AI Hardware Evolution

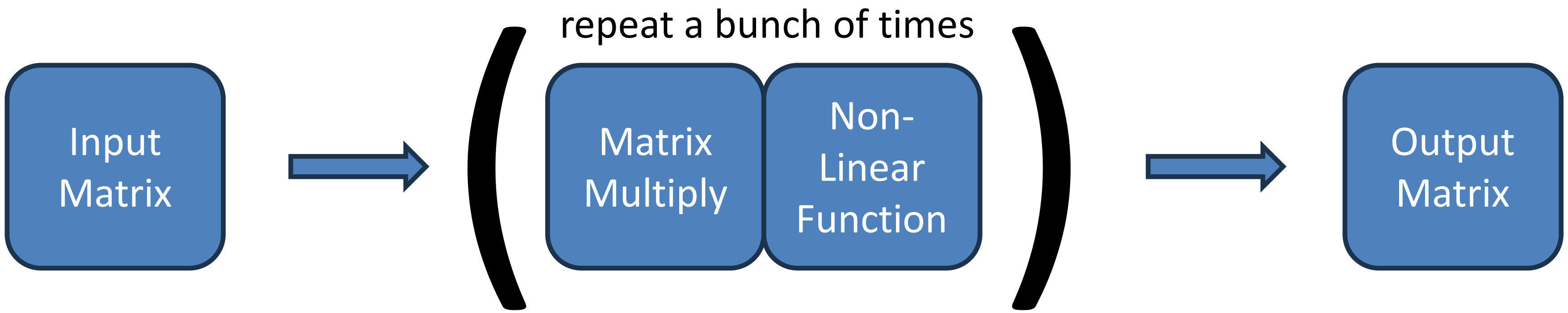
Mike Ferdman and Peter Milder

Brief Intro…

- Peter and I work on hardware accelerators
  - Much of our work is on AI/ML accelerators
  - Primarily targeting FPGAs
  - Two goals of our research
    - Find clever ways of improving computation efficiency
    - Develop techniques for making accelerators easier to program
- Goal of this talk
  - Explain how (I believe) hardware enabled the AI revolution
  - Explain how (I believe) hardware will limit AI progress

The Core of AI Computation

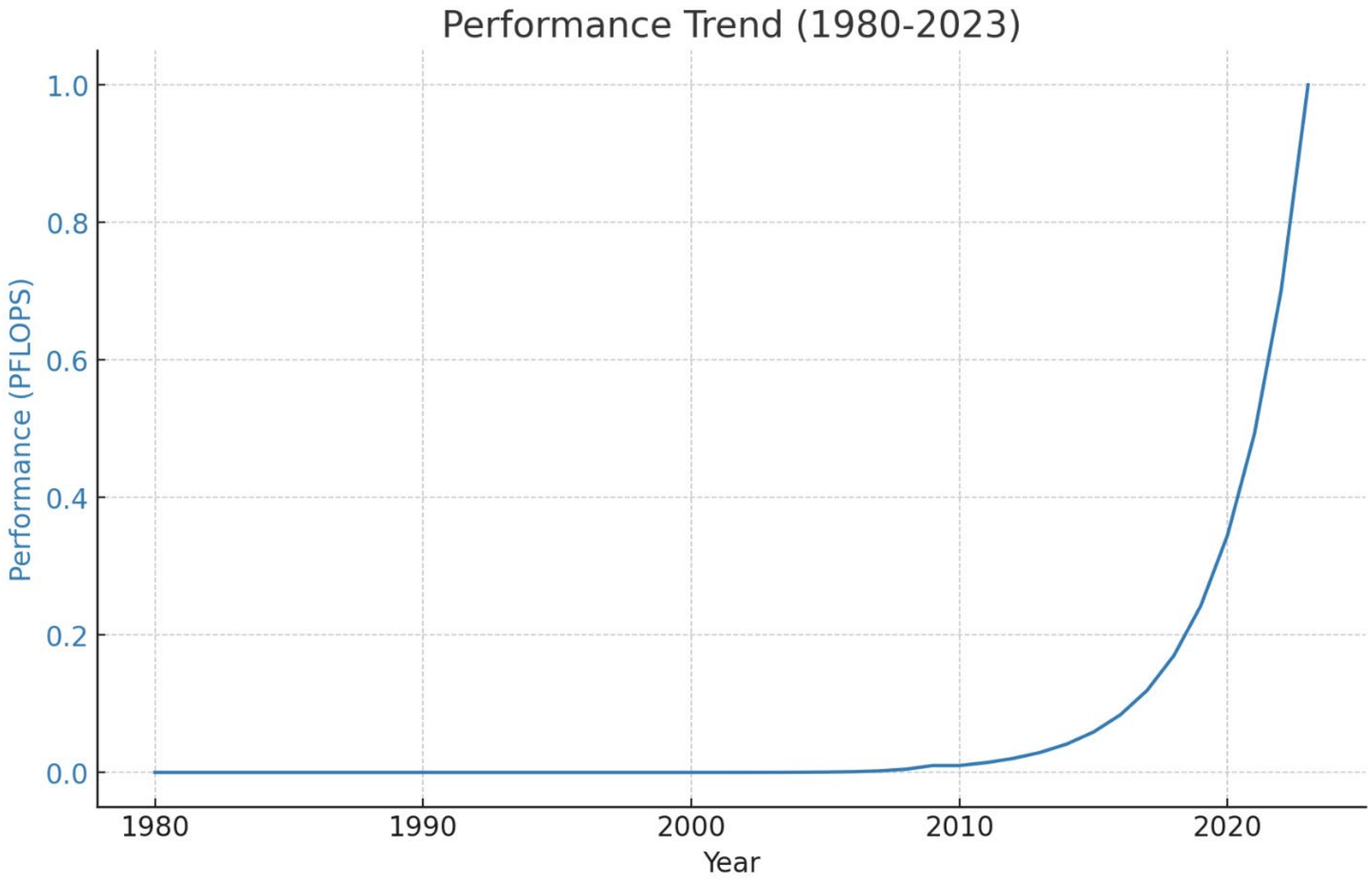
- All compute-intensive tasks are essentially the same: **Matrix Multiply**
  - AI is not an exception
  - IBM figured out how to do Matrix Multiply in the 60s



