

Architecting Waferscale Processors

A GPU Case Study

Saptadeep Pal*, Daniel Petrisko, Matthew Tomei, Puneet Gupta*,
Subramanian S. Iyer*, Rakesh Kumar*

University of Illinois at Urbana-Champaign

*University of California, Los Angeles

HPCA 2019

Presented by Amirhossein Heidari
Seminar on Computer Architecture

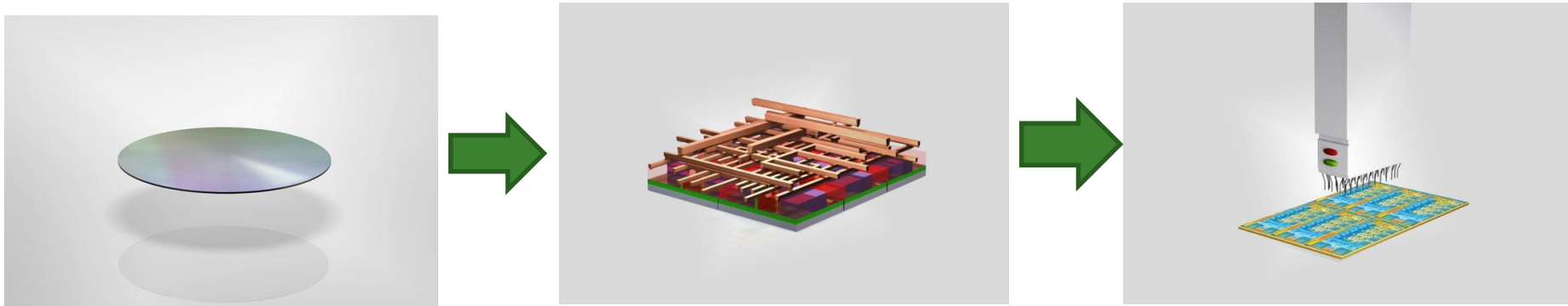
Executive Summary

- Parallel hardware requires **low overhead communication** between different computing nodes.
 - **Problem:** Overhead of communication has been increasing at an alarming pace.
 - **Goal:** Explore and evaluate the possibility of using waferscale processor.
 - **Solution:** Prototype a GPU architecture and study its performance.
 - **Evaluation:**
 - Significant performance and energy efficiency advantages
 - Outperforms state-of-art scheduling and data placement policies
-

Background

Waferscale processors

- Traditionally processors are manufactured by using **one** wafer for many copies of a single processor.
- Largest size of the die is determined by the yield.
- Post manufacturing, the wafer is diced into individual processor dies which are packaged and integrated into a parallel system



Waferscale processors

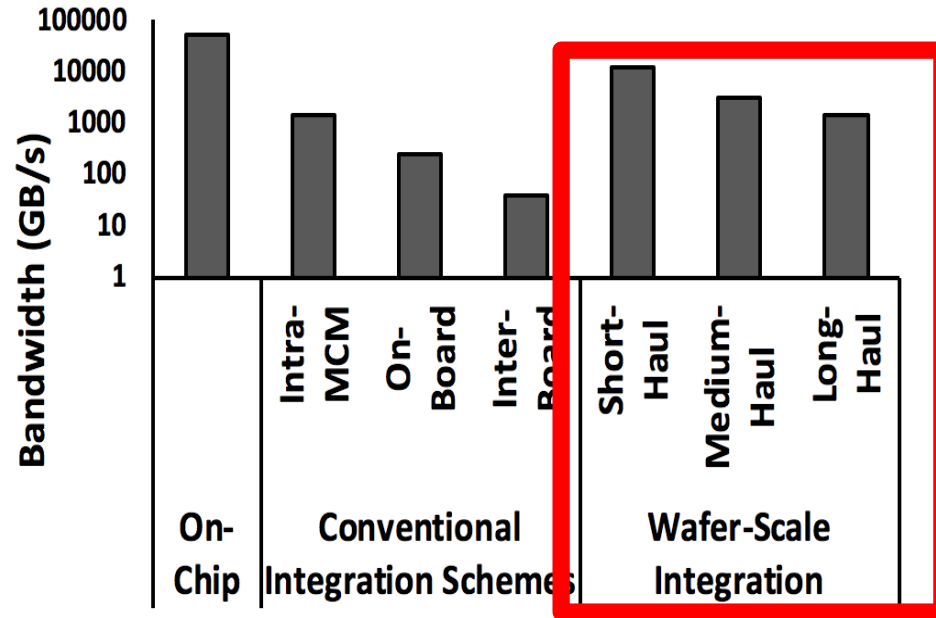
- In Waferscale the **wafer is the processor**.
- Either a monolithic processor is designed to be as large as an entire wafer.
- Or a set of processors are designed that continue to reside on the wafer and the processor die are connected on the wafer itself using a low cost interconnect.

Waferscale processors

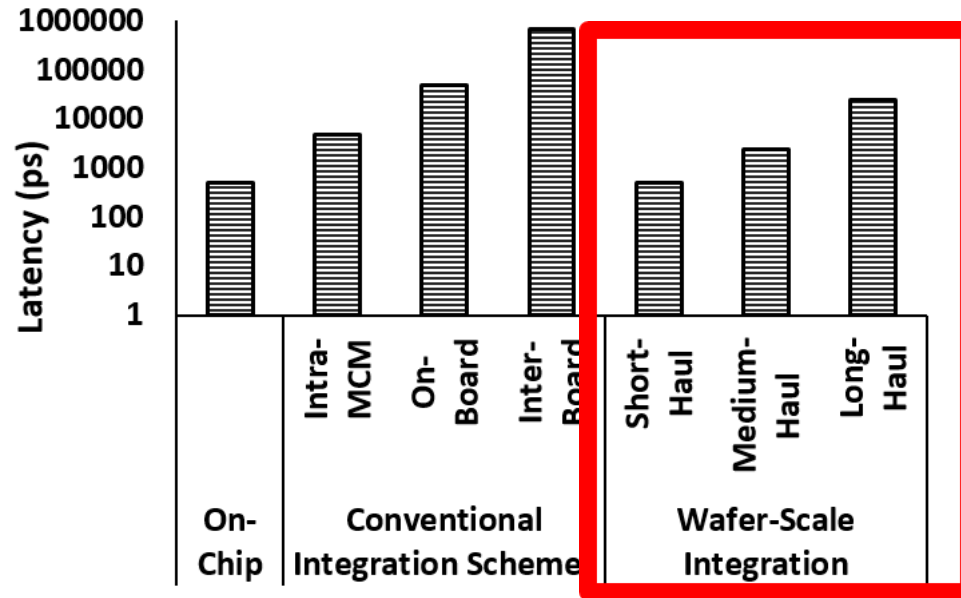
- Enables much **larger bandwidth** than what conventional integration schemes can provide.
- Links are smaller and high density
- Simple parallel communication protocol can be used where a massive number of links run at relatively lower frequency

Bandwidth Vs Conventional Schemes

Bandwidth comparison



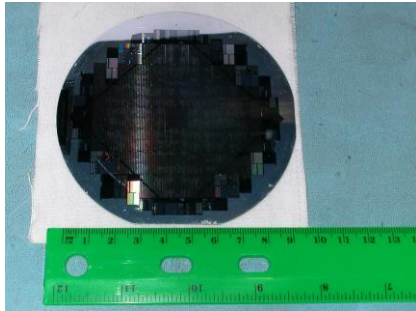
Latency Vs Conventional Schemes



Latency Comparison

Waferscale processors

- Studied heavily in the 80s but abandoned due to **yeild issues** .
- The larger the size of the processor the lower the yield .
- Nowadays considerable advances have been made in manufacturing and packaging technology.
- It is now possible to bond pre-manufactured dies **directly on the wafer**.
- Connecting them through **Silicon interconnect (SI-Fi)**.
- Potential benefits are much larger now .

