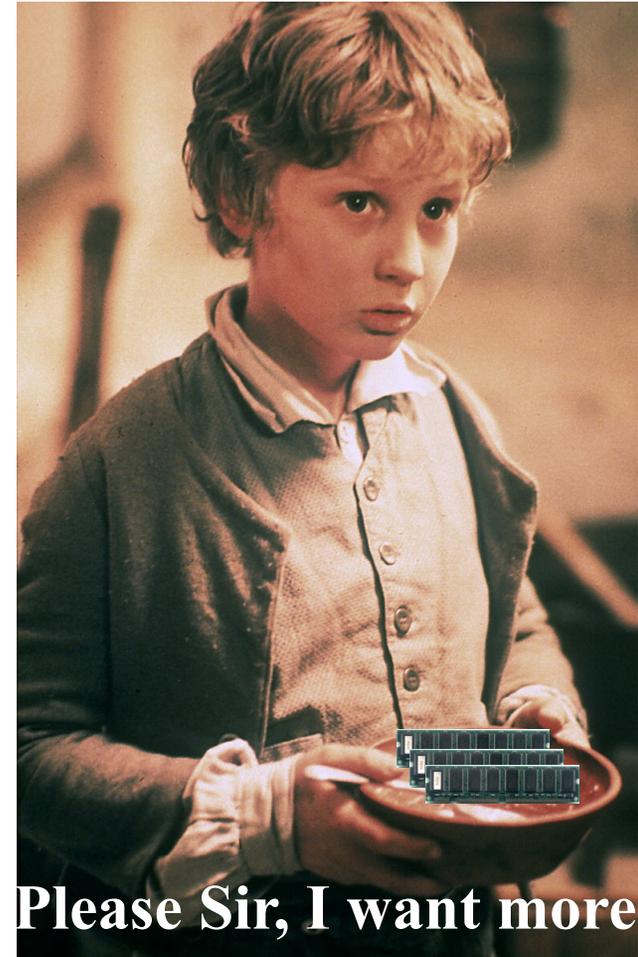


PIM & Memory: The Need for a Revolution in Architecture

Peter M. Kogge
McCourtney Prof. of CSE
Univ. of Notre Dame
IBM Fellow (retired)
7/29/13



Please Sir, I want more

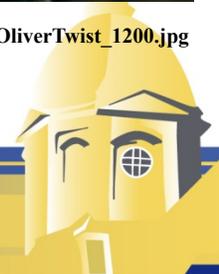
http://www.ket.org/pressroom/2000/38/MPT_OliverTwist_1200.jpg



UNIVERSITY OF
NOTRE DAME

ATPESC July 29, 2013

*ENABLING
INNOVATION*



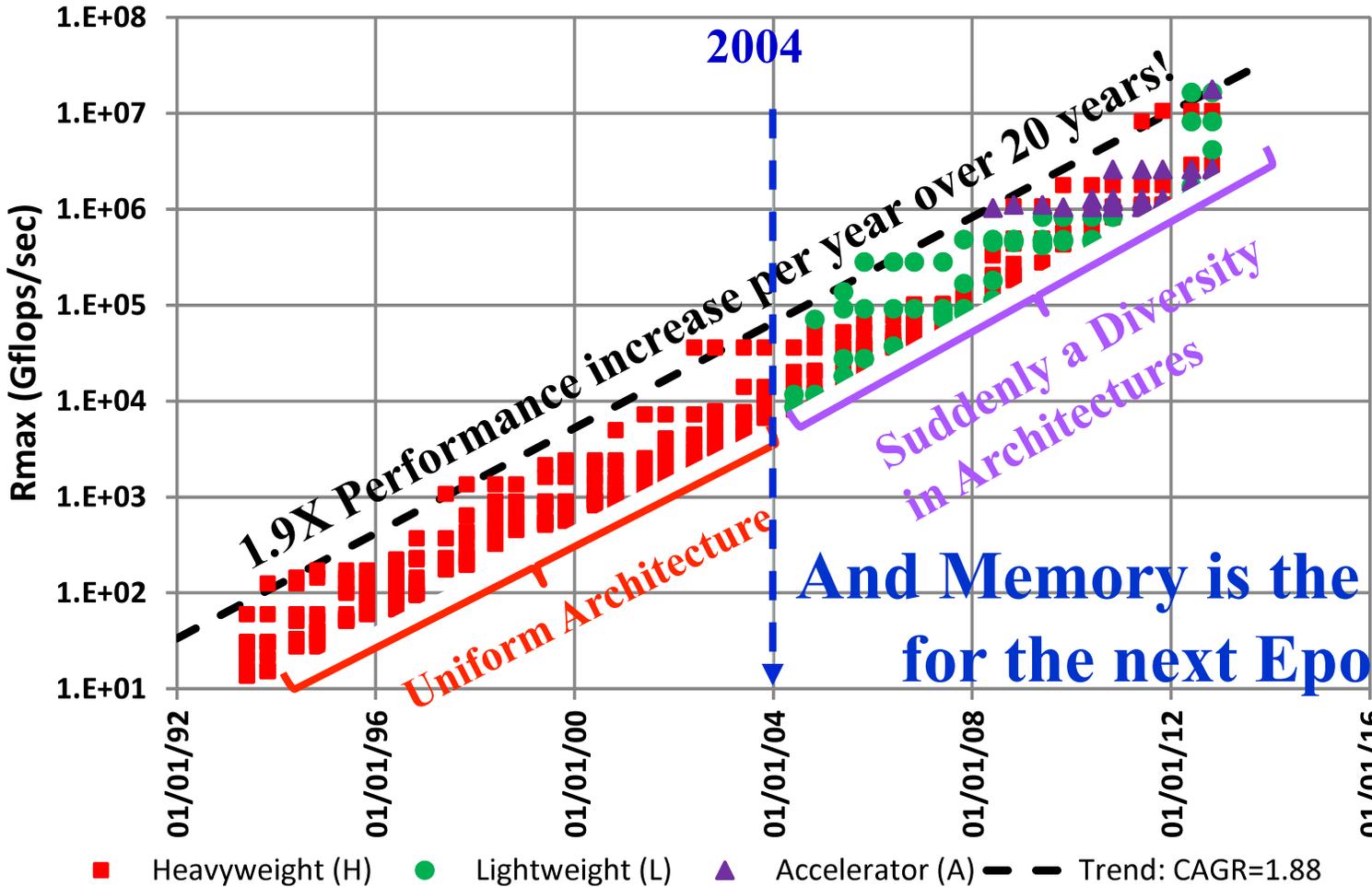
Comment

- Following has *lots* of charts and pictures
- Key take-aways are ***trends***
- Original charts in 2008 Exascale report
- Updates in SC11 paper
- More updates shortly

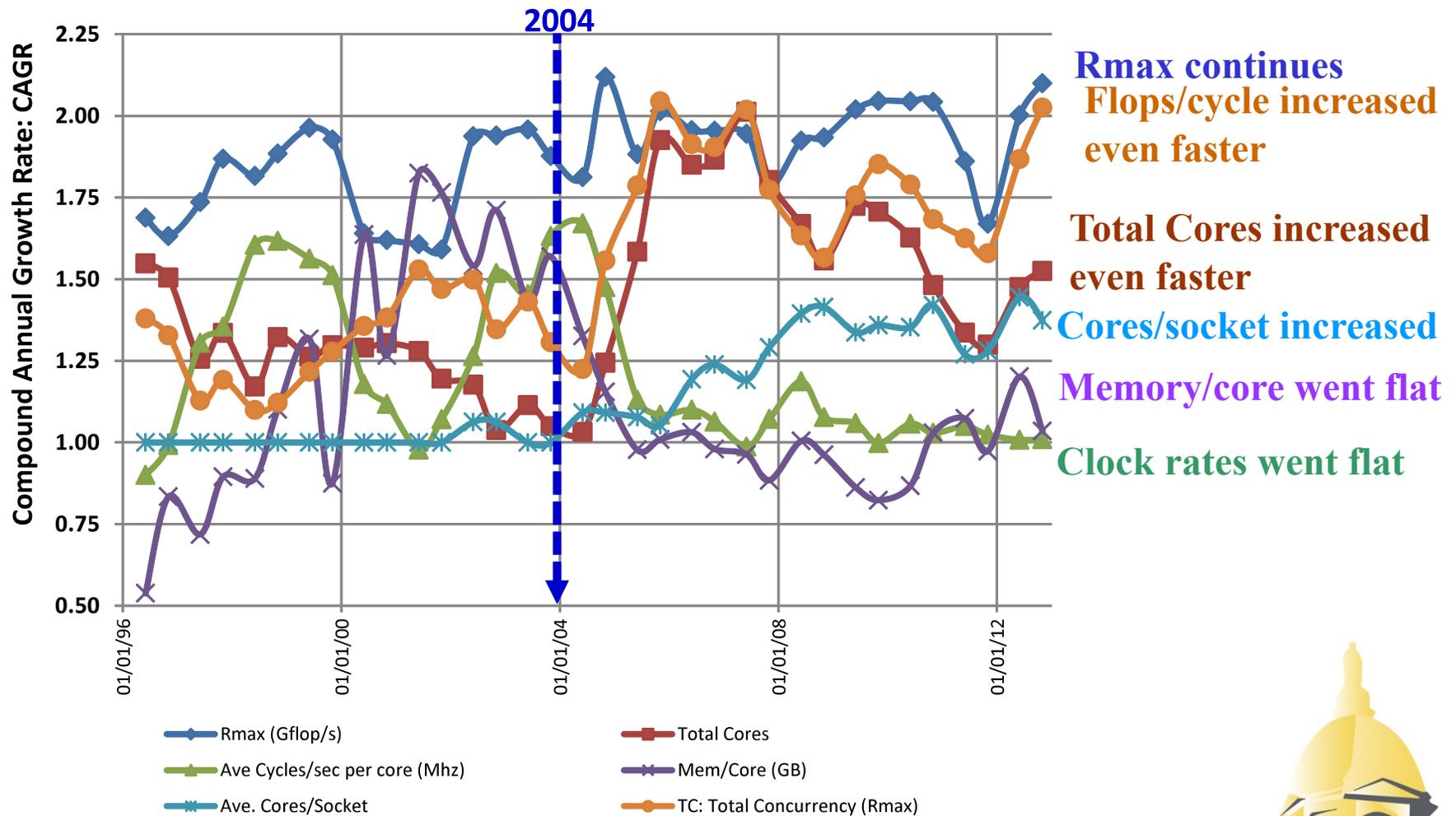
Acknowledgement: The data in this presentation was funded in part by the US Dept. of Energy, Sandia National Labs, as part of their XGC project.



We All Know The Story: Unbroken Growth in Rmax



2004: The Power Wall Changed Architecture



Key Memory Characteristics

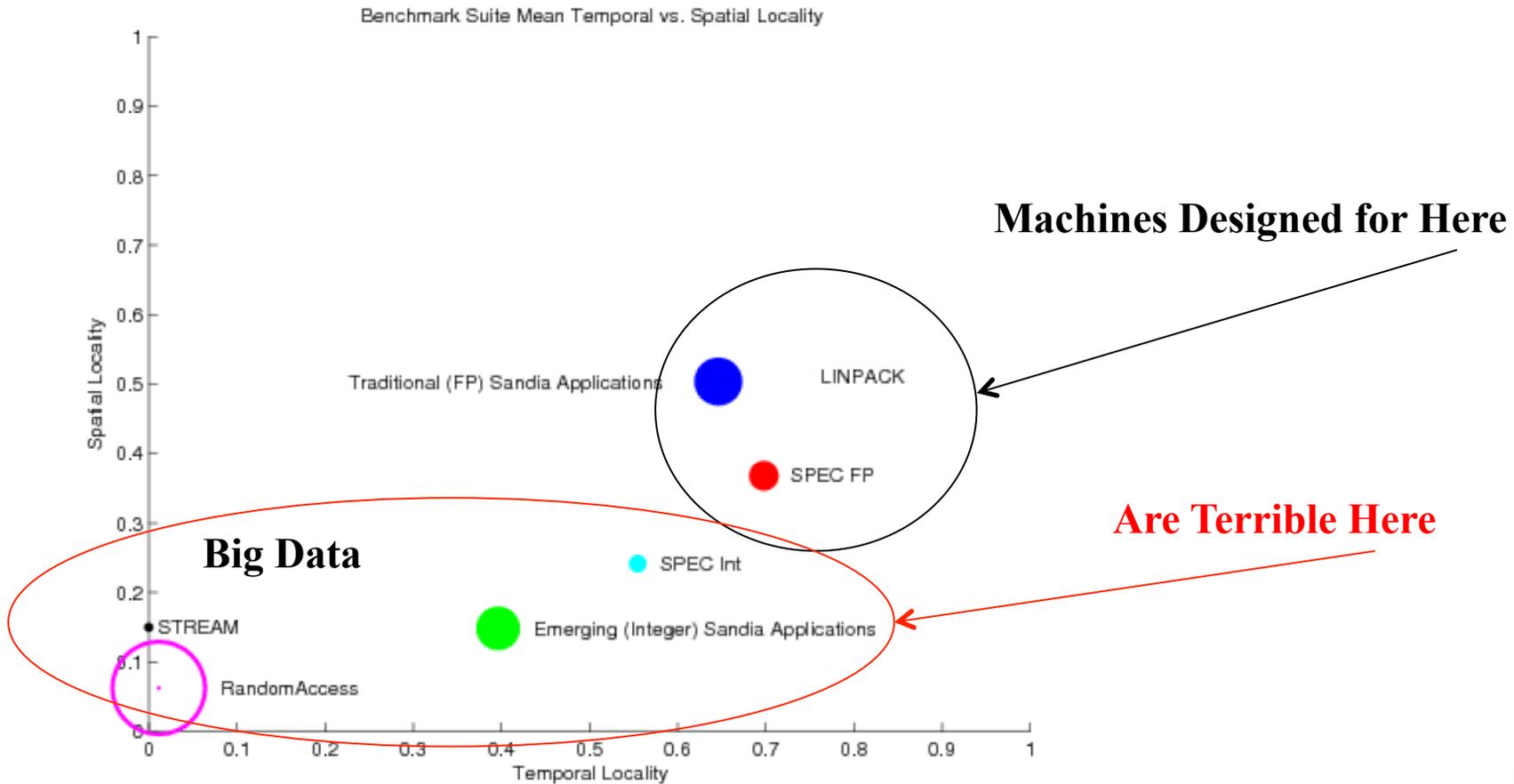
- **Capacity:** esp. per node/socket/core...
- **Bandwidth:** esp. per flop
- **Latency:** as a function of size
- **Energy:** esp. compared to computation

Looking Forward: Problems in All Areas!

**My view: Architecture must focus
on memory,
not computation**



And What Do We See in Apps?



