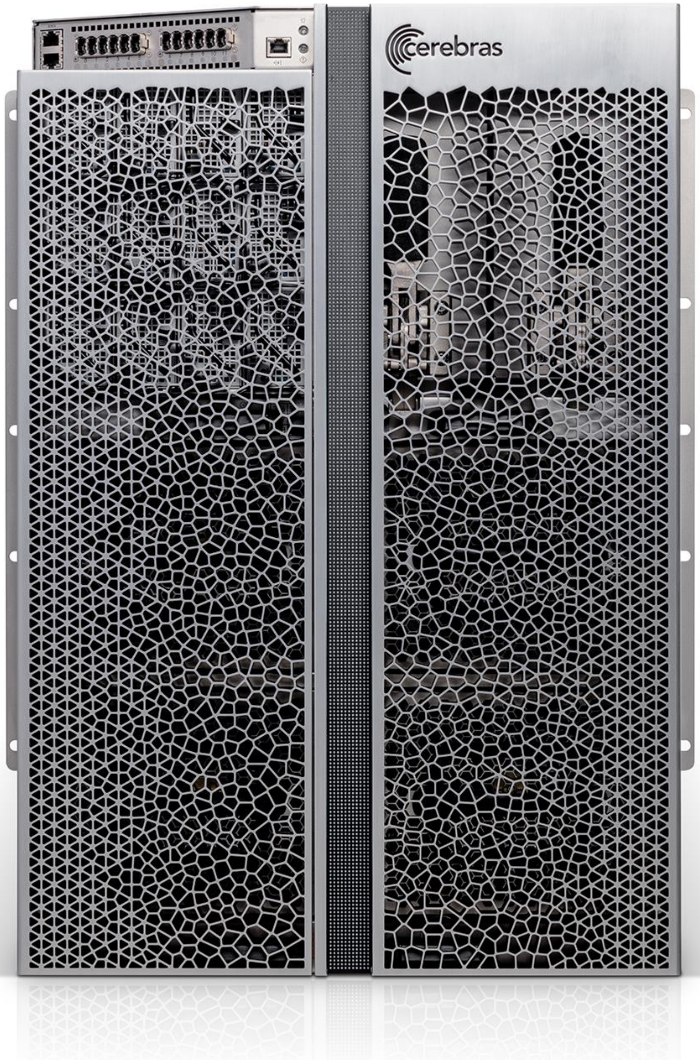


Flexible Resource Provisioning and Management

- Resources configured to meet workload need
  - MemoryX capacity: size of models
  - MemoryX quantity: number of parallel jobs
- Cluster internally manages all resources
- Sub-cluster partitioning
  - Dynamically partitioning to any sub-cluster size
  - e.g. 16x cluster = 8x + 4x + 2x + 1x + 1x
- MemoryX memory allocation
  - Larger models use higher capacity MemoryX
- SwarmX fabric allocation
  - BR node assignment to sub-cluster needs
- Redundancy / fail-in-place
  - No single point of failure
  - Resume operation with alternative resources