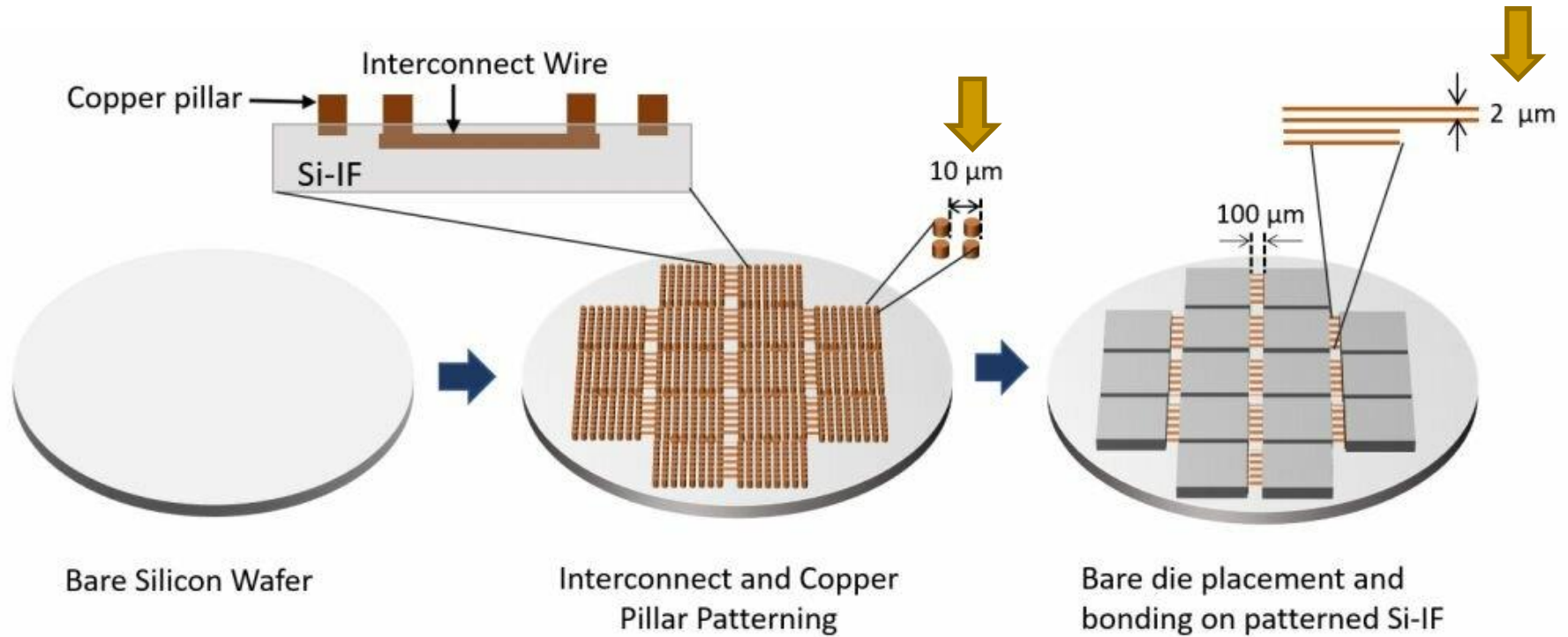


Trilogi Sytems
(1980 – 1985)
210 million
dollars
investment

Readiness

Silicon Fabric Interconnect

- Replaces the organic printed circuit board (PCB).
- Silicon substrate allows placing and bonding bare silicon dies **directly on to the thick silicon wafer** using copper pillar based I/O pins.
- Smaller high-yield dies are interconnected on the passive interconnect substrate using mature fabrication techniques.
- Different system components such as processors and VRM can be directly bonded on the Si-IF.



The system assembly process flow is shown. Interconnect layers and copper pillars are made by processing the bare silicon wafer. Bare dies are then bonded on the wafer using thermo compression bonding (TCB).

Yield issues

- Three components to the final yield
 - 1) The Die → can be ensured using known-good-die testing.
 - 2) Copper pillar → ensured to be higher than 99%.
Not prone to extrusions unlike solder based connections.
 - 3) Si – IF substrate → high since it is a passive wafer with only thick interconnect wires and no active devices.

Yield issues

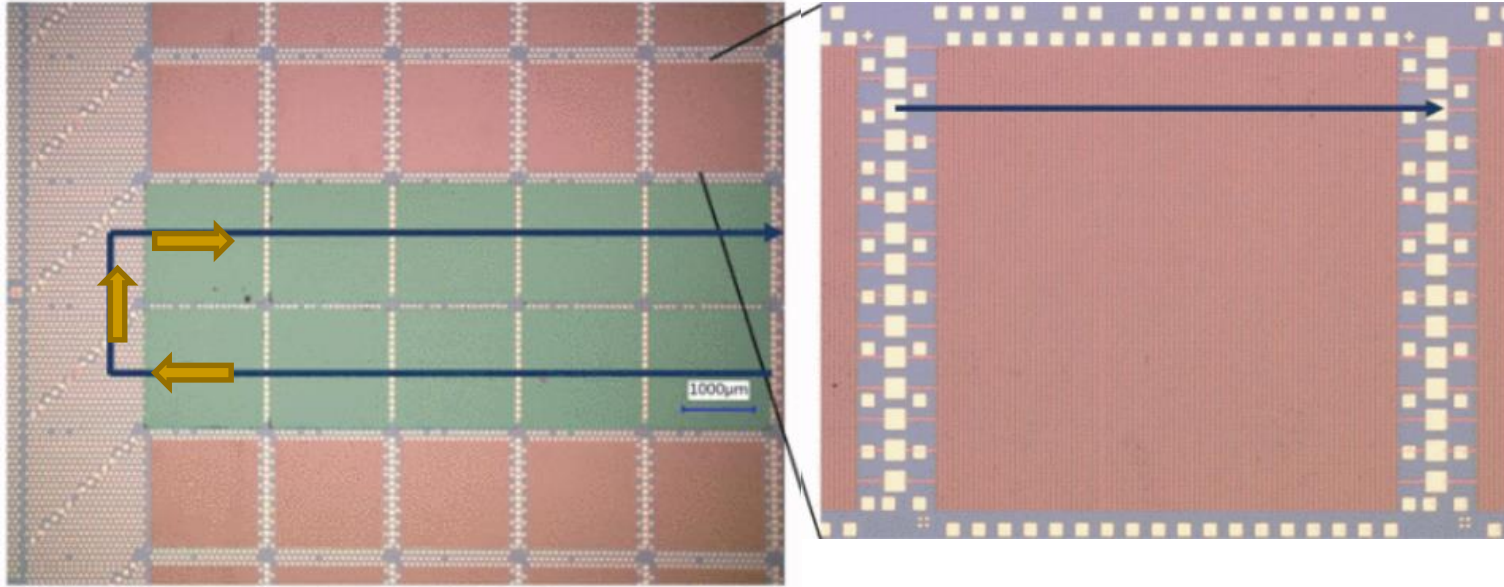
- The expected yeild is calculated for different number of metal layers and metal layer utilization using industry standard **yield modeling equations**.