Murphy, Kogge. On The Memory Access Patterns of Supercomputer Applications: Benchmark Selection and Its Implications, IEEE TC, 7/07



The 2008 Exascale Report

ExaScale Computing Study: Technology Challenges in Achieving Exascale Systems

Peter Kogge, Editor & Study Lead

Keren Bergman
Shekhar Borkar
Dan Campbell
William Carlson
William Dally
Monty Denneau
Paul Franzon
William Harrod
Kerry Hill
Jon Hiller
Sherman Karp
Stephen Keckler
Dean Klein
Robert Lucas
Mark Richards
Al Scarpelli
Steven Scott
Allan Snavely
Thomas Sterling
R. Stanley Williams
Katherine Yelick

September 28, 2008

This work was sponsored by DARPA IPTO in the ExaScale Computing Study with Dr. William Harrod as Program Manager; AFRL contract number FA8650-07-C-7724. This report is published in the interest of scientific and technical information exchange and its publication does not constitute the Government's approval or disapproval of its ideas or findings

NOTICE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

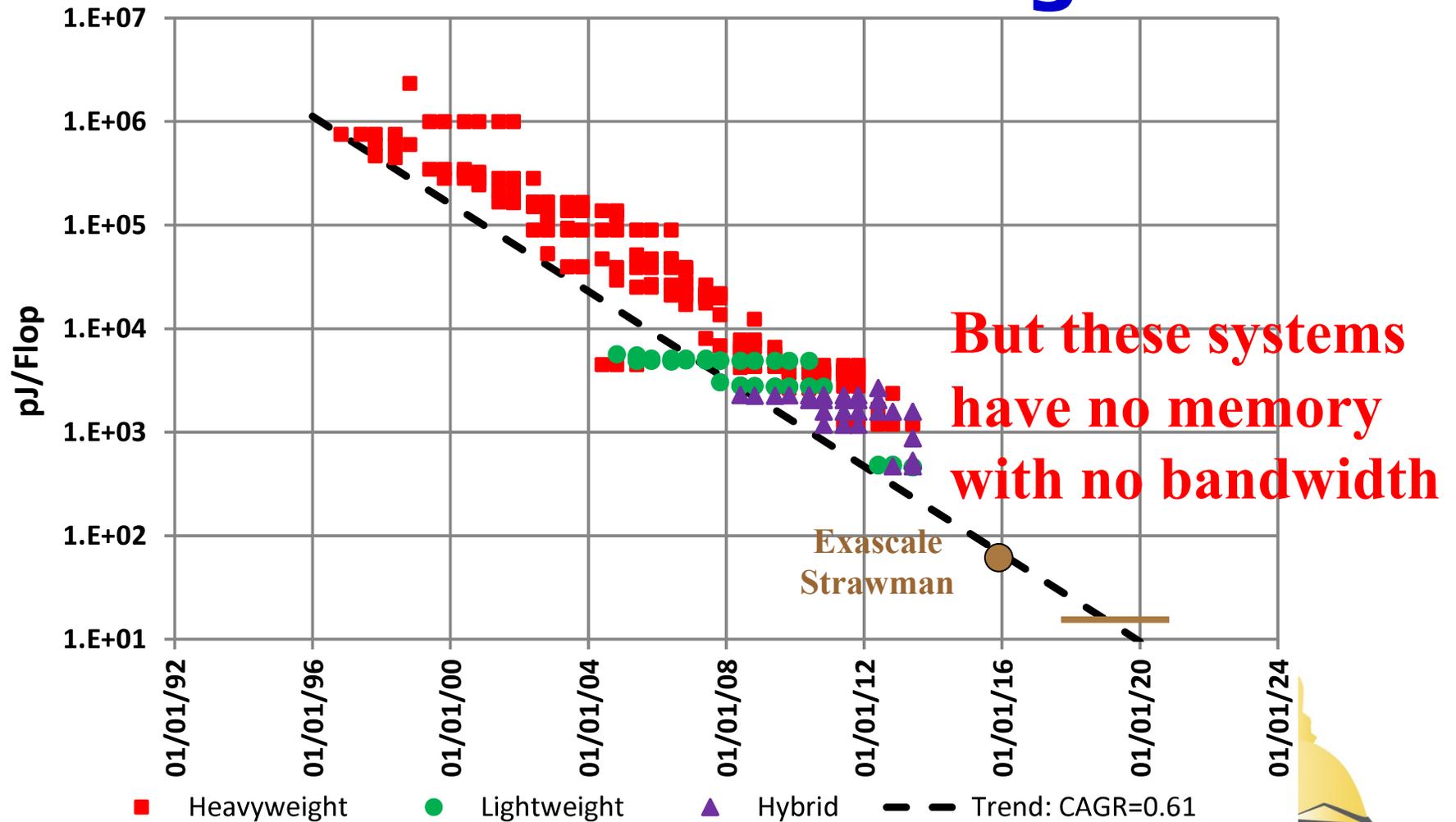
APPROVED FOR PUBLIC RELEASE, DISTRIBUTION UNLIMITED.



- Goal: "Exascale" – 1000X Petascale
 - Exa supercomputer
 - Peta rack
 - Tera embedded
- 2015 Exa supercomputer in 20MW = 20pJ/flop
- 4 problems
 - **Power/Energy**
 - **Memory**
 - Resiliency
 - Programming



Energy per Flop is Dropping: But Not Fast Enough



Topics

- Today's architectures
- Memory as a Technology
- Why is memory a growing problem
- The first attempt at alternative architectures: **Processing In Memory**
- The emerging future: **Processing Near Memory**



Memory in Today's Architectures



UNIVERSITY OF
NOTRE DAME

ATPESC July 29, 2013

*ENABLING
INNOVATION*



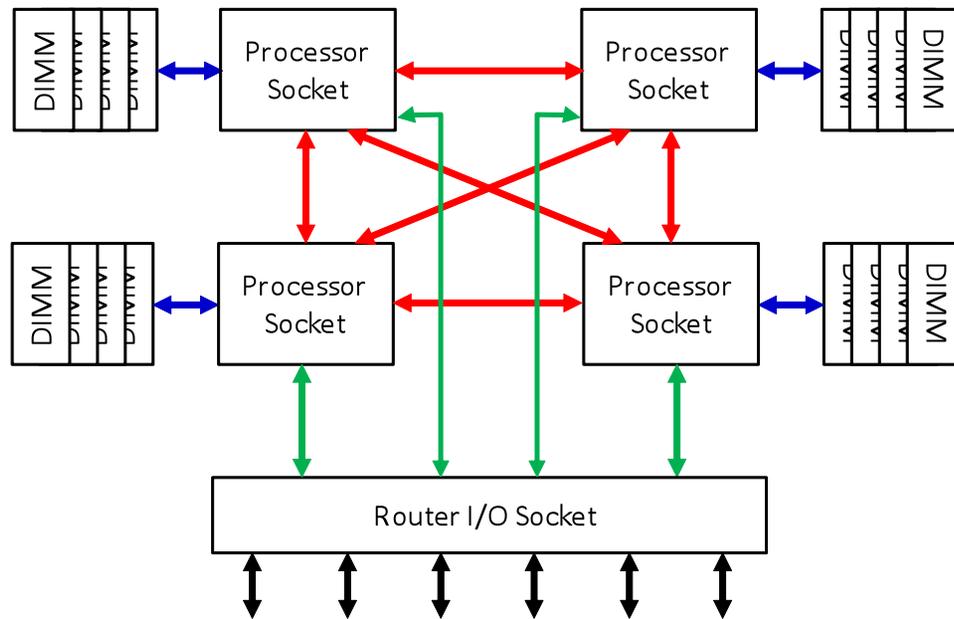
10

Today's Architecture Classes

- **Heavyweight:** traditional 100+W multi-core
 - Often requires supporting chip set
- **Lightweight:** lower power single chip system
 - Lower performance but denser packaging
- **Hybrid/Heterogeneous:** Heavyweight/GPU combination, with radically different ISAs
- **Big/Little:** Multi-core, same ISA, but different core sizes
- **But wait!** There's more when we try for very large shared memory
 - And more on the way



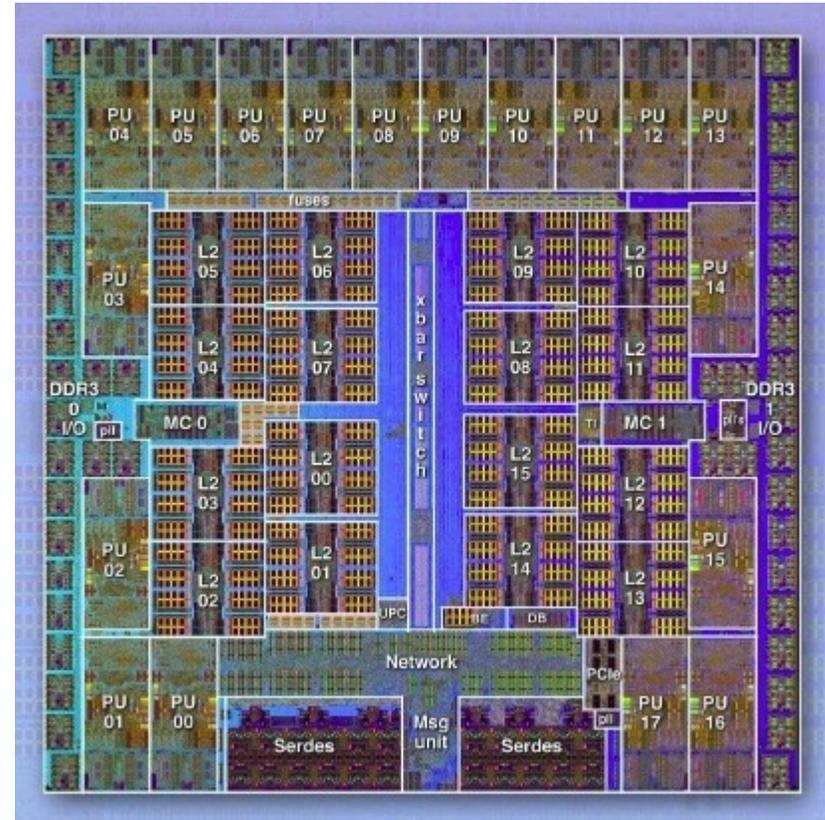
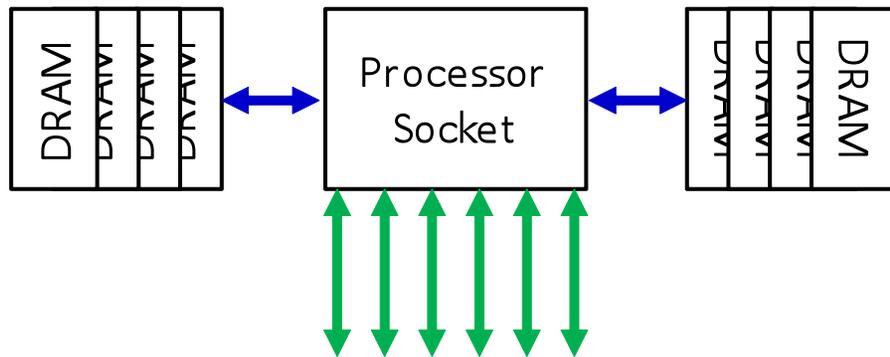
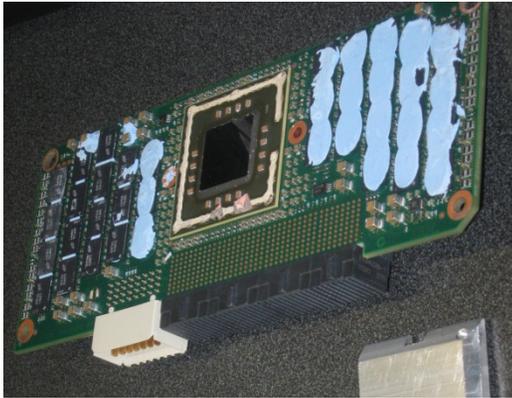
Today's Heavyweight Blade



A Power 7 Drawer



Lightweight: Eg. BlueGene/Q



Integrated

- NIC
- Memory controllers

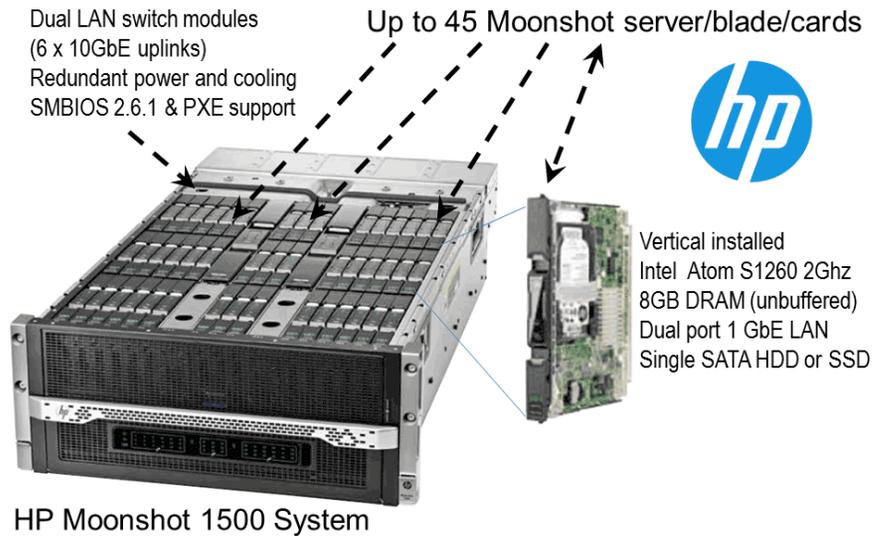
<http://www.heise.de/newsticker/meldung/SC-2010-IBM-zeigt-BlueGene-Q-mit-17-Kernen-1138226.html>