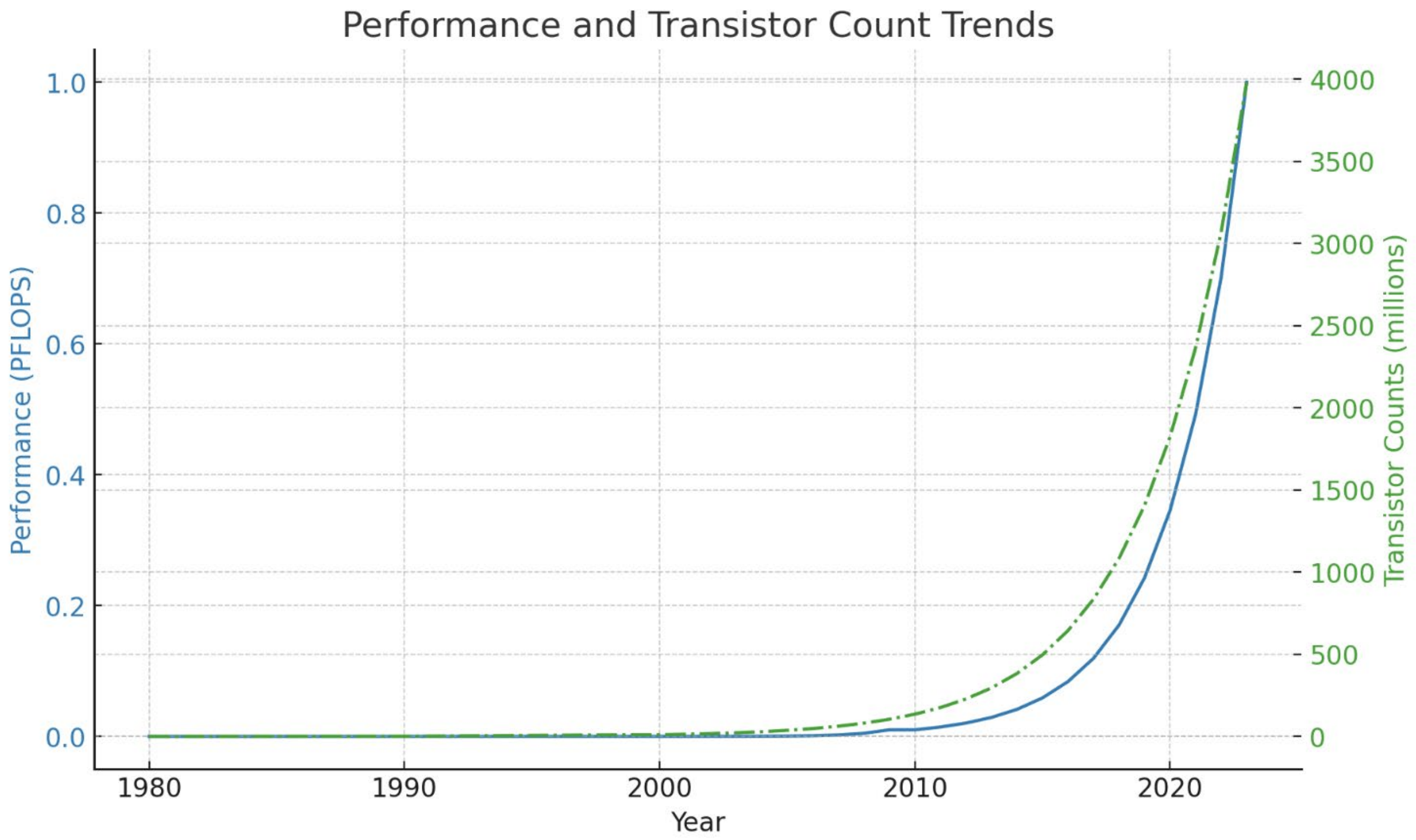
- The key is fast matrix multiply
  - Where did it come from?
  - How long will it last?