Si-IF Metal Layer Utilization (%)	Number of Layers		
	1	2	4
1	99.6	99.19	98.39
10	96.05	92.26	85.11
20	92.29	85.18	72.56

Prototype

- To assess viability of inter-die interconnect on Si-IF, the authors built a prototype.
- Bonded connectivity testing dielets on a 100nm wafer-scale Si – IF.
- Electrical tests show that 100% of the interconnects in this prototype were connected.
- Thermal cycling showed that all the copper pillars and interconnects withstood the cycles without any noticeable degradation.

Prototype



Ten 4 mm² dies are bonded and tested for continuity of a signal across the dies.

Case For GPUs

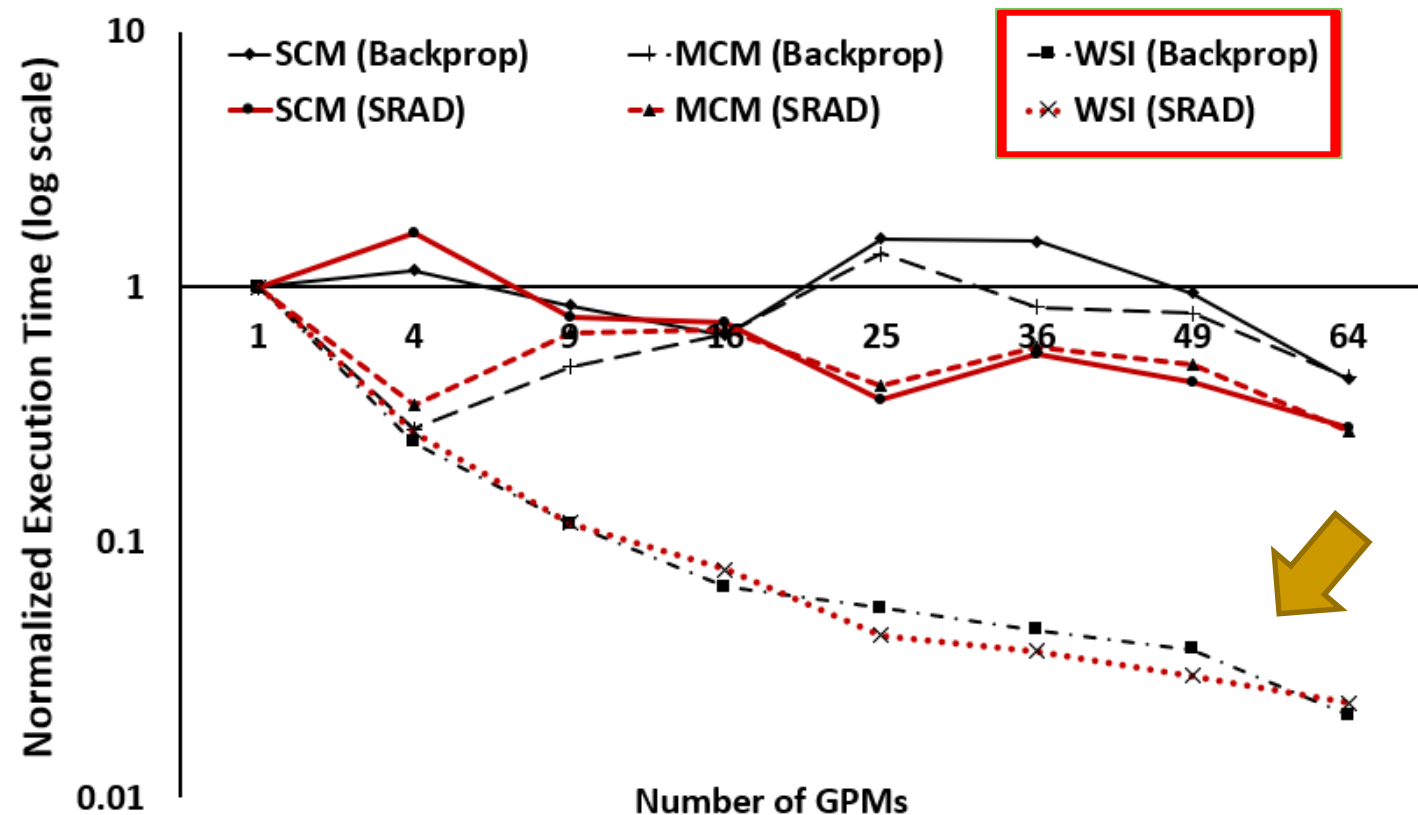
- GPU applications have large amounts of parallelism.
- Limited only by cooling and yield.
- Large class of applications from the domain of physics simulations, linear algebra and machine learning.
- Great benefits from increasing the effective size of a GPU.