Other Lightweight Systems Emerging



Calxeda quad-socket, quad-core ARMs

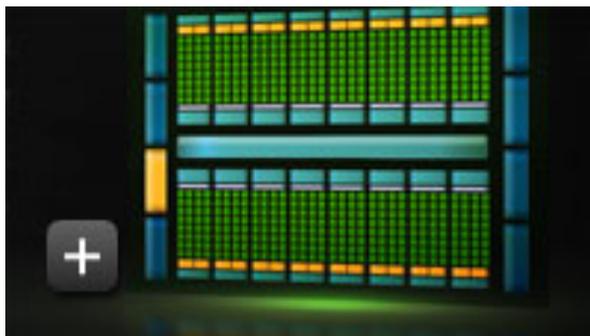


HP Moonshot

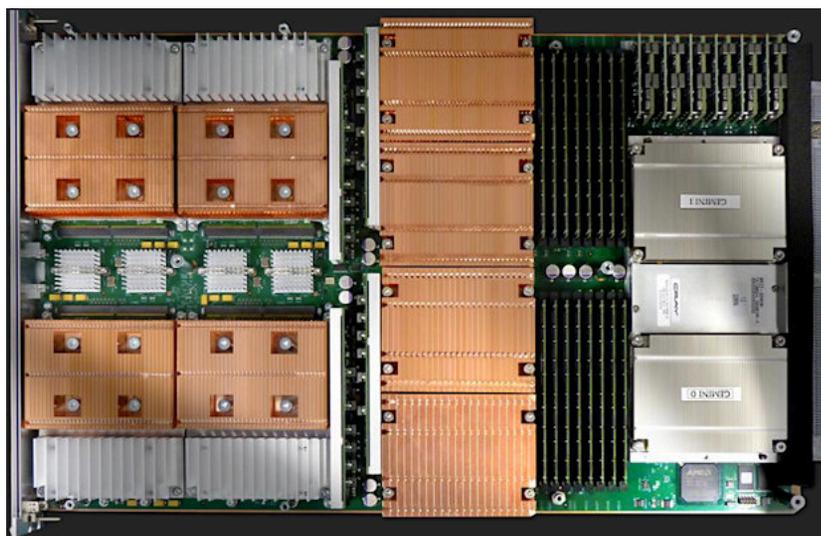
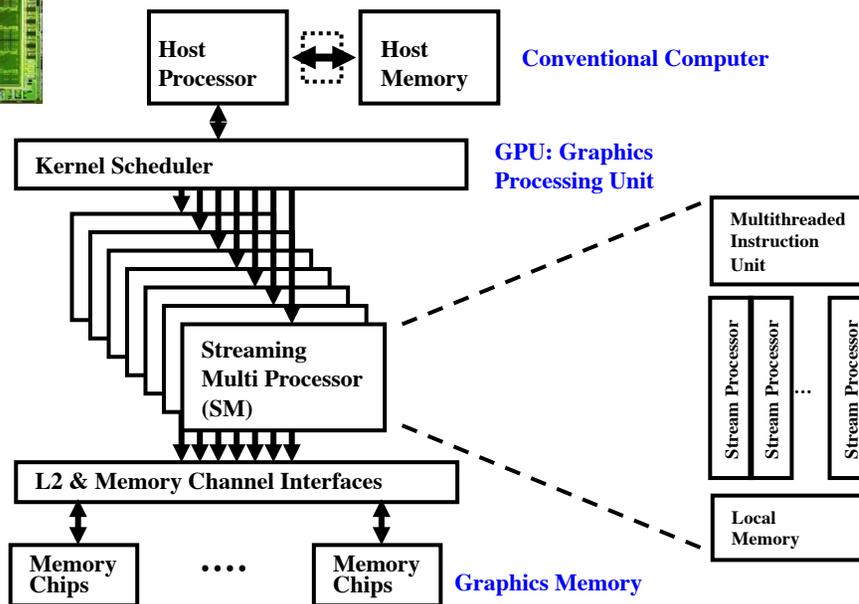
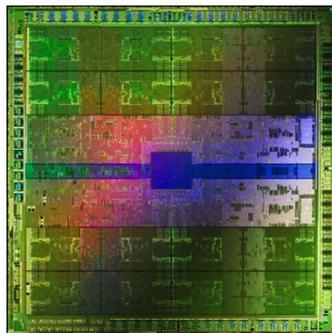


Heterogeneous Architectures

- Mix of heavyweight masters and GPU compute engines



http://www.nvidia.com/object/fermi_architecture.html



A Titan Blade



Big Little Architectures

- Heterogeneous multi-core with same ISA
- “Bigger” cores have higher performance (more instructions per second)
 - But are less energy efficient
- “Littler” cores have less performance
 - But are much more energy efficient
- Ability to move program states from core to core
- Examples:
 - ARM Cortex-A15 and A7, A53 and A57
 - Intel Xeon and Xeon PHI  **#1 in June 2013**

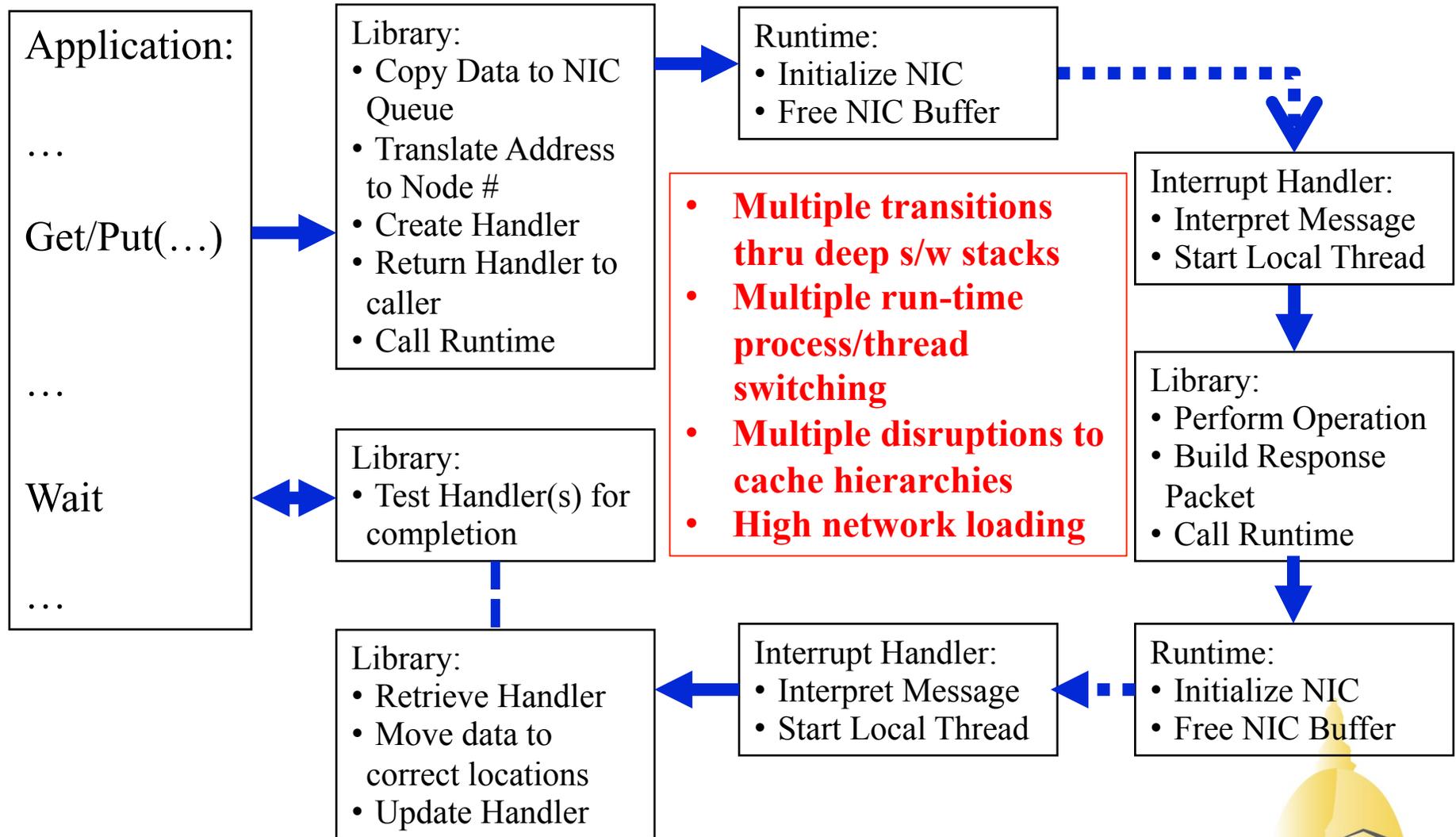


Memory In Any of These

- On the end of a **memory channel** and NOT *on* the processor chip
- At most 2-4 such channels per socket
 - Limited by off-chip pins
- At most 4 sockets sharing memory over specialized interfaces before complexity too great
- Energy of access/transport becoming dominate
- Increasingly deep cache structures on processor socket
 - With complex rules for coherency/consistency
 - And very complex protocols for “atomic” operations
 - And punt to software when non-local access



Accessing Remote Memory Today

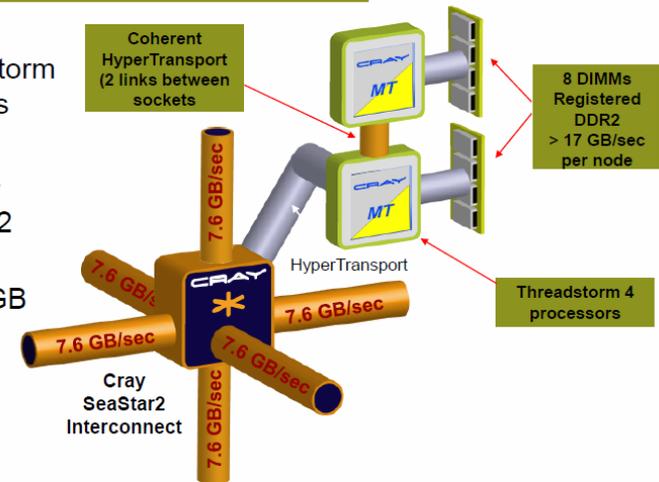


Cray MTA (and Follow-ons)

- Heavily multi-threaded cores
 - With fast thread create/switch
- True PGAS memory
 - With non-local load/store detected/managed/routed by hardware
- 2 tag bits per memory word
 - Full/empty
 - extended
- Extended load/store semantics to interact with full/empty words

Cray uRiKA High Density Node

- 4 Threadstorm processors
- 4 memory controllers
- 4 SeaStar2 NICs
- Up to 64 GB per node

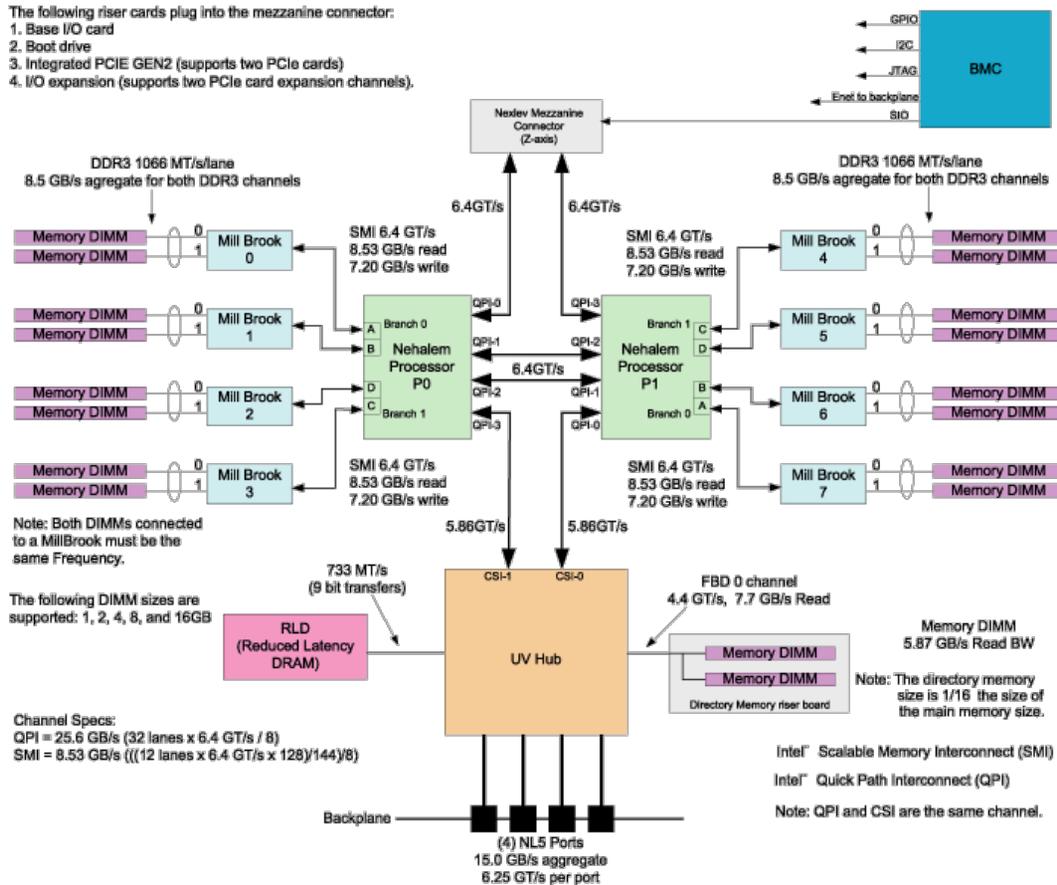


http://www.adms-conf.org/uRiKA_ADMS_keynote.pdf

- All non-local memory "equally remote"
- Relatively less dense memory (6TB/rack)
- Atomics still require interaction with remote host

SGI UV 2000 cc NUMA

The following riser cards plug into the mezzanine connector:
 1. Base I/O card
 2. Boot drive
 3. Integrated PCIE GEN2 (supports two PCIe cards)
 4. I/O expansion (supports two PCIe card expansion channels).



http://techpubs.sgi.com/library/tpl/cgi-bin/getdoc.cgi/linux/bks/SGI_Developer/books/LX_86_AppTune/sgi_html/ch05.html

- Each blade
 - 2 8-core sockets
 - Up to 128GB
 - **Separate Hub chip**
- 32 blades/rack
 - 512 cores
 - **only 4TB/rack**
- +: Cache-coherent shared memory
 - **But via complex off-chip directories**
- Up to 64TB
- **Limited atomics**