Pulling it All Together


cerebras


cerebras
Cerebras
ASA
NVMe


cerebras
cerebras
cerebras


cerebras
CRV-2B
cerebras

Andromeda Wafer Scale Cluster

16

CS-2 Systems

1 ExaFLOPs

sparse compute

13.5M

AI-optimized cores

120 PetaFLOPs

dense compute

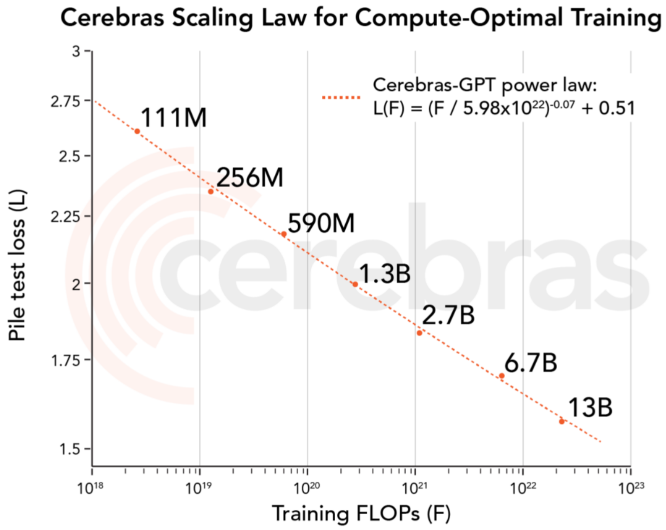
cerebras

Cerebras-GPT: Open Compute-Optimal LLMs

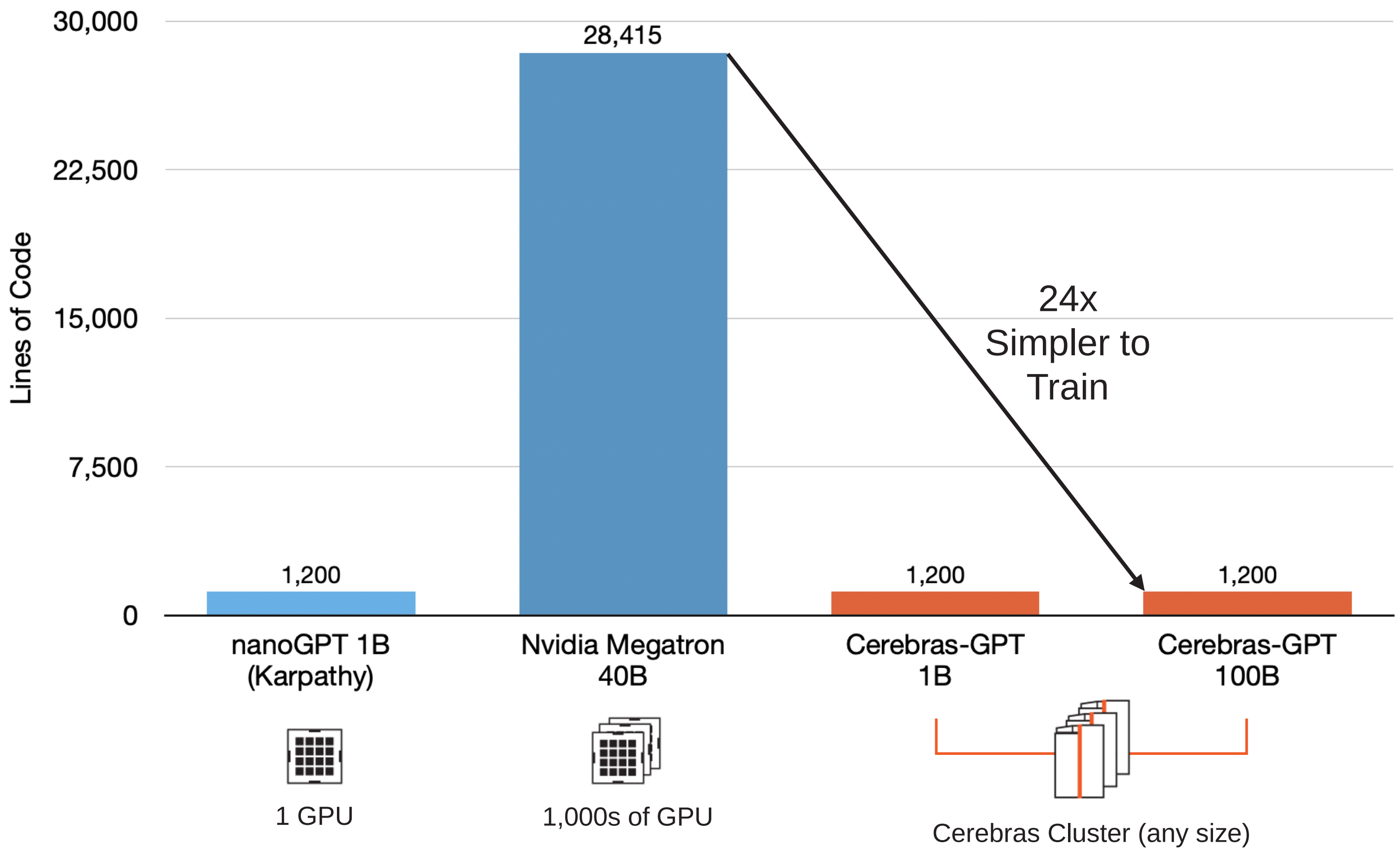
Trained on Andromeda in just weeks!