Hint: Exponentials always stop

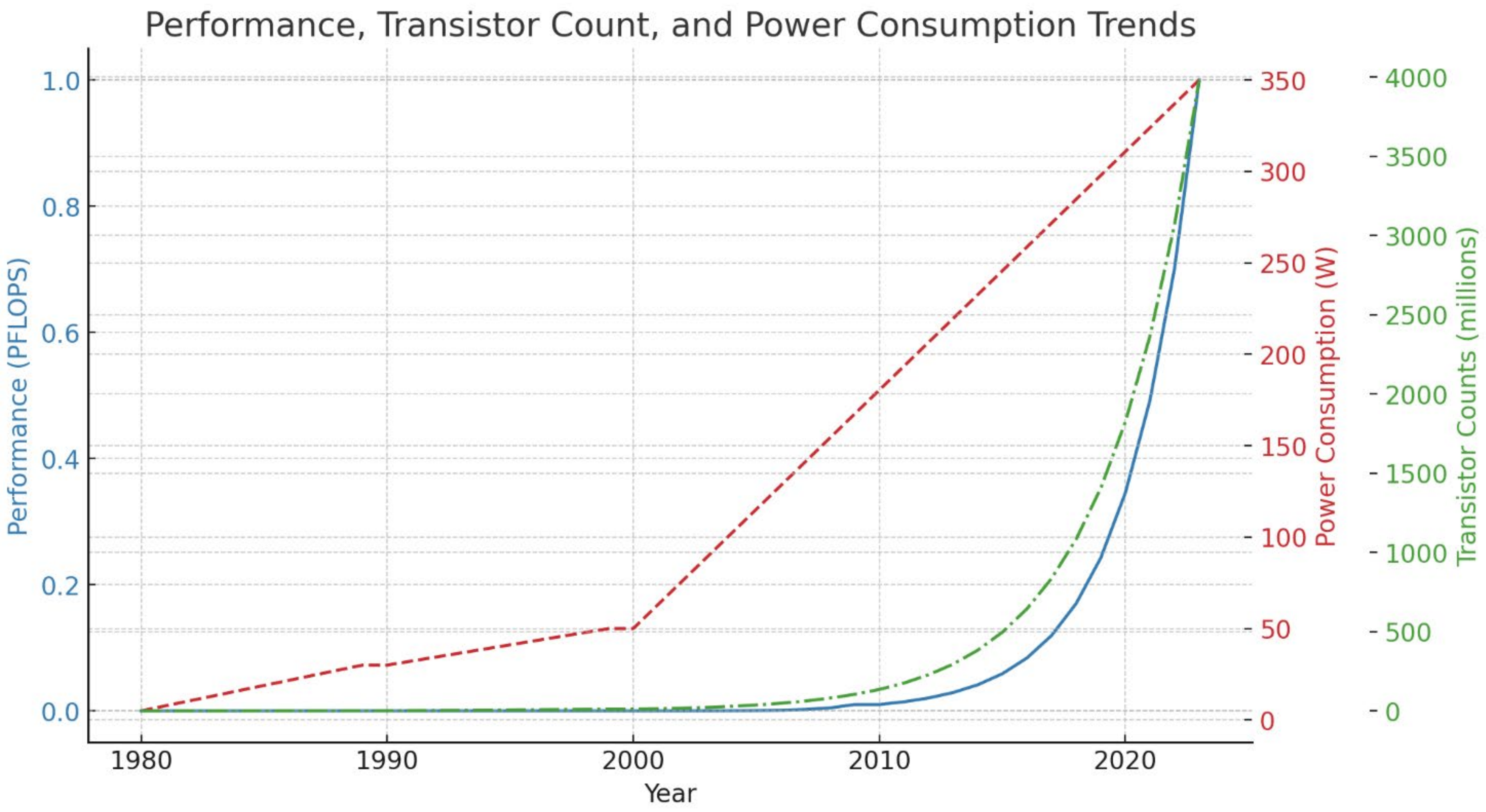
The Data (hallucinated by ChatGPT)