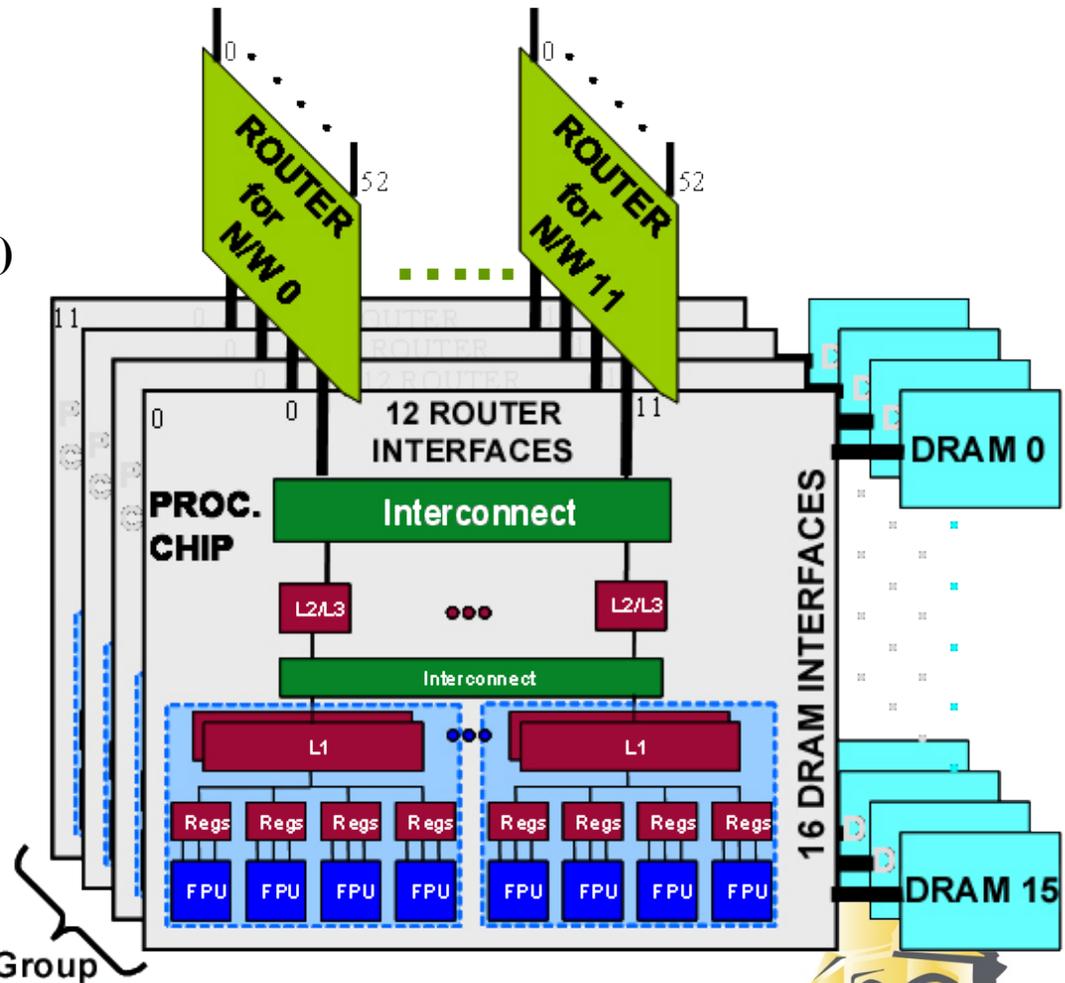


2008 Exascale Report Strawman

Sizing done by “balancing” power budgets with achievable capabilities

- 4 FPUs+RegFiles/Core (=6 GF @1.5GHz)
- **1 Chip = 742 Cores** (=4.5TF/s)
 - 213MB of L1I&D; 93MB of L2
- 1 Node = 1 Proc Chip + 16 DRAMs (16GB)
- 1 Group = 12 Nodes + 12 Routers (=54TF/s)
- 1 Rack = 32 Groups (=1.7 PF/s)
 - 384 nodes / rack
- 3.6EB of Disk Storage included
- 1 System = 583 Racks (=1 EF/s)
 - **166 MILLION cores**
 - 680 MILLION FPUs
 - 3.6PB = 0.0036 bytes/flops
 - **68 MW** w'aggressive assumptions

Interconnect for intra and extra Cabinet Links



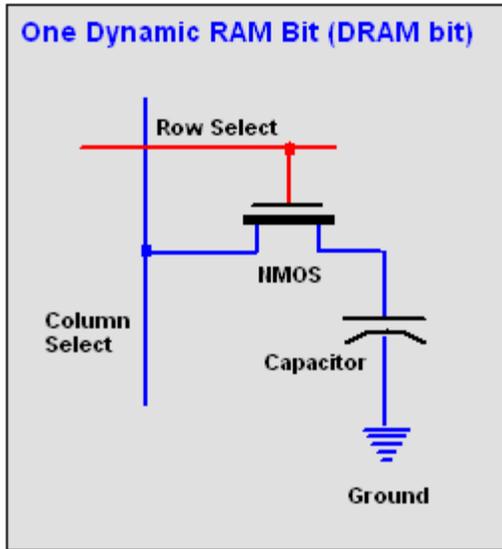
Largely due to Bill Dally, Stanford

Memory: The Technology

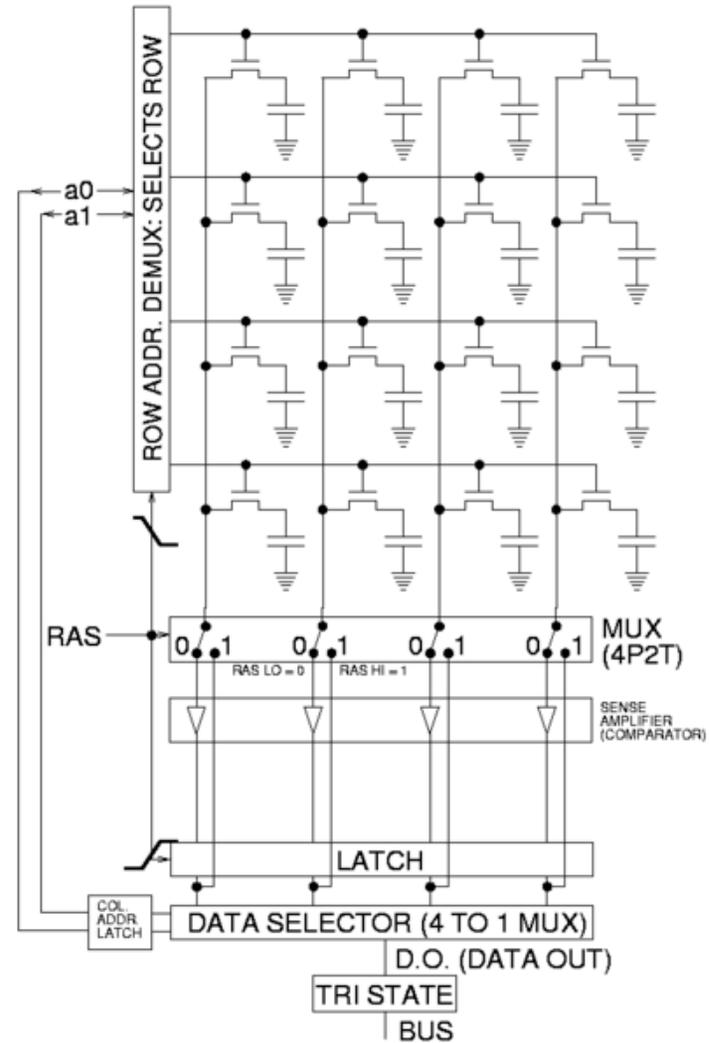


Basic DRAM

From Computer Desktop Encyclopedia
© 2005 The Computer Language Co. Inc.



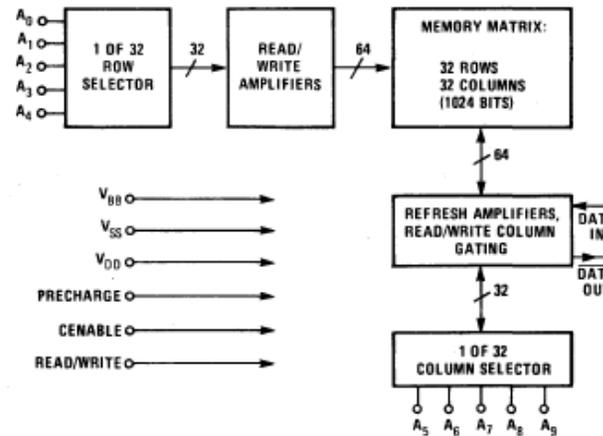
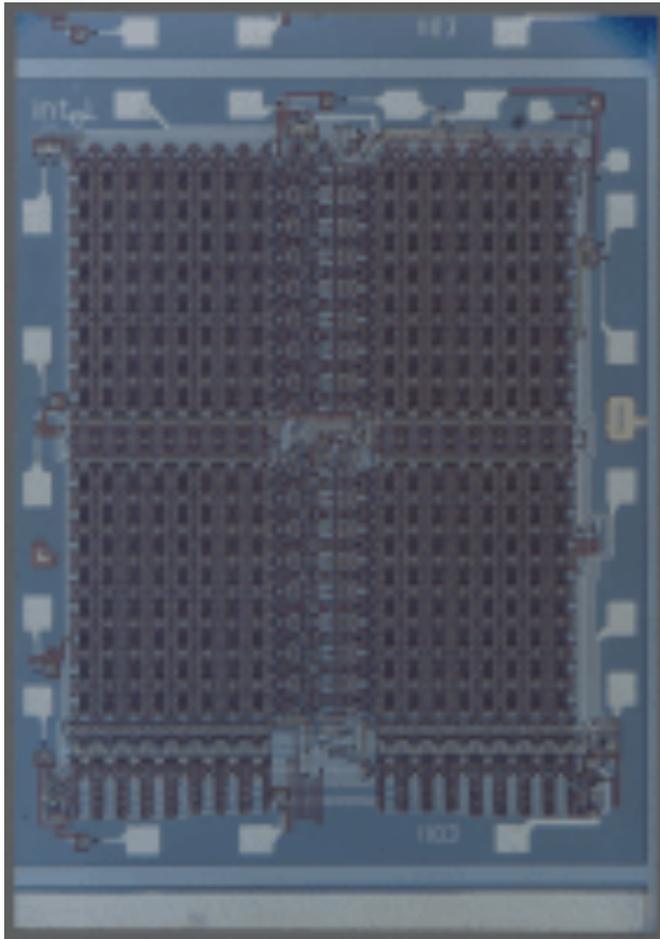
<http://encyclopedia2.thefreedictionary.com/dynamic+RAM>



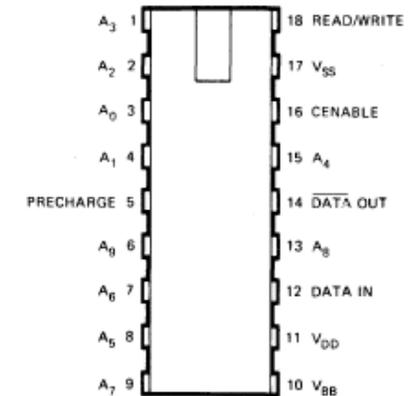
http://en.wikipedia.org/wiki/File:Square_array_of_mosfet_cells_read.png



Intel 1103: Splitting the Address



a. Block Diagram

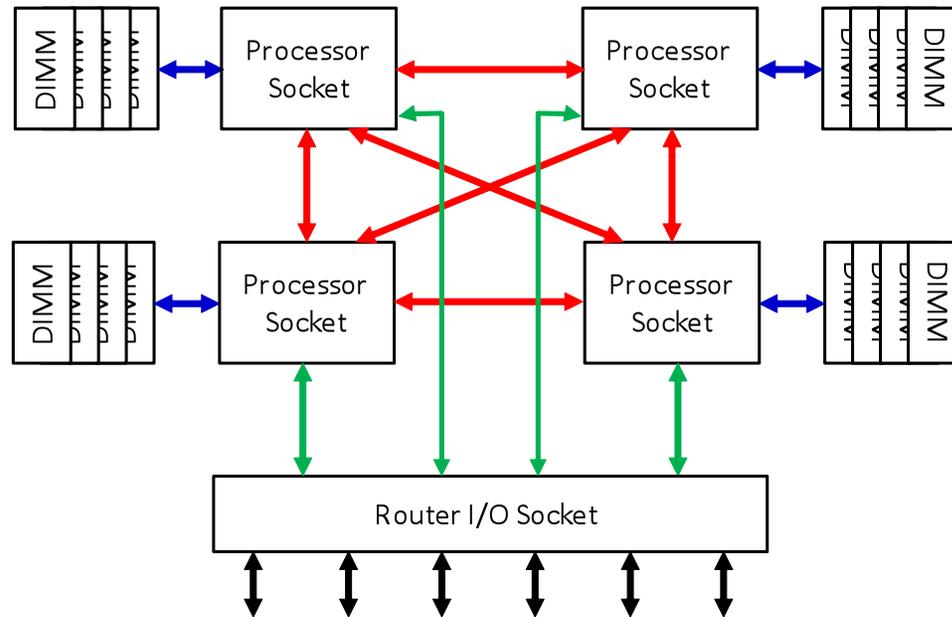


b. Pin Connections

- 1k x 1bit part from outside
- But 10b address split in 2
 - 5b **Row Address**: which of 32 32b words
 - 5b **Column Address**: which bit of that word



“Please Sir, I want more” Multiple Sockets with Coherent Shared Memory

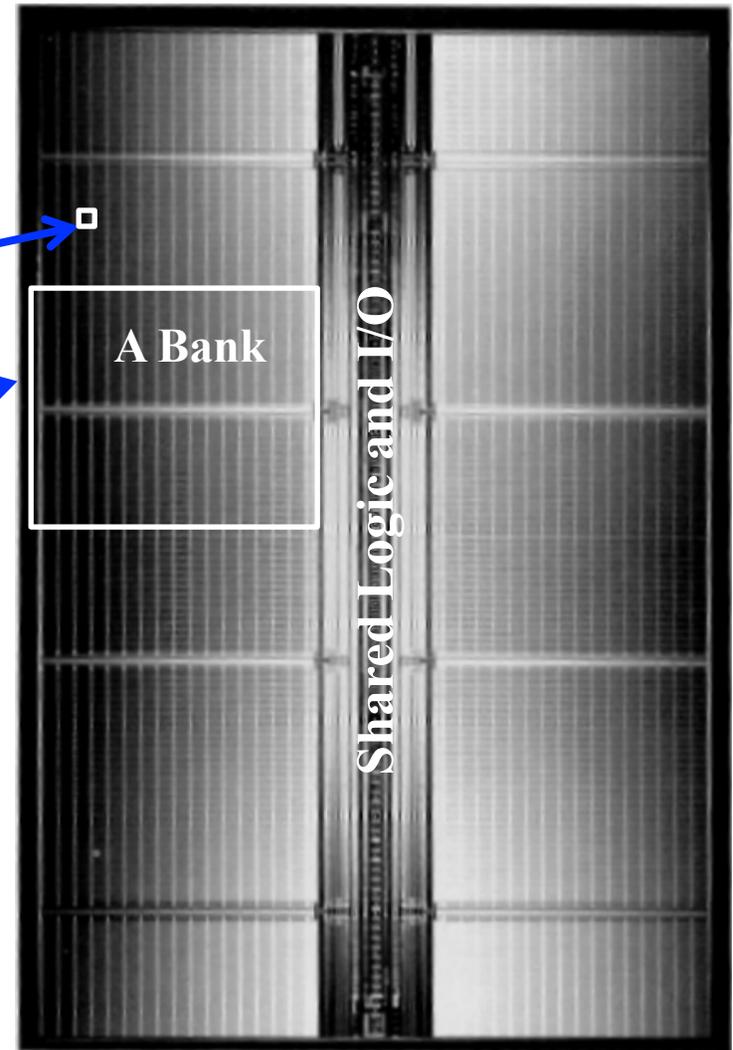


- Now addresses must be sorted by **socket**
 - before they are routed to correct socket
 - before they are routed to correct channel
 - before they are processed by memory controllers