GPU Constructions Considered

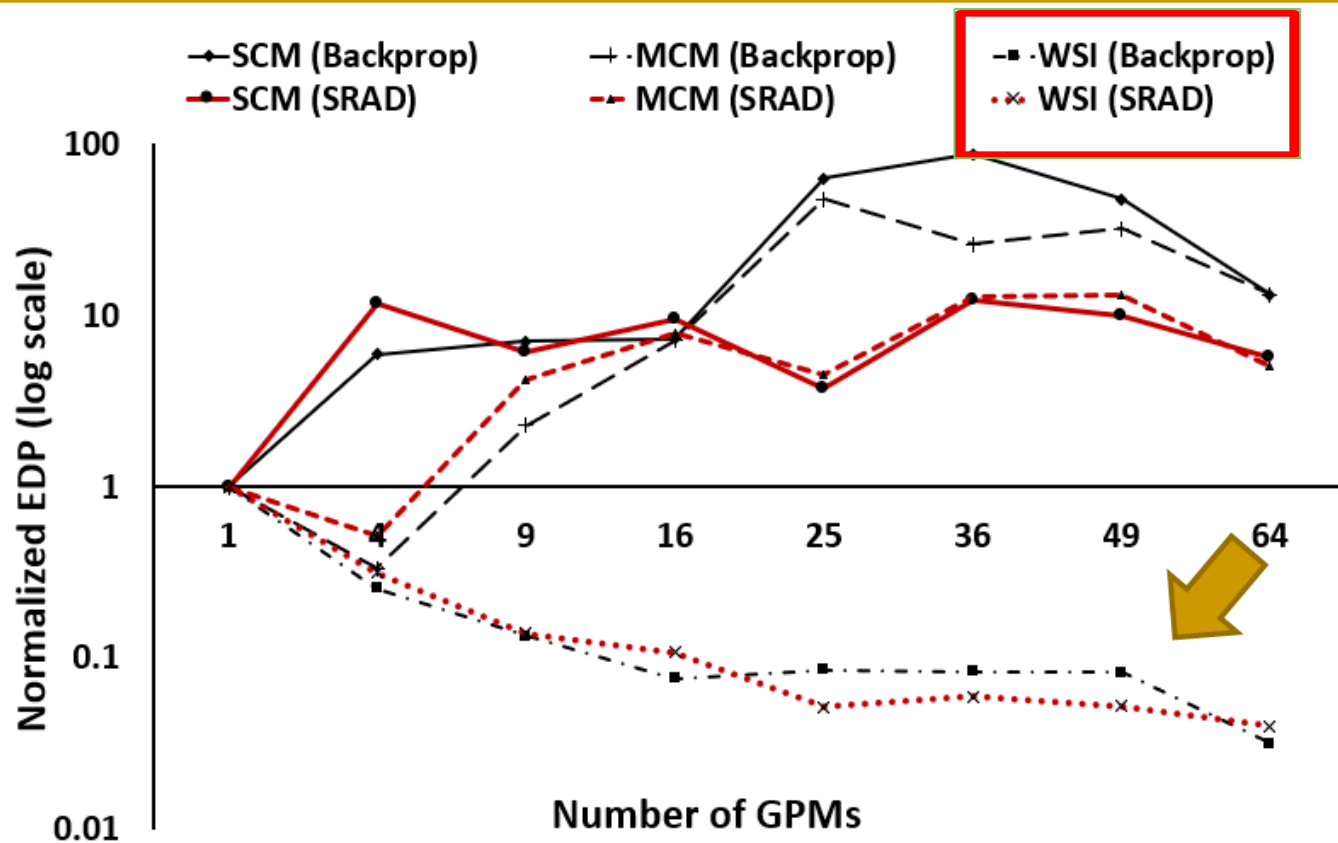
- ScaleOut SCM-GPU (single-chip module GPU), Where each GPM (GPU module) is constrained in its own package.
- ScaleOut MCM-GPU, units are placed in a 2d mesh on a traditional PCB connected with a QPI-like link.
- Hypothetical Waferscale GPU.
- Constitute a **single logical GPU** from the perspective of the programmer.

- Demonstrate the potential benefits of a waferscale GPU.
- **SRAD** and **Backprop** .
- From Rodinia benchmark suite.
- Chosen to represent medical imaging and machines learning.
- Both fields benefit massively from waferscale processing.
- Simulations performed by using gem-GPU to generate memory traces and activity profiles which are fed to a GPU simulator.

Benchmarks For Execution Time



Benchmarks For EDP



Benchmark Results

- **Backprop**: a 47.5x speedup for 64 GPM waferscale GPU over a single GPM system.
- 20.8x and 21.13 over the highest performing ScaleOut
- Speedups limited by memory transfer latency
- **SRAD**: 42.5x speedup over single GPMs
- 24.8x over ScaleOut
- **Without requiring changes to the programming model**

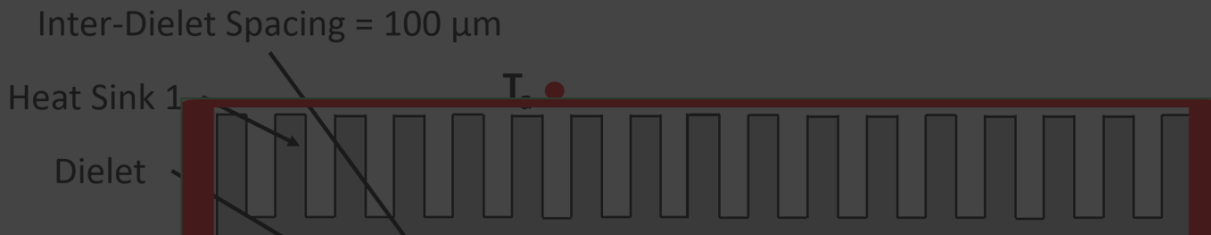
Takeaways

- GPU architectures are a good fit
- Performance and energy efficiency scaling of GPU applications is much stronger on a waferscale GPU
- Worth the effort to explore further the limits and constraints of waferscale GPUs in terms of :
 1. Thermal constrains
 2. Power Delivery
 3. Network Architecture

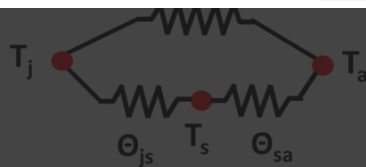
Waferscale GPU Architecture

- **Goal:** Find feasible GPU architectures.
- Unique problem due to the **physical constraints**.
- Needs to operate at kilowatts of power, the architecture must be feasible in presences of the associated **thermal and power delivery concerns**.
- Will also need a considerable amount of **interconnection resources**.
- 500 mm² GPU die, 200 mm² DRAM dies, TDP of 200W and 70W respectively

Schematic Cross-Section



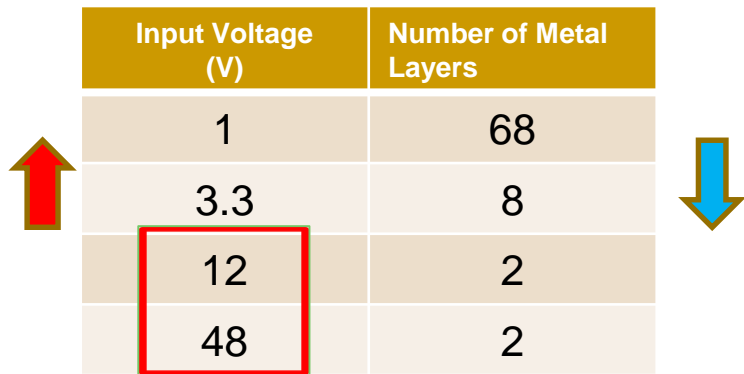
Target Junction Tem- perature (°C)	Dual Heat Sink			Single Heat Sink		
	Power (W)	Num GPMs w/o VRM	Num GPMs with VRM	Power (W)	Num GPMs w/o VRM	Num GPMs with VRM
120	9300	34	29	6900	25	21
105	7600	28	24	5400	20	17
85	5850	21	18	4350	16	14



Resistance	(°C/W)
Θ_{ja}	0.014
Θ_{js}	0.025
Θ_{sa}	0.02

Power Delivery Considerations

- Constrained by the heat sink to a total TDP (thermal dissipation power) of up to 9.3 kW.
- Network must be able to deliver 12.5 kw of power.
- The system is **area constraint** not TDP constraint.