- Open compute-optimal GPT models up to 13B trained on Cerebras Wafer-Scale Cluster
- Compute optimal scaling law model family 111M, 256M, 590M, 1.3B, 2.7B, 6.7B, 13B

- Hugging Face: huggingface.co/cerebras/Cerebras-GPT-13B
  - Hundreds of thousands of downloads!
- Paper: arxiv:2304.03208



Experiencing Reduced Scaling Complexity



- A 1B parameter is simple to write and train on one GPU
- A large model across a cluster of Cerebras CS-2s is also easy to train
- **On Cerebras, all model sizes have the same code and train the same way**

CG-1: Condor Galaxy-1 Wafer Scale Cluster

64
CS-2 nodes

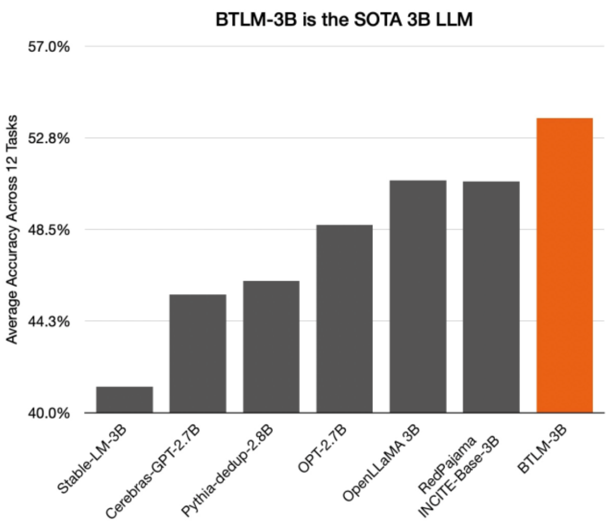
54 million
AI cores

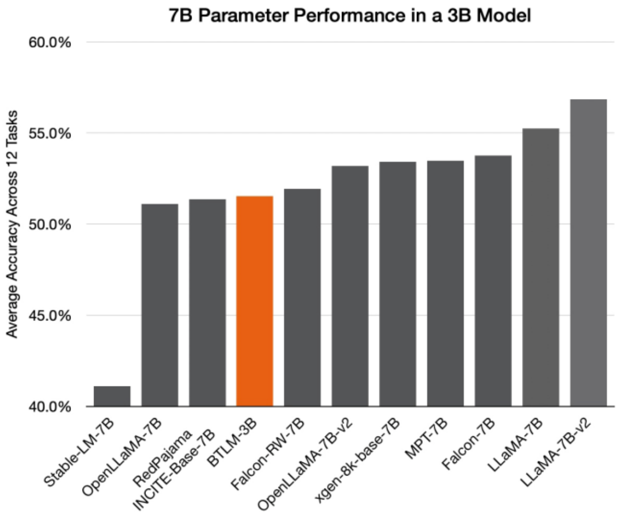
4 ExaFLOPS
Sparse AI compute

BTLM: 7B Model Performance in a 3B Package

First public model trained on CG-1!

- New SOTA benchmark for 3B model
    - Outperforms all existing 3B models
    - Even outperforms many 7B models
    - While trained on less data, 2x less compute
- Most popular 3B model in community
    - Hugging Face: huggingface.co/cerebras/btlm-3b-8k-base
- Commissioned by the Opentensor foundation for use on the Bittensor network





Cerebras BTLM-3B-8K Hits 1 Million Downloads!

Opentensor

Enabling All to Train Largest Models

Scale out capability is critical to pushing to larger models

Wafer Scale Cluster architecture is inherently scalable

1. Largest models on a single device
2. Data parallel only scale-out
3. Native unstructured sparsity acceleration

There’s no end in sight

- Models continue to grow exponentially
- Few companies have access to largest models today
- Cerebras architecture makes running largest models fast and easy

Making the largest models available to everyone

Thank you