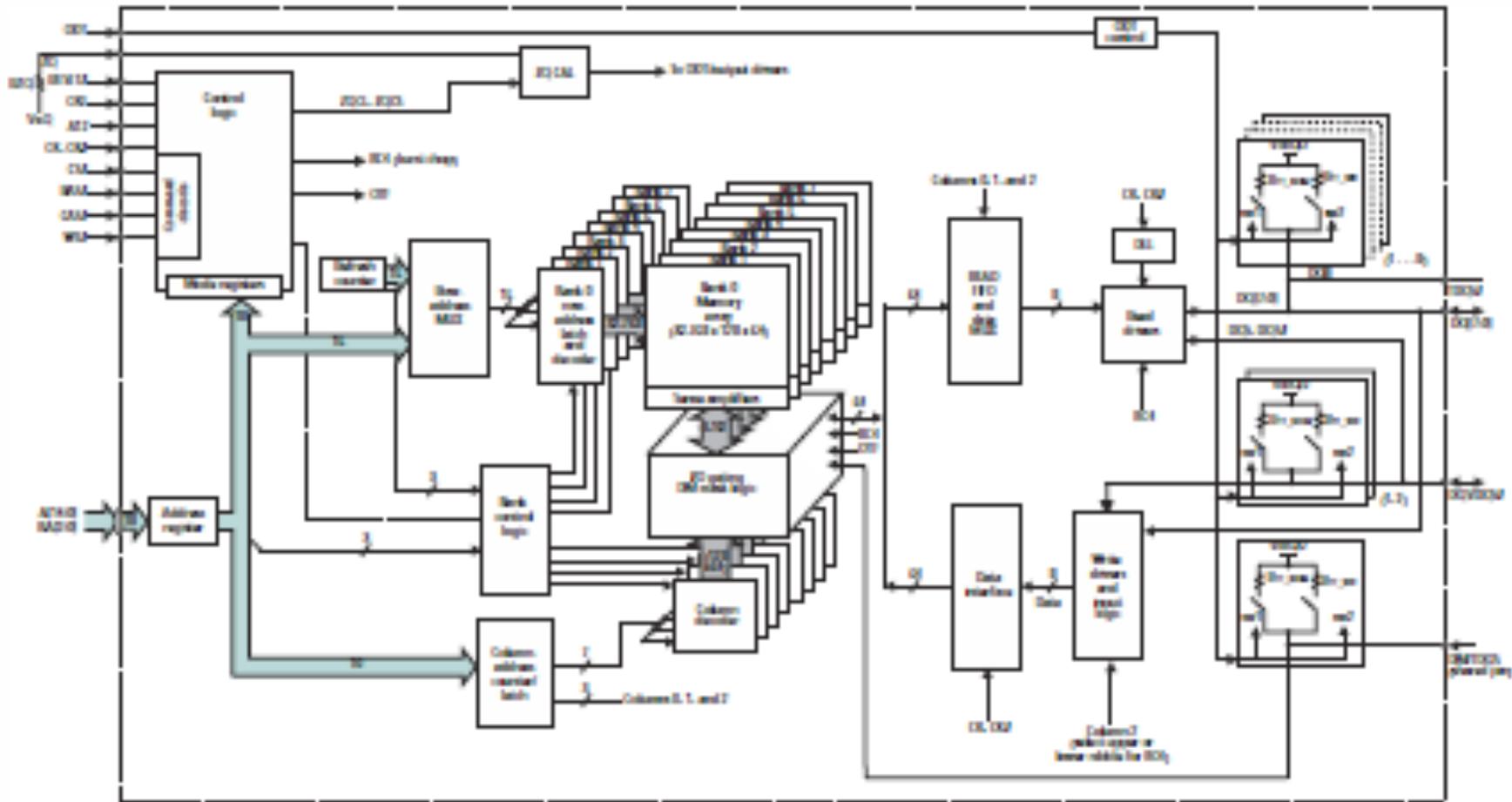


A Bigger Die

- Cannot organize Gb chips as 1G rows by 1b
- Must break into "**Blocks**"
 - Typically $\sim 1\text{Kb} \times 1\text{Kb}$
- Arrange blocks into "**Banks**"
- Address now:
 - Which bank
 - Which block in bank
 - Which row in block
 - Which bits in row

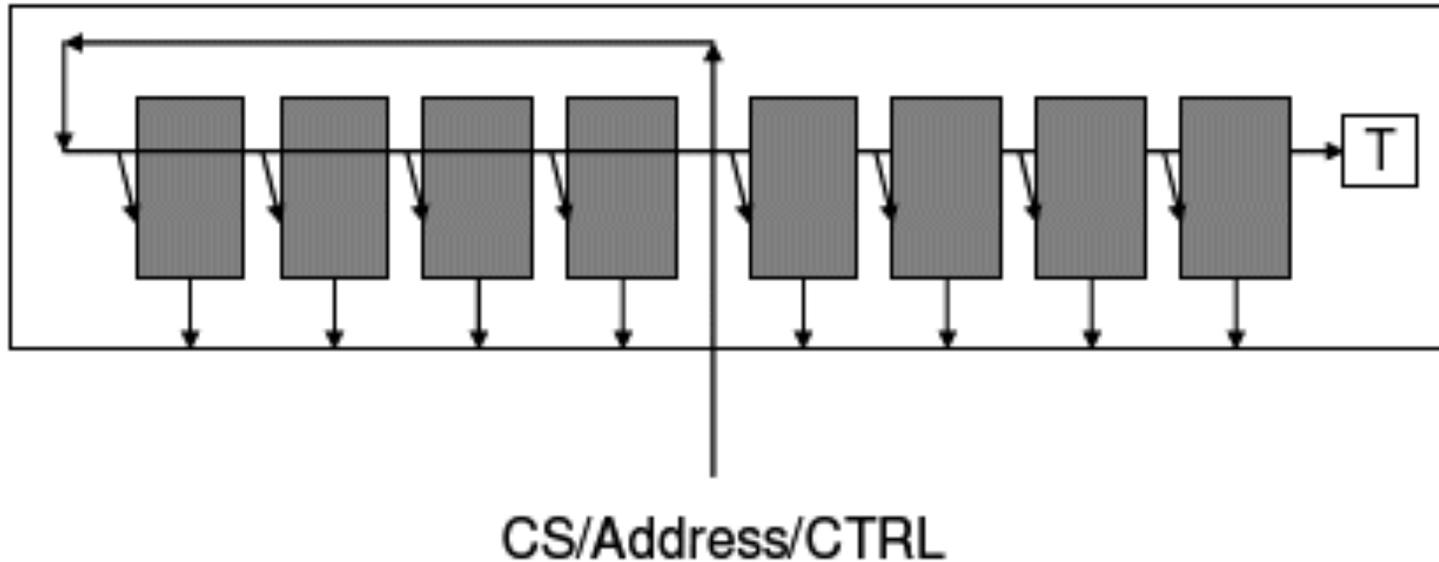


But Now We Can Run Banks "Concurrently"



from Micron MT41J256M8 32M x b x 8 Banks

A "Simple" DIMM

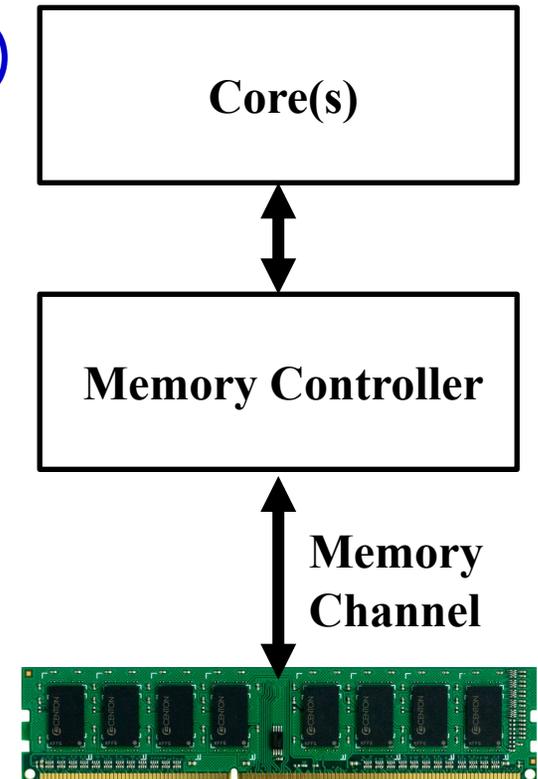


- All chips get same address/command
- Each chip contributes its 4 or 8 bits to data bus
- Interface speed rated in "Transfers/sec"
- DIMM "looks like" 8 concurrent banks of 64b

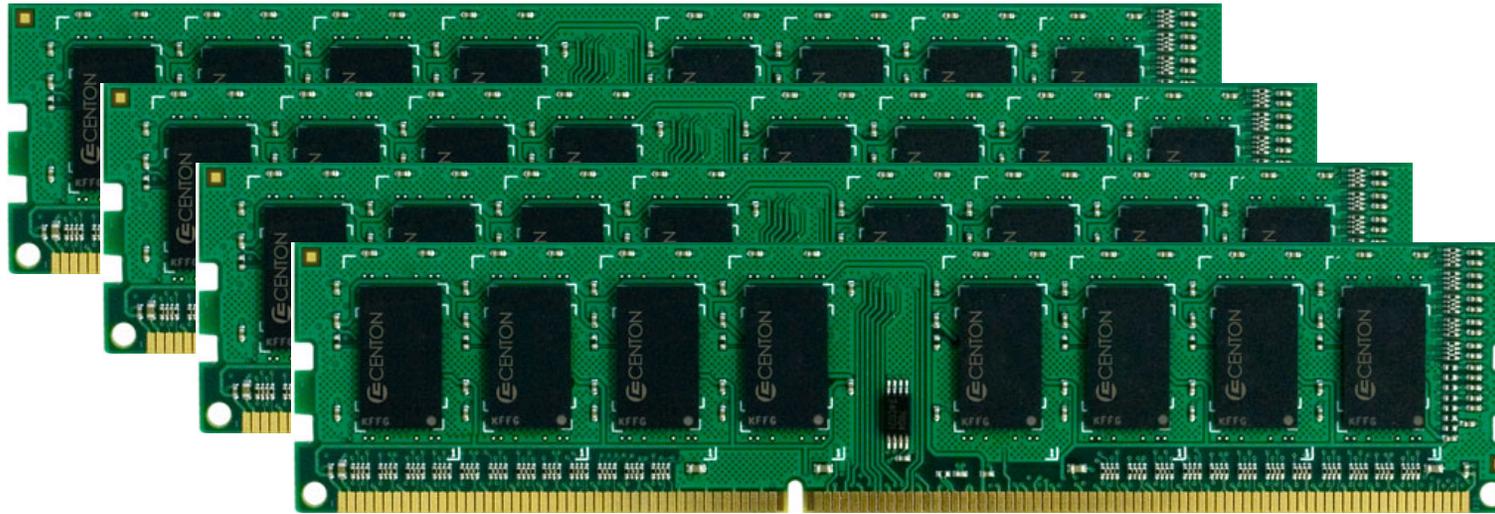


What Does the Memory Controller Do?

- Stream of addresses from core(s)
- Sort by bank number
- Within same bank, sort by row #
- For same row of same bank:
 - Issue initial row read request
 - Issue word reads and writes to that row
 - Close row when done to refresh memory
- Remember, sets for other banks can be executed concurrently
 - Sequentially interleaved over single common memory channel



“Please Sir, I want more”



- All share same wires to microprocessor
- But can only talk to 1 DIMM at a time
- Add “DIMM #” to address – called “**Rank**”
- Now Memory Controller must sort by rank also
- Capacitive loading from all DIMMs slows transfers



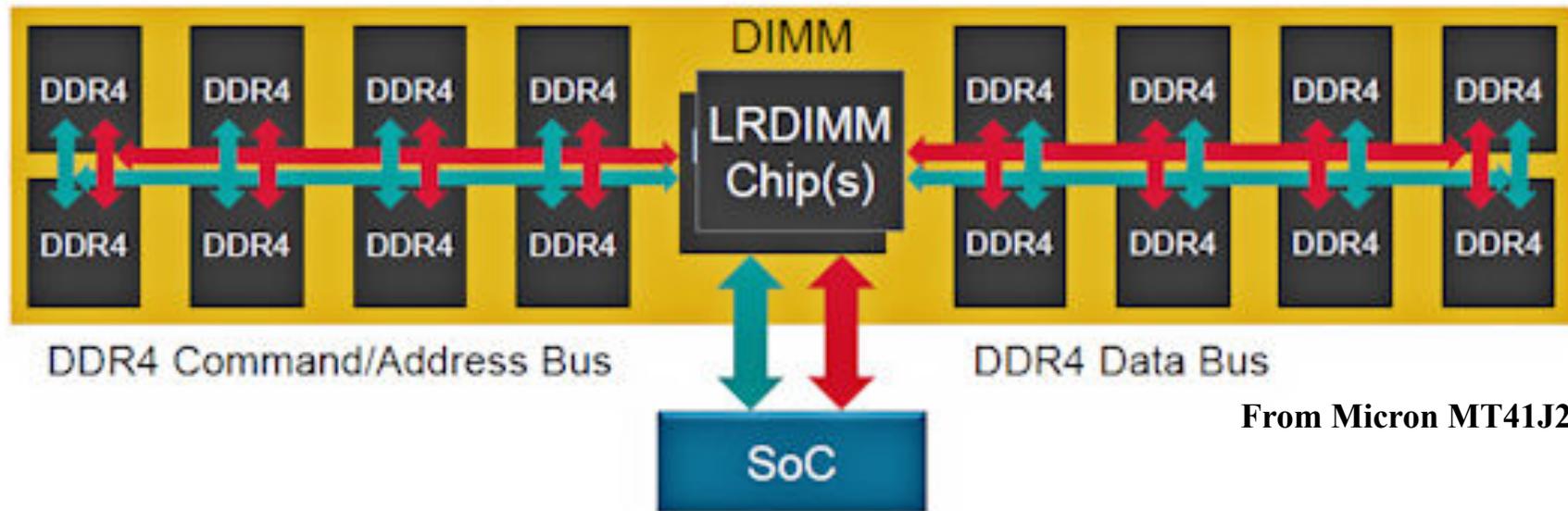
“Please Sir, I want more”



- Put multiple ranks on same DIMM
- Still can only talk to 1 rank at a time
- Electrical loading problem continues
 - Each rank still loading same bus
 - Even worse with multiple multi-rank DIMMs