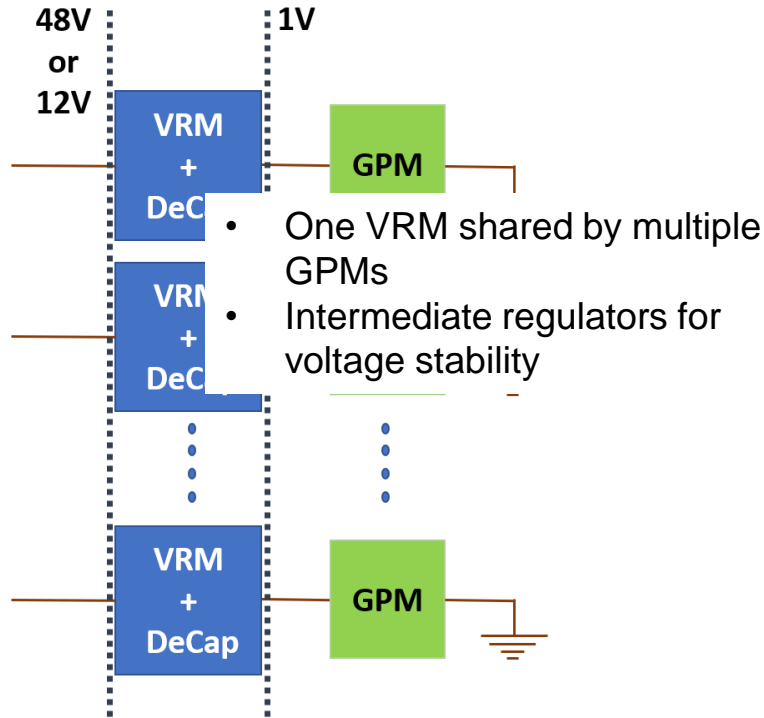
A diagram illustrating the relationship between input voltage and the number of metal layers. A table is shown with two columns: 'Input Voltage (V)' and 'Number of Metal Layers'. The table has four rows. A red arrow points upwards from the bottom row to the top row, and a blue arrow points downwards from the top row to the bottom row. The bottom row (48V, 2 layers) is highlighted with a red border.

Input Voltage (V)	Number of Metal Layers
1	68
3.3	8
12	2
48	2

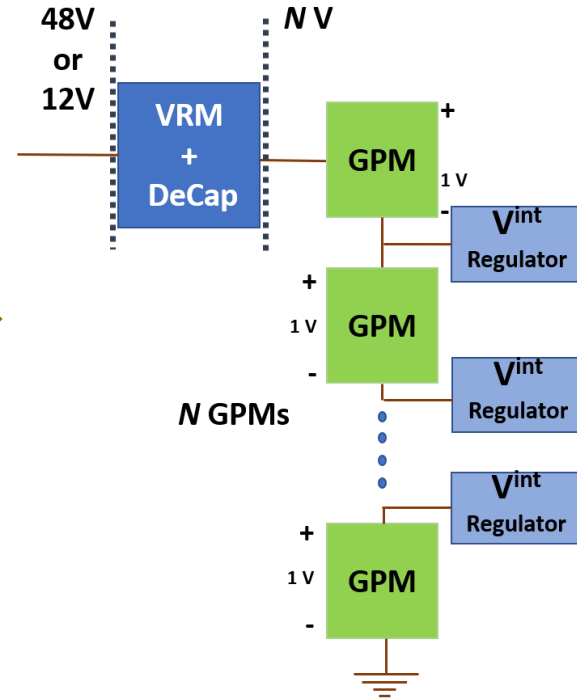
Power Delivery Considerations

- If N GPMs are stacked the supply voltage to the stack should be N times the supplied voltage required for one GPM.
- Reduced per GPM footprint
- 34 GPMs can be accommodated with 4 Gpms per stack and 48V power supply to the wafer.