The Data (hallucinated by ChatGPT)



The Data (hallucinated by ChatGPT)



Putting Power in Perspective

- Data center power density per rack
  - Traditional server rack
    - 2x 208V circuits, 30A breakers (per code, shouldn't exceed 80%, so 24A)
    - 10kW per rack (150W per sq. ft.)
  - Nvidia power consumption (8x GPU)
    - DGX A100: 6.5kW
    - DGX B200: 14.3kW
  - "Typical" AI racks: 40kW per rack
    - Steal power from nearby empty racks
- For comparison:
  - NCS building: 300kW total

We have ~4 generations of GPUs left (18 months per generation)

Two Distinct Modes of AI Computation

Training

- Run batches of data through model
- At each batch…
  - Compute gradient
  - Update model

- Time to train? (~GPT4)
  - ~1T parameters
  - 10,000 GPUs for 90 days
- Cost
  - $100M in hardware + 8MW power

Inference

- Run once per output
  - … per word in text generation
  - … per diffusion in image generation

- Time to cheat on a homework?
  - 20 seconds on a series of GPUs
  - ~$0.02
- Cost
  - Estimate 10B tokens per day
  - $20M per month

Both are bad, but I'm interested more in the Inference side

Training is once, but inference is forever

- We train models on a bunch of high-end GPUs
  - Once we're done with training, someone else uses the GPU cluster
  - Quickly moving toward a world where only select few can train models
- We deploy models and continuously do inference on them
  - Efficiency necessary to mitigate hardware cost and power

Mechanisms to Improve Efficiency

- Model distillation
  - Transfer knowledge from larger model to a smaller one
- Quantization / reduced precision computation
  - Use fewer bits (e.g., 8-bit / 12-bit) computation
- Number formats
  - Fixed-point compute
  - Block-float compute (sharing exponents across multiple mantissas)
- Sparsity
  - Lots of zeros from non-linear functions, only multiply the relevant bits

Bottom three are features coming to a hardware accelerator near you

Potential Collaborations

- What we want (from our collaborators)
  - Real-world workloads so we can design (useful) efficient hardware
  - Expose us to AI/ML trends, guiding our work on accelerator programmability
- What we can offer (to our collaborators)
  - Help to build (FPGA-based) inference engines
  - Help evaluate, optimize, and tune AI/ML hardware platforms

Our BNL Collaborations

- Neural network models on FPGAs w/Ray Ren
  - Real-time data compression for sPHENIX
    - Disks aren’t fast enough to write all data coming from experiment
  - Evaluating FPGA implementations of models for ATLAS

- FPGA Virtual Memory Support w/Lingda Li
  - Explore advanced Virtual Memory support on FPGAs
  - Study Unified Memory assist for hardware-accelerated apps
    - Large memory applications that don’t fit into device memory
    - Need to move data between host and device memory
    - Provide “smart” Virtual Memory support infrastructure

Thanks!