“Please Sir, I want more” Load Reduced DIMMs

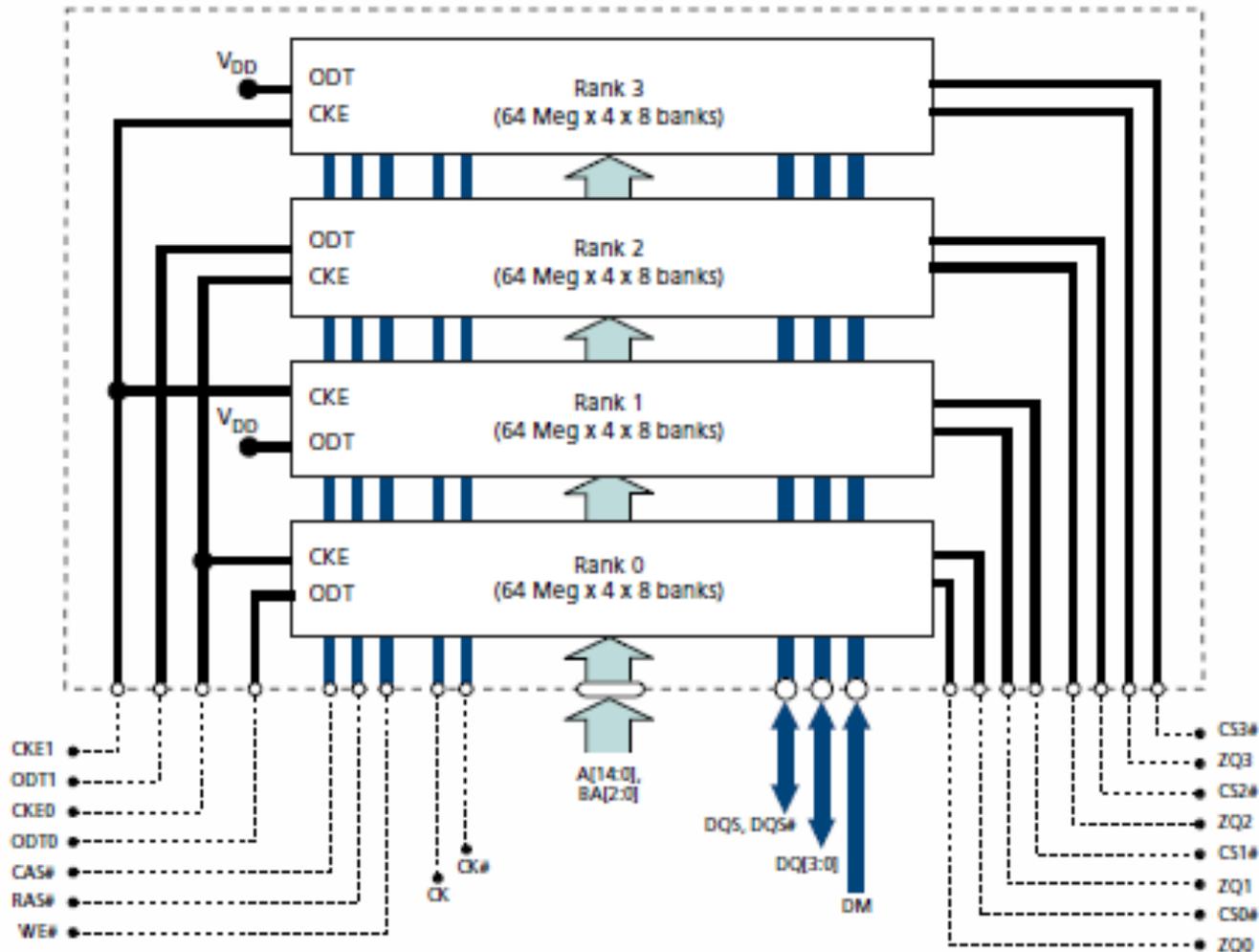


From Micron MT41J2G4

- Helps improve electrical transfer speeds
- But still deal with multiple ranks, banks, blocks, rows
- And typically increase latency



“Please Sir, I want more” Multiple Ranks Move “On Die”

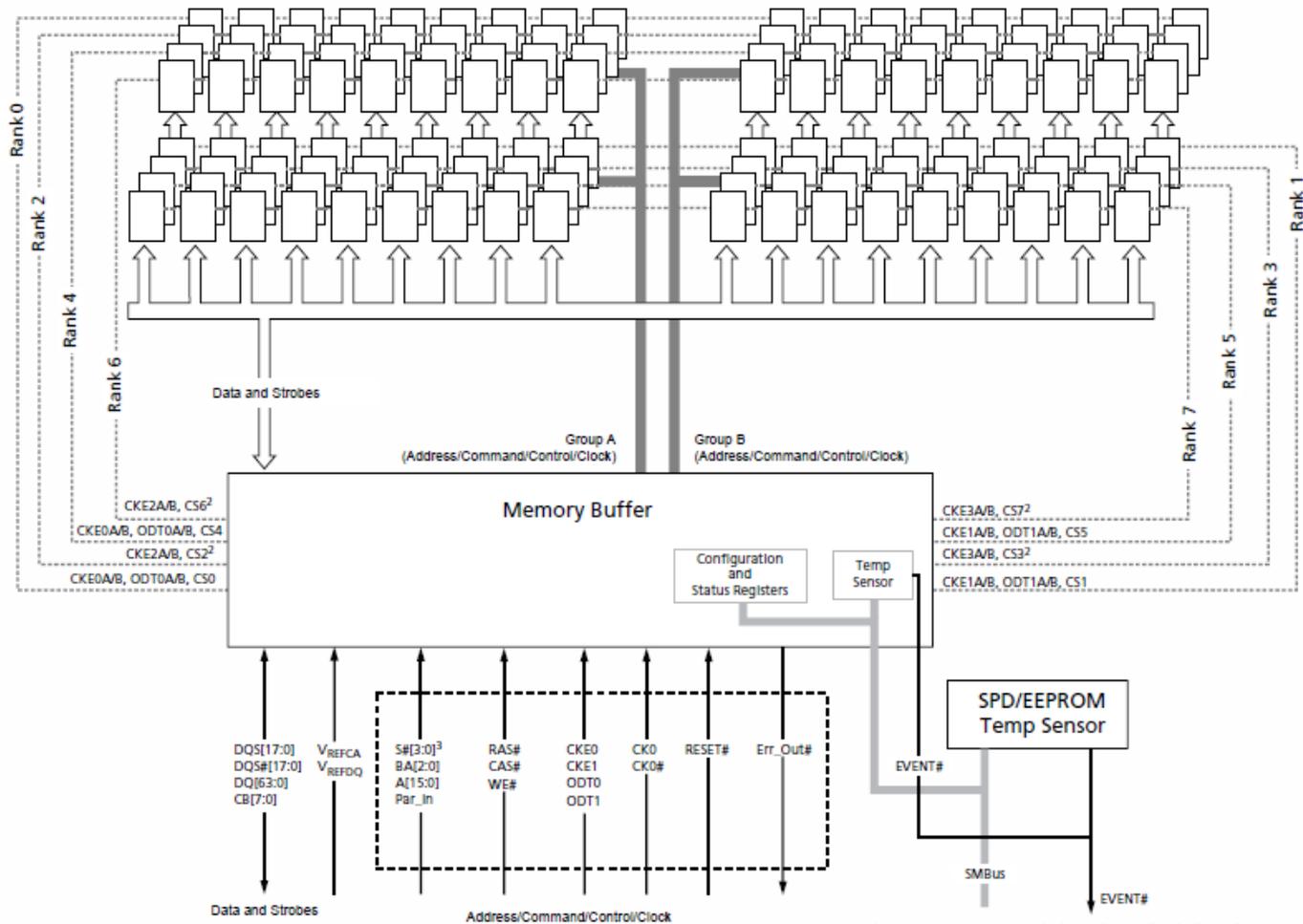


From Micron MT41J2G4



Towards a Single DIMM Per Channel

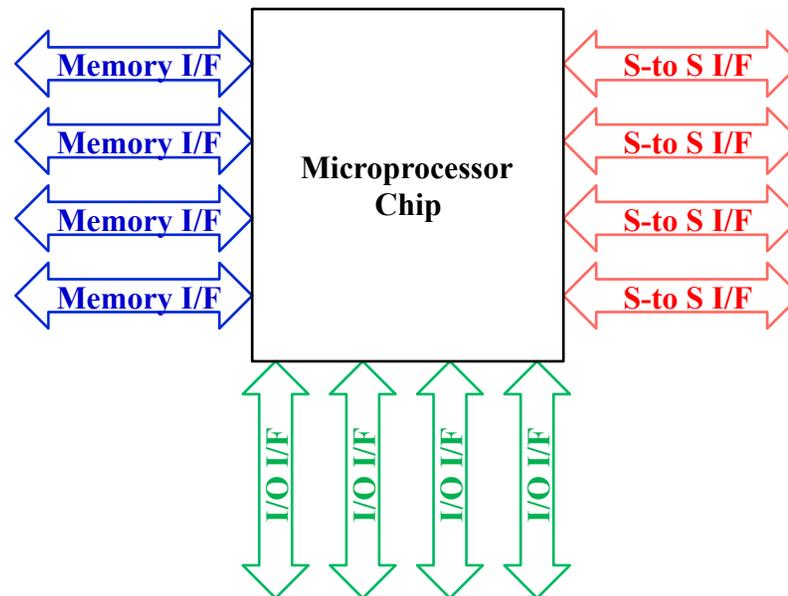
36 QuadDie DRAM, in 8 Module Rank Configuration



From Micron MT144KSZQ4G72LZ 32GB LRDIMM



“Please Sir, I want more” Multiple Memory Channels/Socket



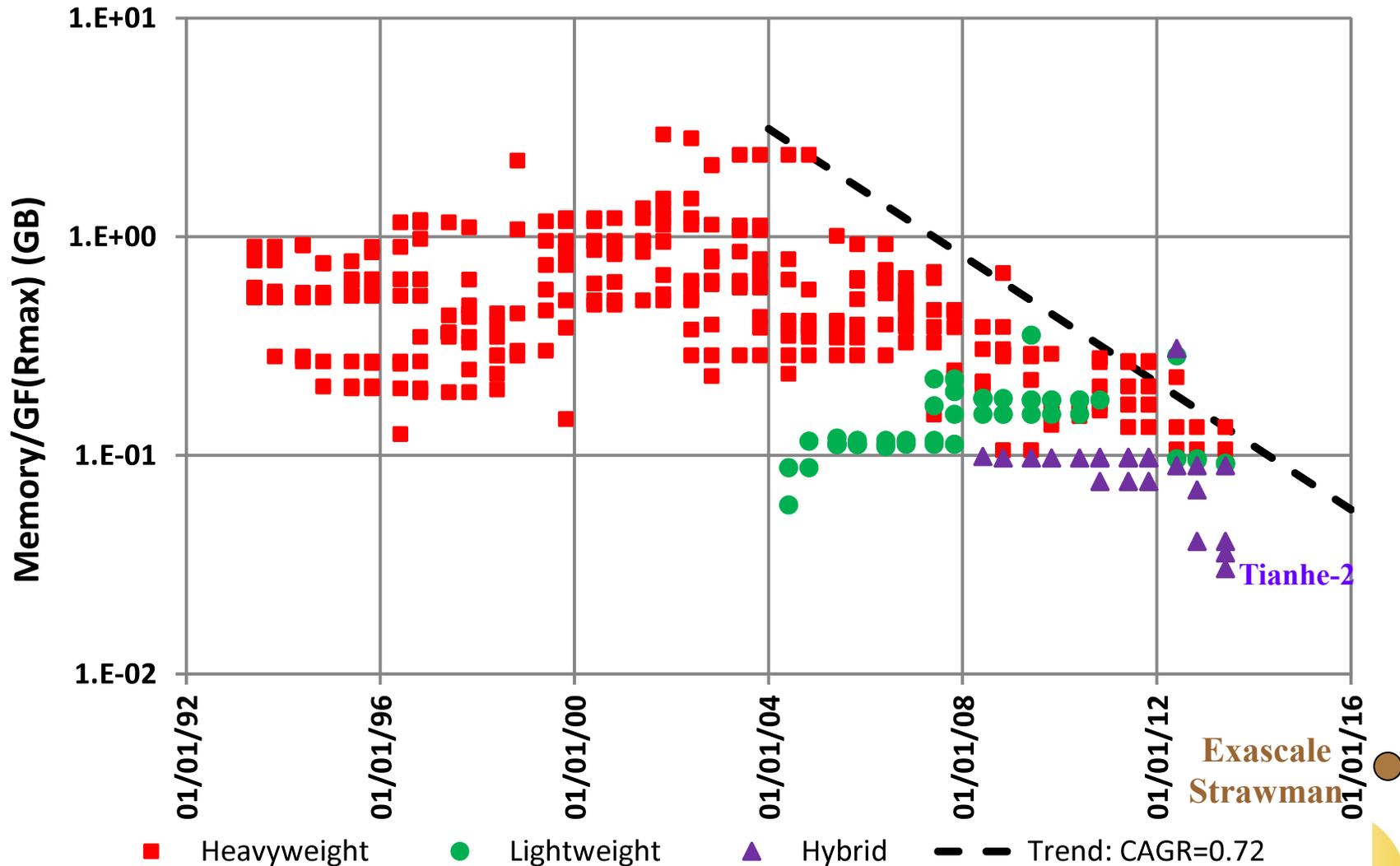
- Multiple cores on socket all contribute to address streams
- Now addresses must be sorted by **channel** before they are processed by memory controllers