Power Delivery Considerations

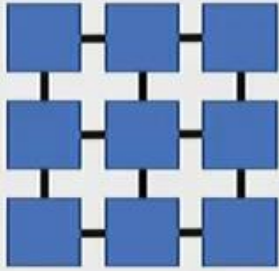


One VRM per PDM

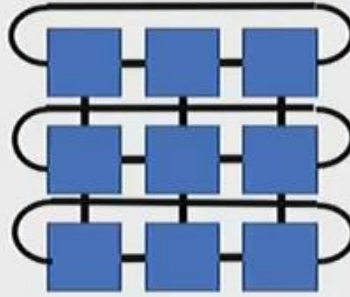


Stacked voltage supply

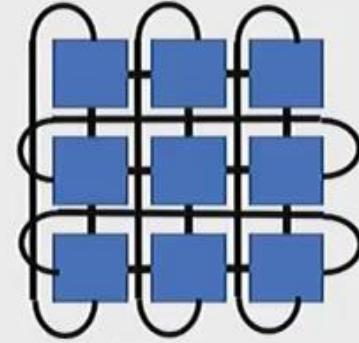
Inter-GPM network



Mesh



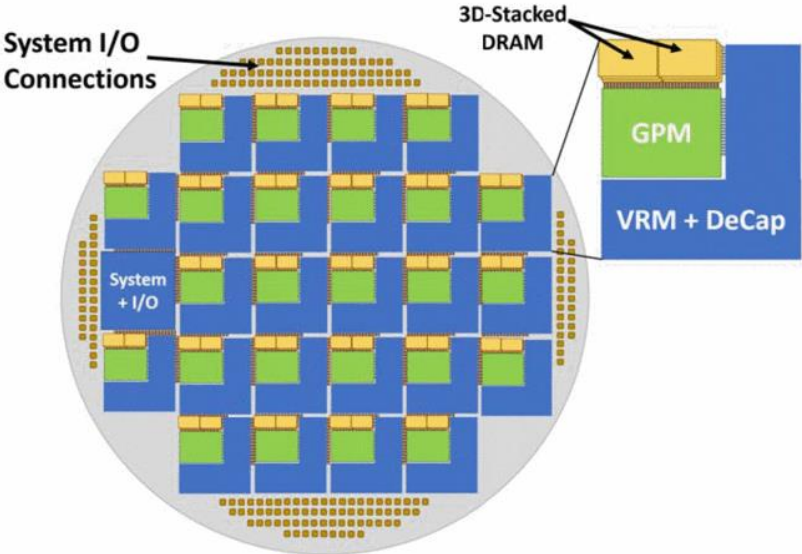
1D-Torus



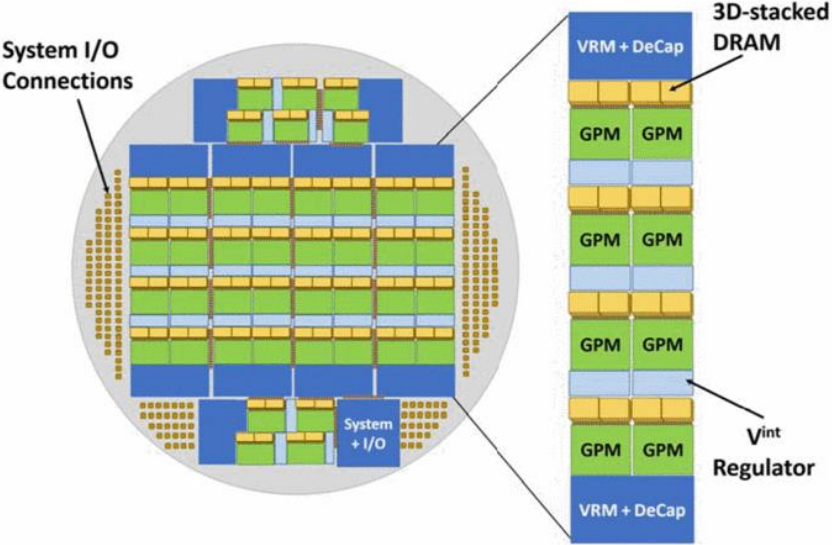
2D-Torus

Num. Layers	Topology	Inter-GPM BW (TBps)	Si-IF Yield
1	Mesh	0.75	95.9%
2	Mesh	1.5	91.9%
	1-D Torus	1.5	84.3%
3	2D Torus	1.9	74%

Overall Architecture

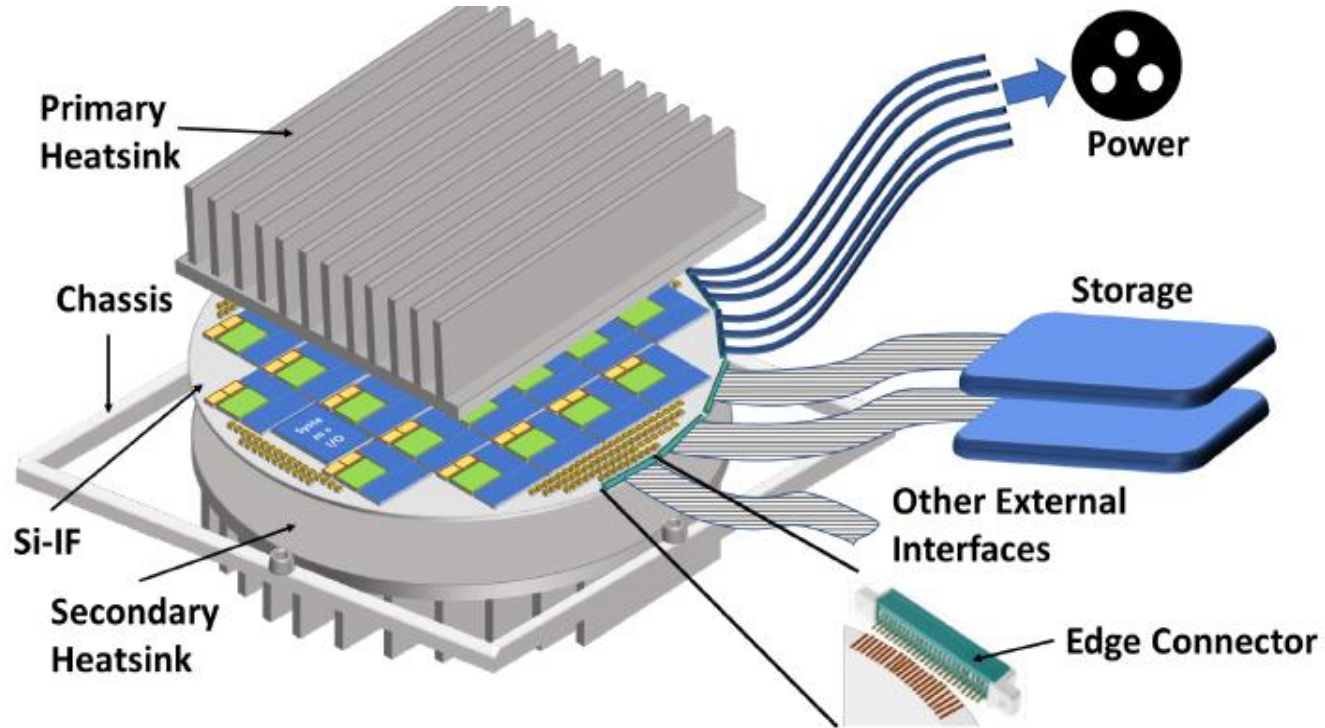


25 GPM units , 2 3D-stacked DRAM per unit,
No voltage stacking



42 GPM units, 2 3D-stacked DRAM per unit,
With Voltage stacking

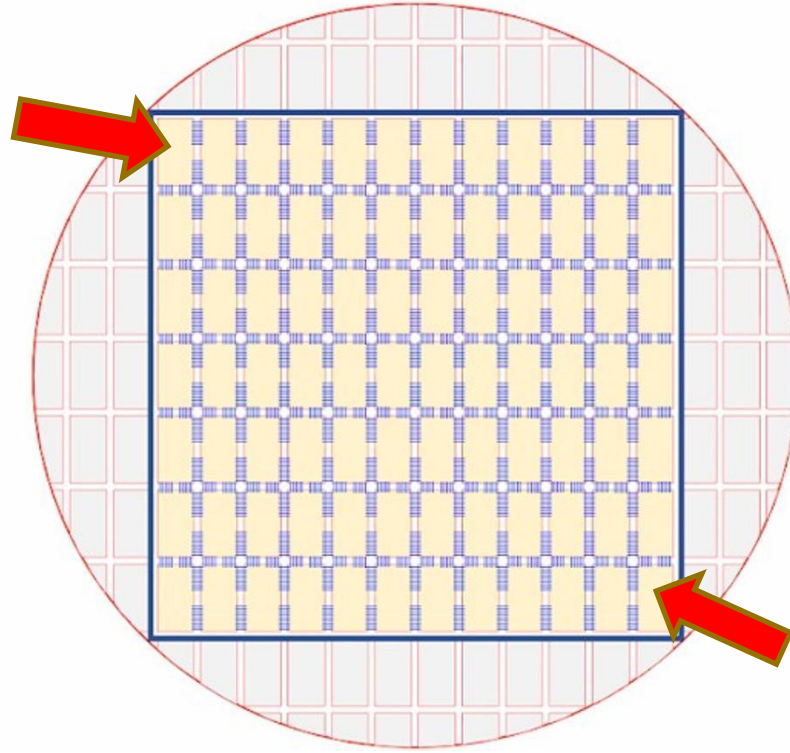
System integration



Thread Block Scheduling and Data Placement

- Performance will also depend on how compute and data is distributed across the system
- Conventionally thread blocks are dispatched to the compute units in a round-robin order based on availability
- Such a fine grained could place the **threads across multiple GPMs**
- Thus destroying the performance and energy results.