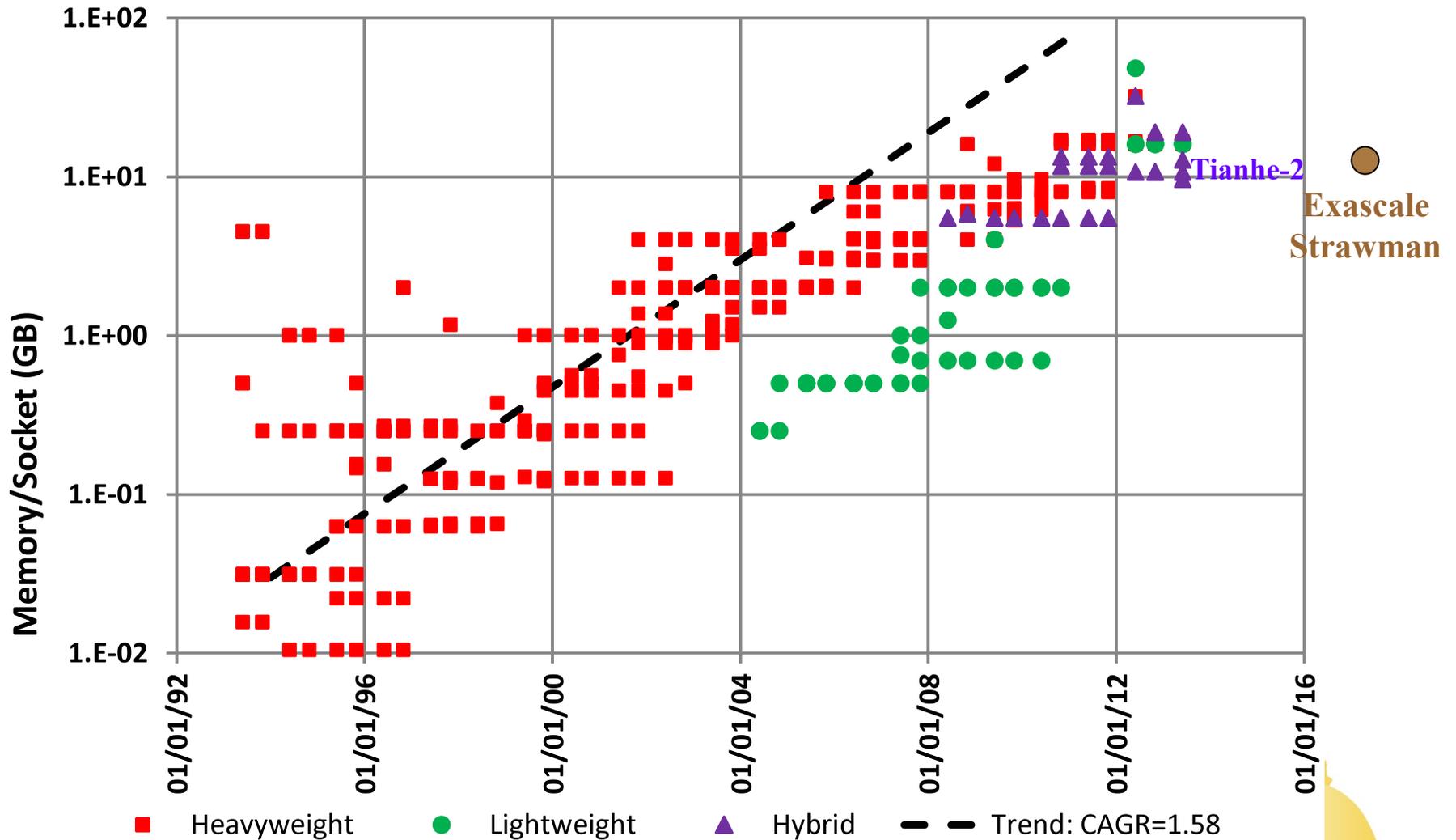
Memory: The Growing Problem



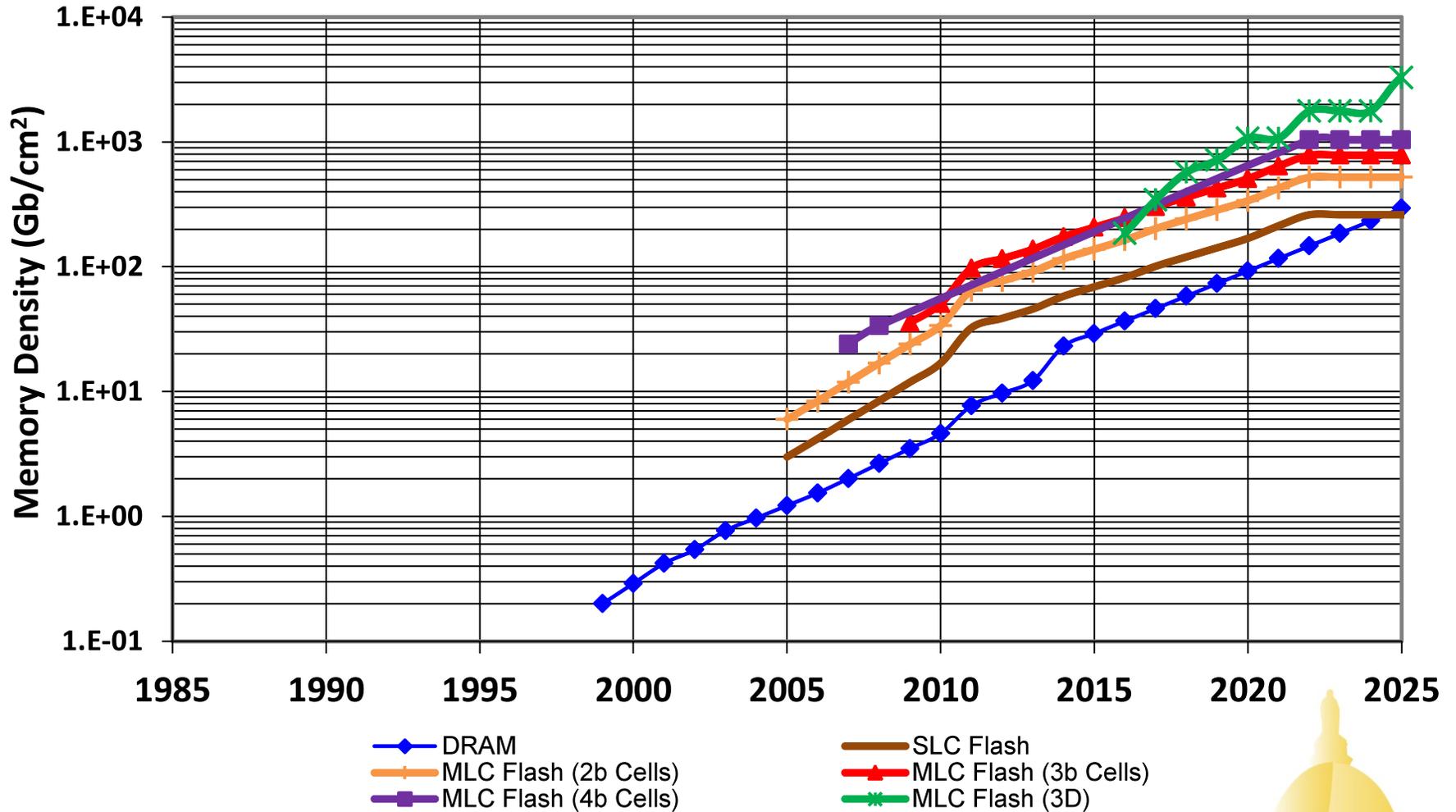
The Traditional Rule of Thumb



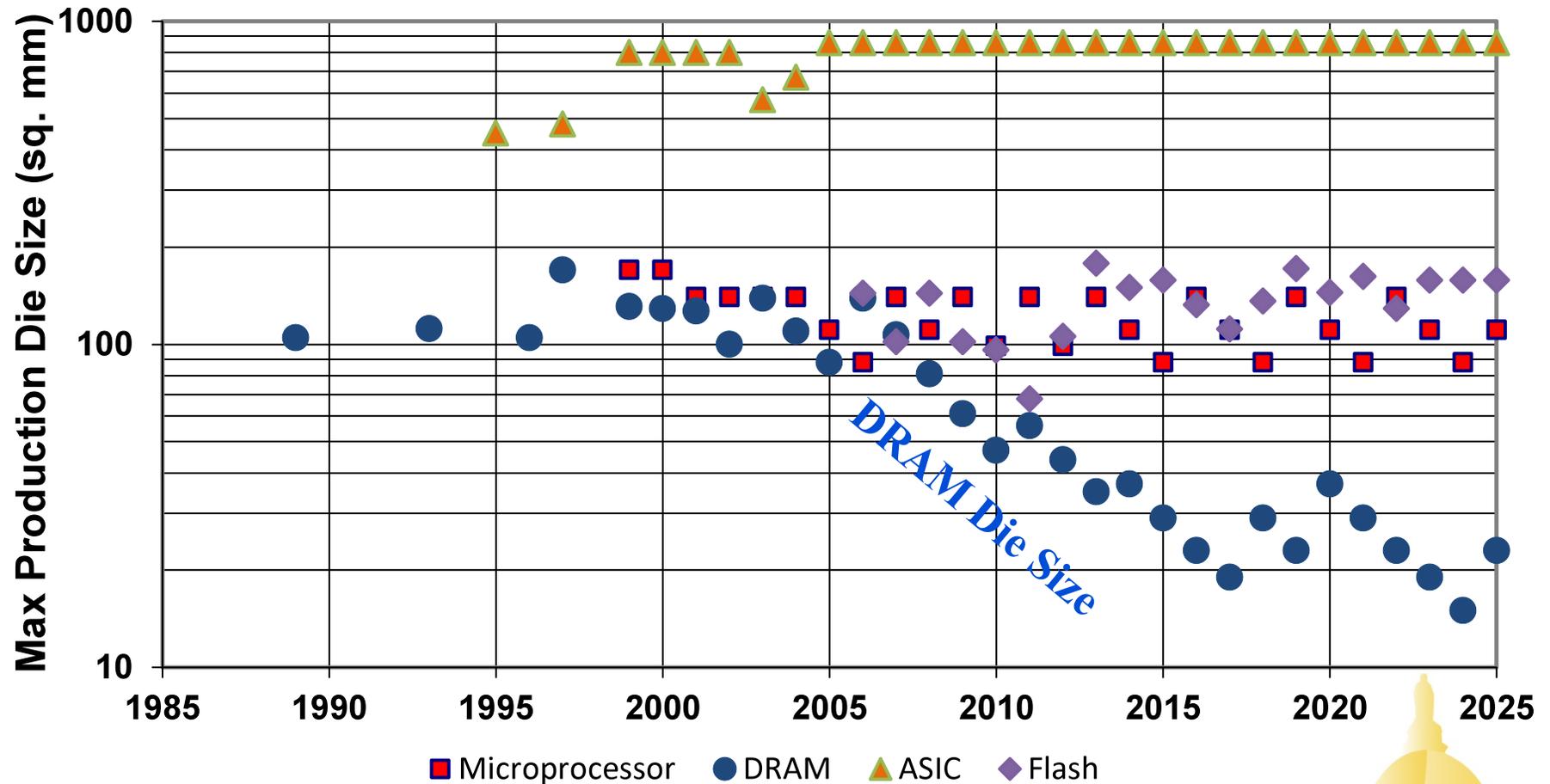
Capacity per Socket



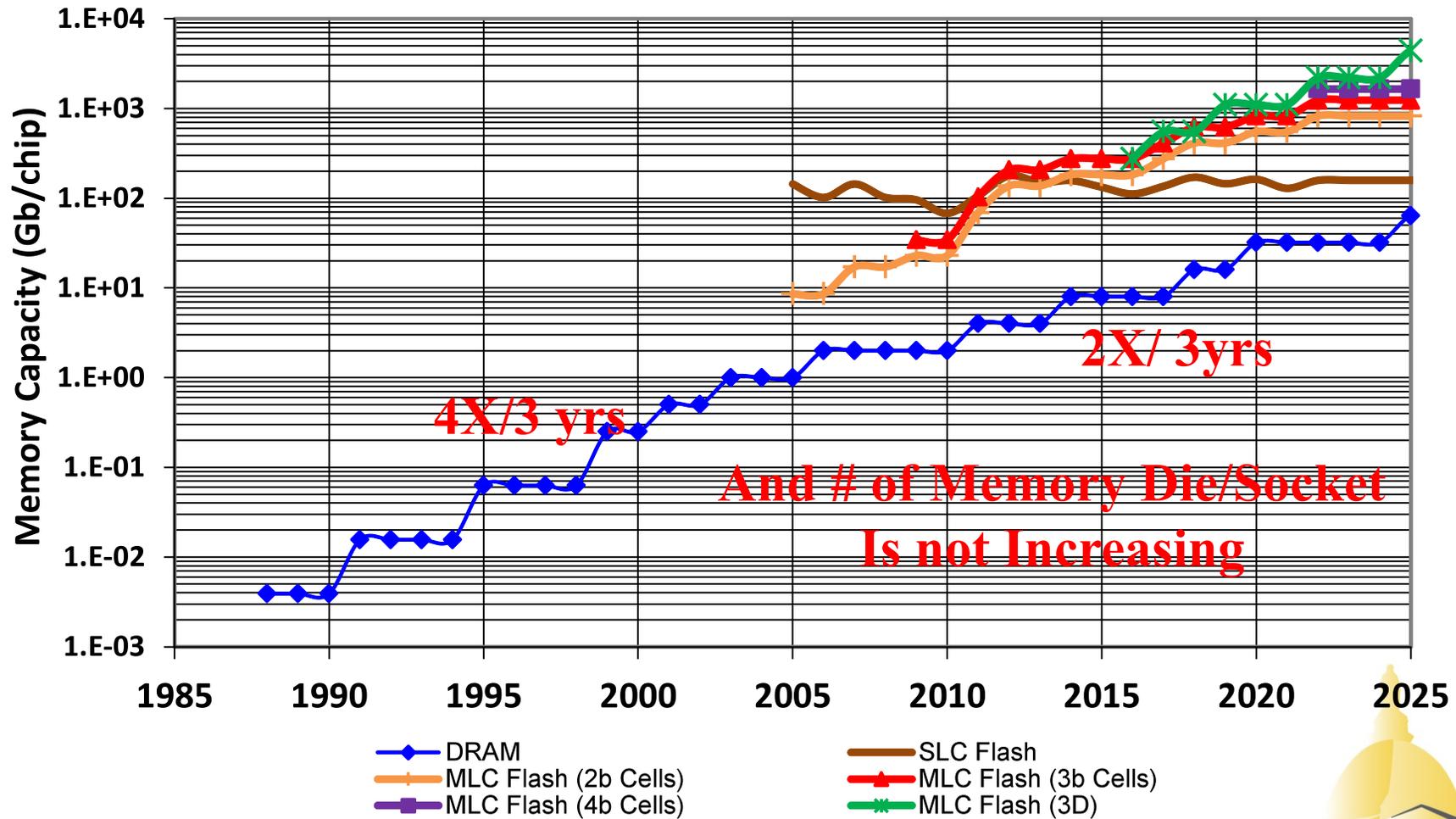
Memory Density Increasing



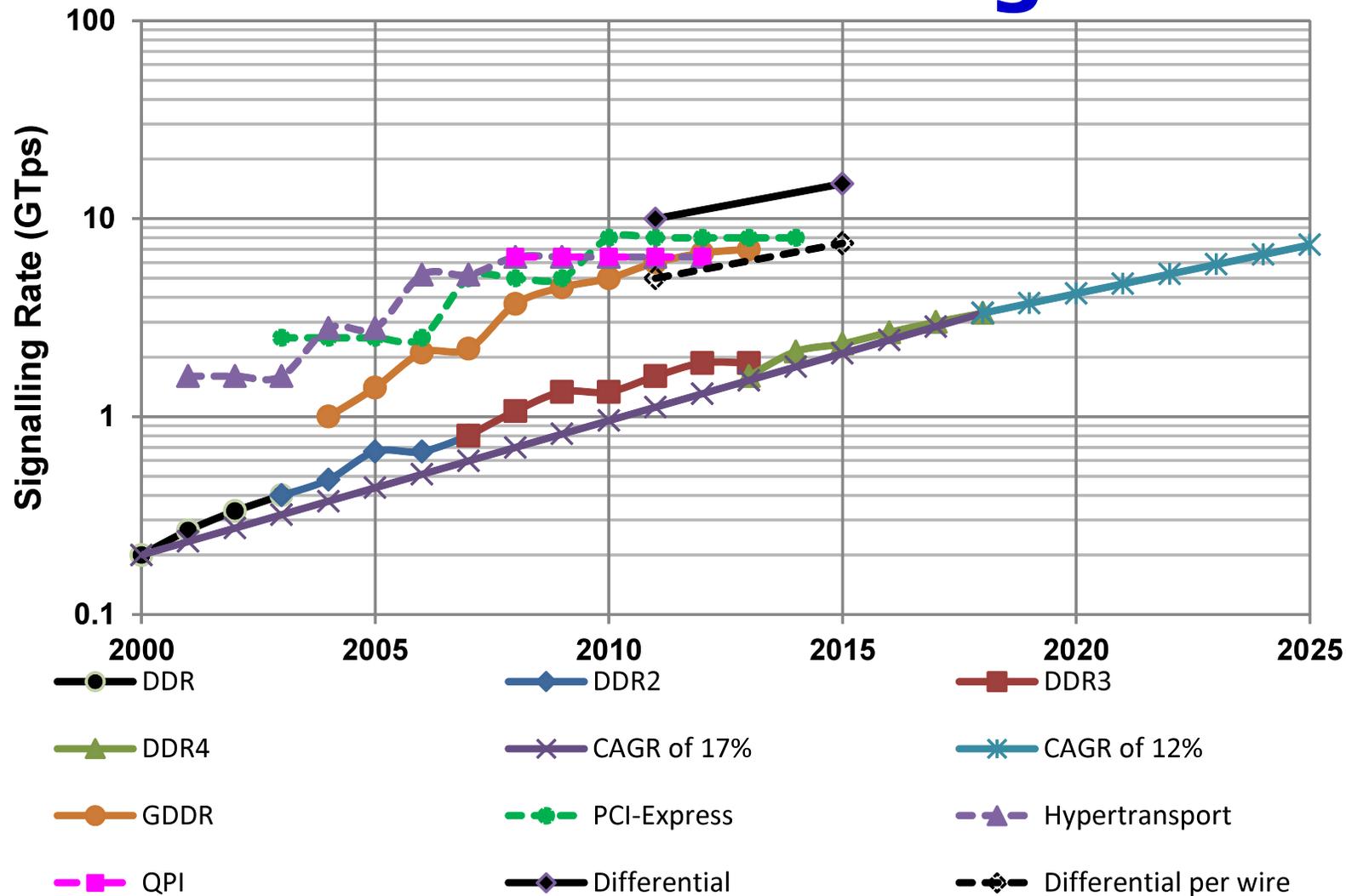
But DRAM Die Sizes Are Flattening or Decreasing