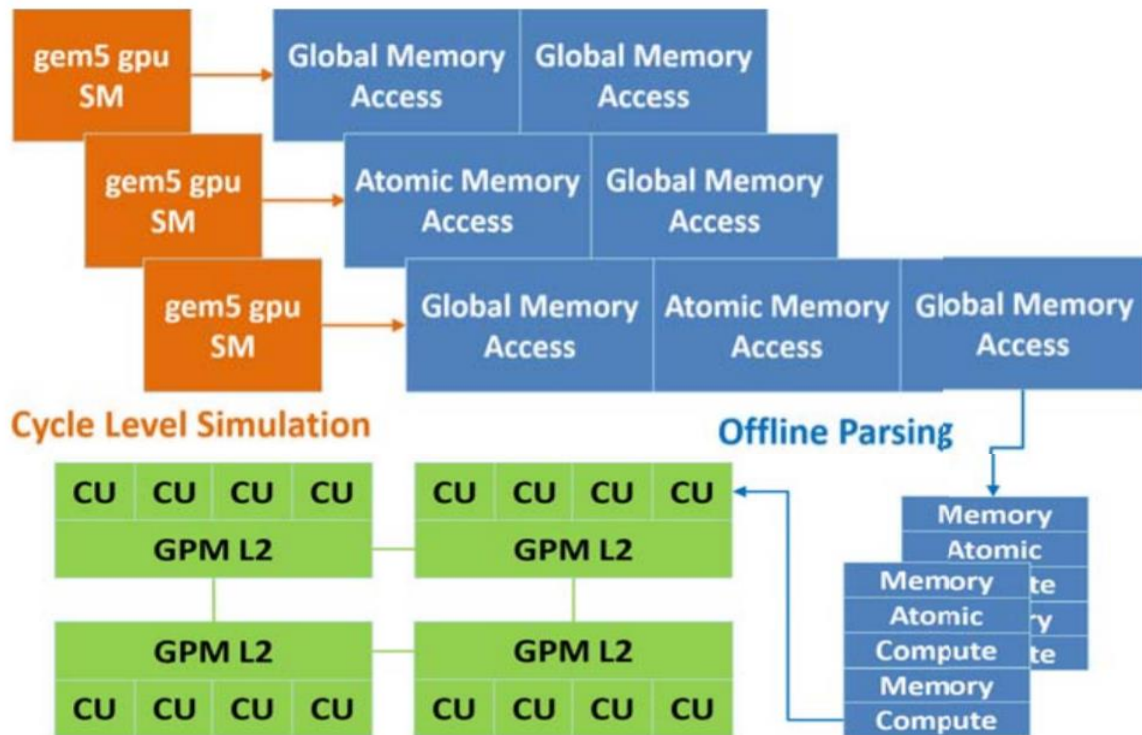
Thread Block Scheduling and Data Placement



Scheduling and Data Placement

- Distributed Scheduling instead of centralized scheduling.
- Data Placement is first-touch, When the first memory access to a page is done the page is moved to the local DRAM of the GPM from which the memory reference was made.
- Policies that allow TBs which share a large amount of data to be placed on **neighbouring GPMs** to minimize data access latency.

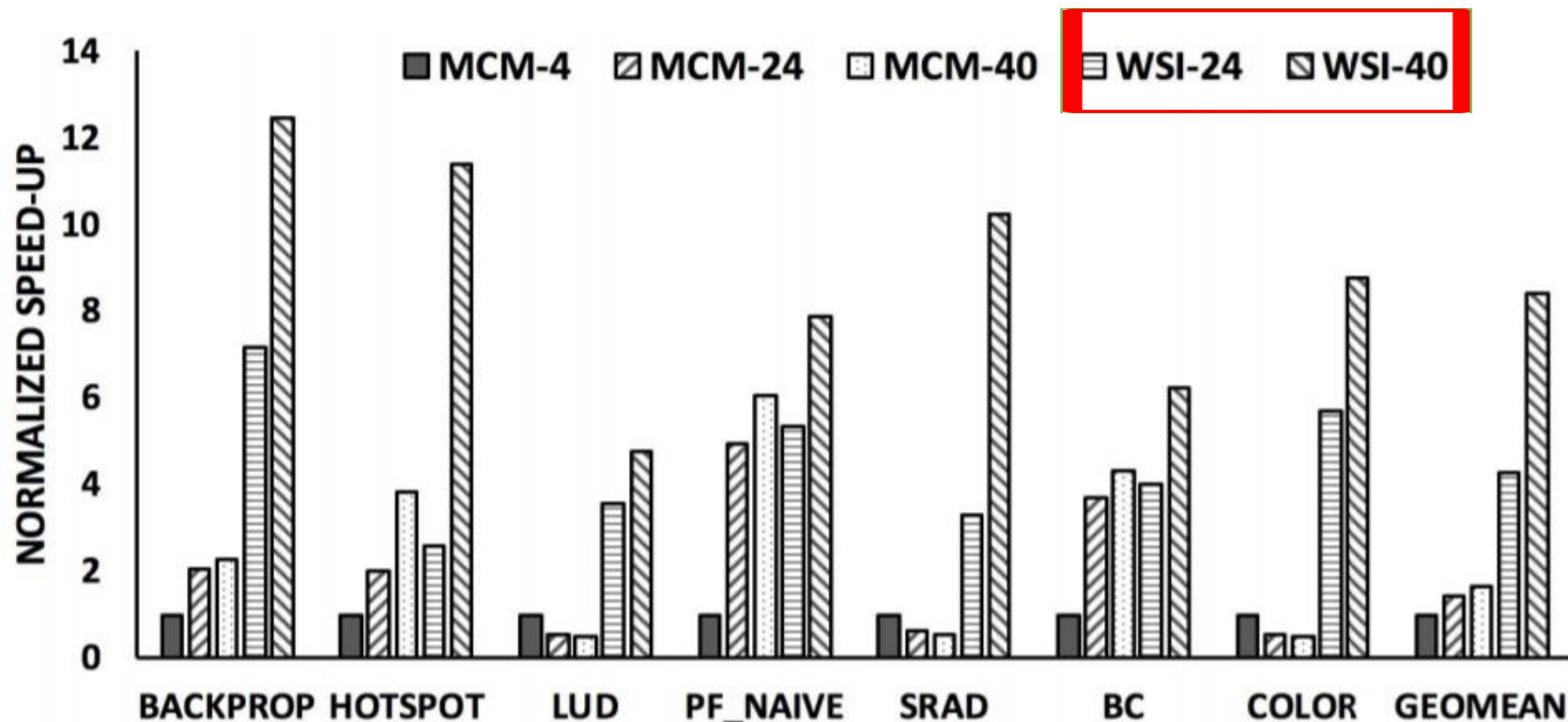
Benchmarks Methodology



1. Collect memory traces.
2. Run each benchmark until beginning of the region of interest
3. Run ROI of application in detailed mode, collecting memory trace of every global read, write and atomic operation.

Parse the traces, gather relative timing virtual address and type of operation

Results



Benchmarks

- A 40-GPU wafer ran benchmarks an average of **5.2x** faster, and a maximum of **18.9x**, compared to a scaled-out 40-MCM configuration (a board of ten four-GPU packages). The 24-GPU wafer outran its competition (a board of six four-GPU packages) by an average of **2.3x**, and a maximum of **10.9x**.
- Researchers attributed the speed-ups to the Si-IF's higher data bandwidth compared to the on-board network in the MCM configurations.

Conclusion and takeaway

- The wafer GPUs they devised ran at a relatively modest clock speeds: 575 MHz for the 24-GPU one and 408 MHz for the 40-GPU version. If higher frequencies could be used, the researchers claim their performance advantage would also increase.
- Whether Waferscale ever makes it out of university labs remains to be seen. Commercial viability is often a tricky thing, even with technology that appears to be poised for productization. If these researchers really believe waferscale is ready for prime time, perhaps a spin-off is in the cards.

Analysis

Strengths

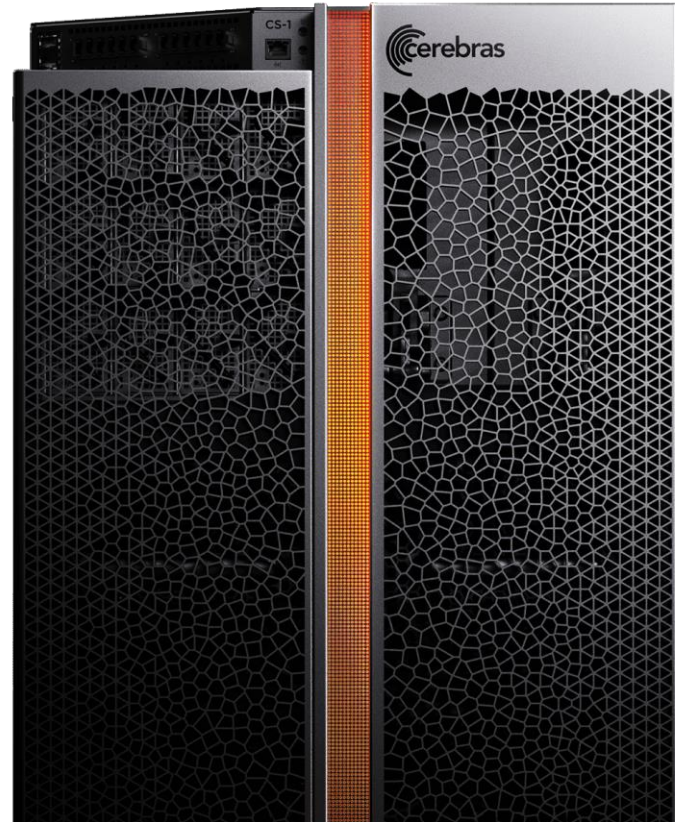
- Focuses on solutions to contemporary problems.
- Well-written, insightful paper.
- Results based approach.
- Fully explores each problem and potential solutions.
- **Solution:** Revisits already proven concept combining it with new solutions.
- **Evaluation:** Takes into account the circumstances surrounding the research.

Weakness

- Lots of specific vocabulary making it difficult to follow.
- Problems from every angle hard to pin-point a central idea.
- **Solution:** Did not take into consideration the aspects of financial and production.
- **Evaluation:** not evaluated against high workloads.
- Only a narrow benchmark suite use.

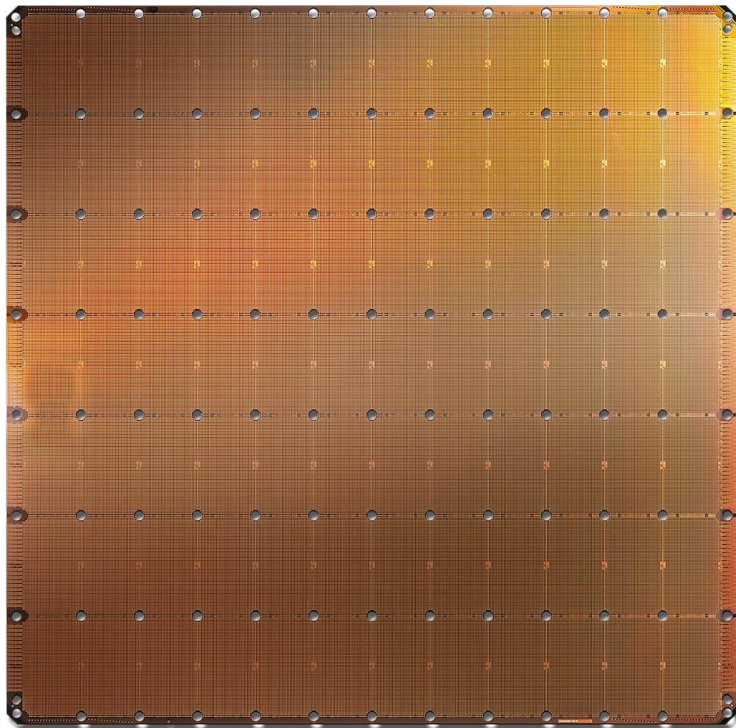
Discussion

Is this feasible to bring to market?



Discussion

What about failure mechanisms?

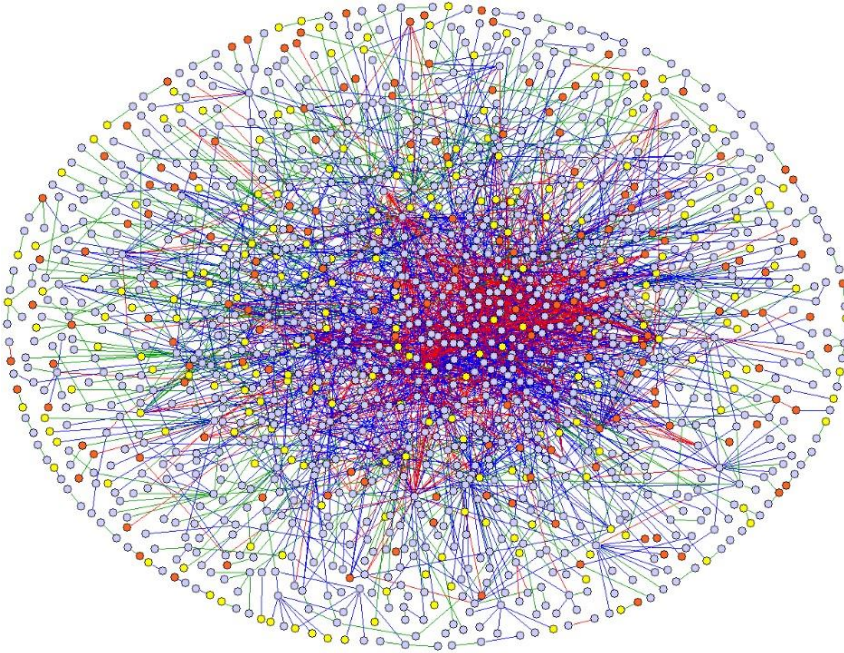


The Cerebras Wafer Scale Engine is 46,225 mm² with 1.2 Trillion transistors and 400,000 AI-optimized cores.



Discussion

What problems require a new type of architecture like this?



Each node = compute unit