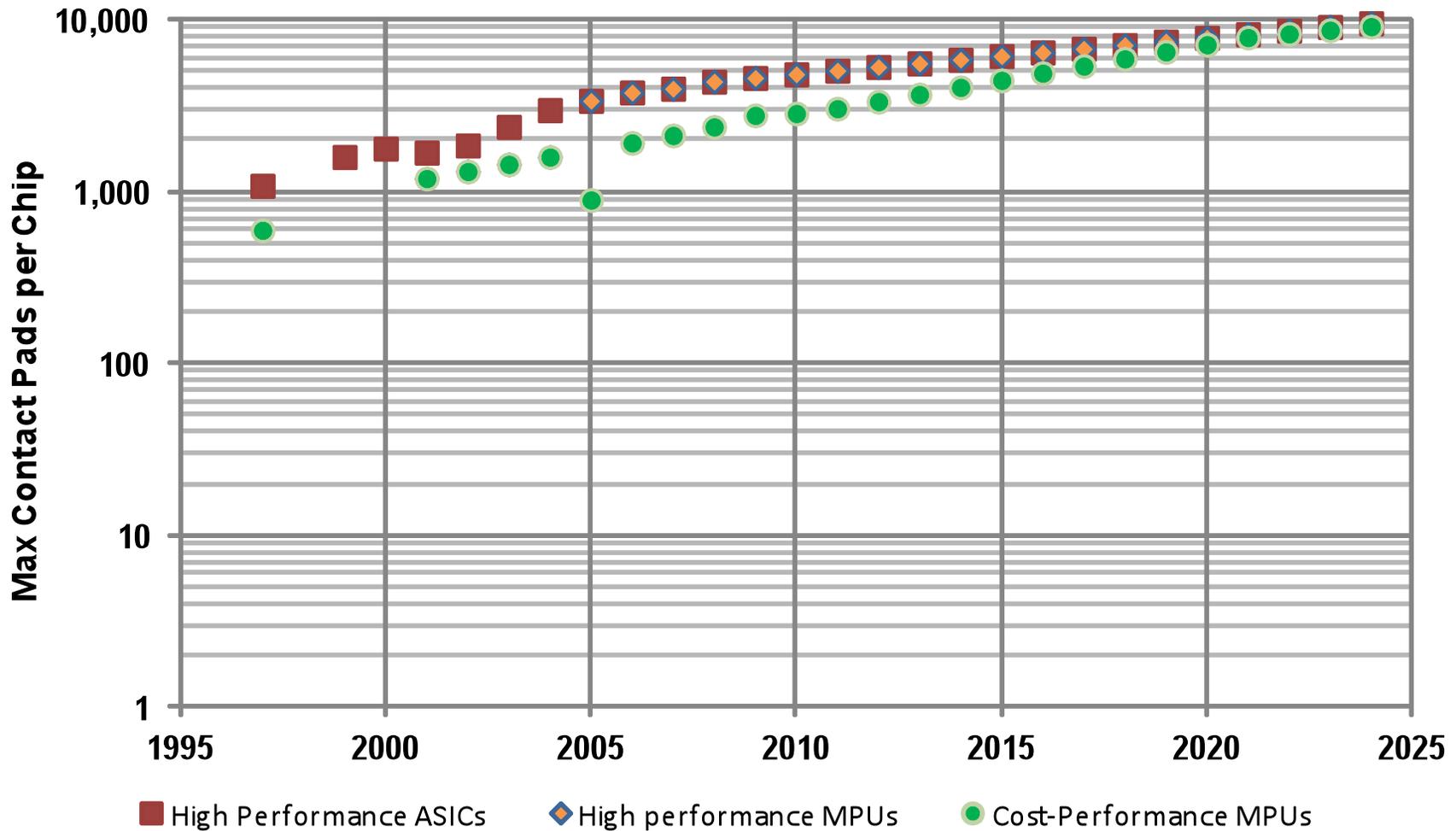
So Memory Density Growth/ Die is Slowing



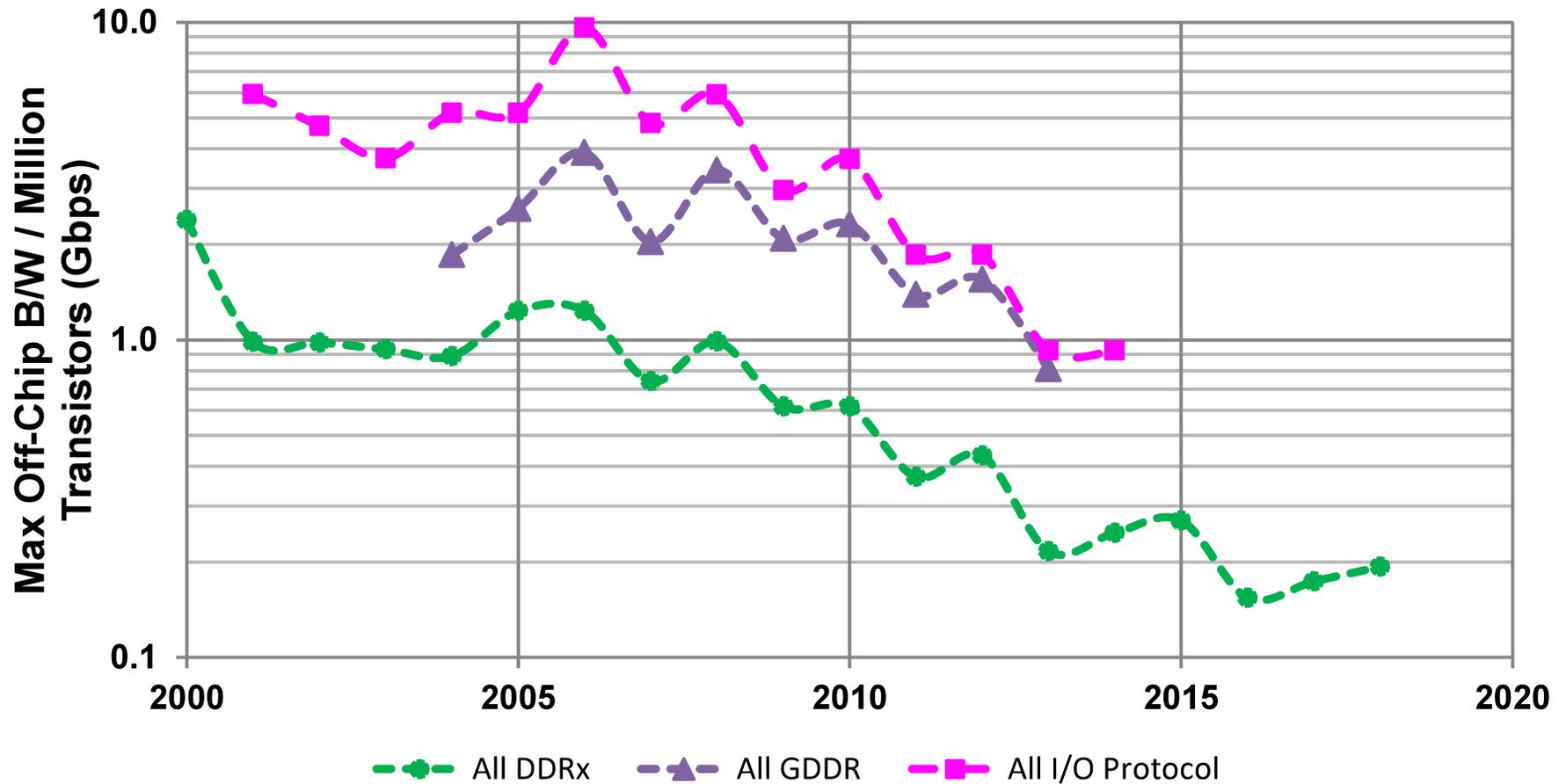
Off-Chip Signaling Rates Have Hit A Ceiling



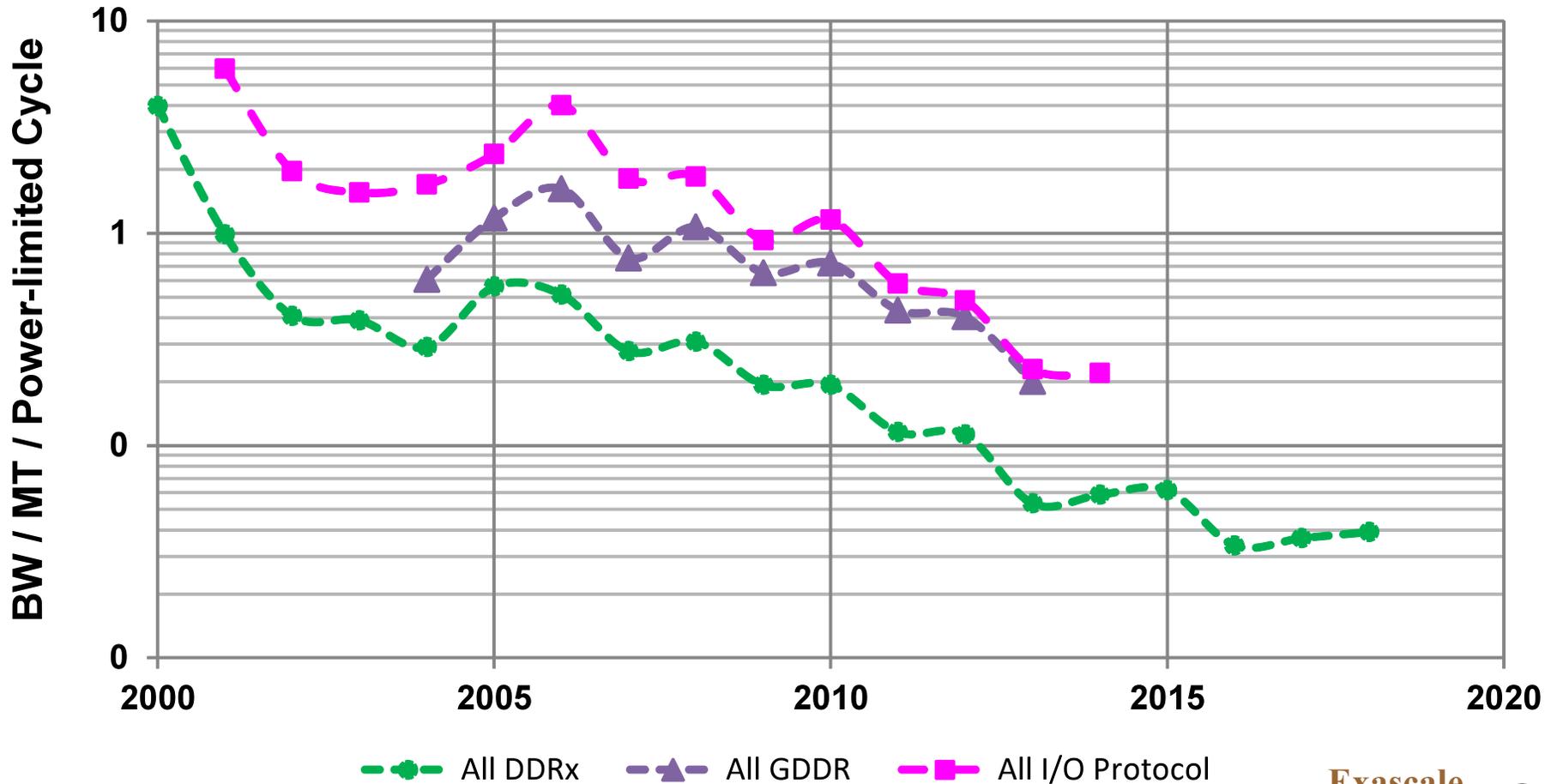
But Growth In Chip I/O is at Best Slow



With Max "Per Unit Logic" Off-Chip B/W Decaying



With Even Less B/W When We Consider Real Clocks



Exascale
Strawman



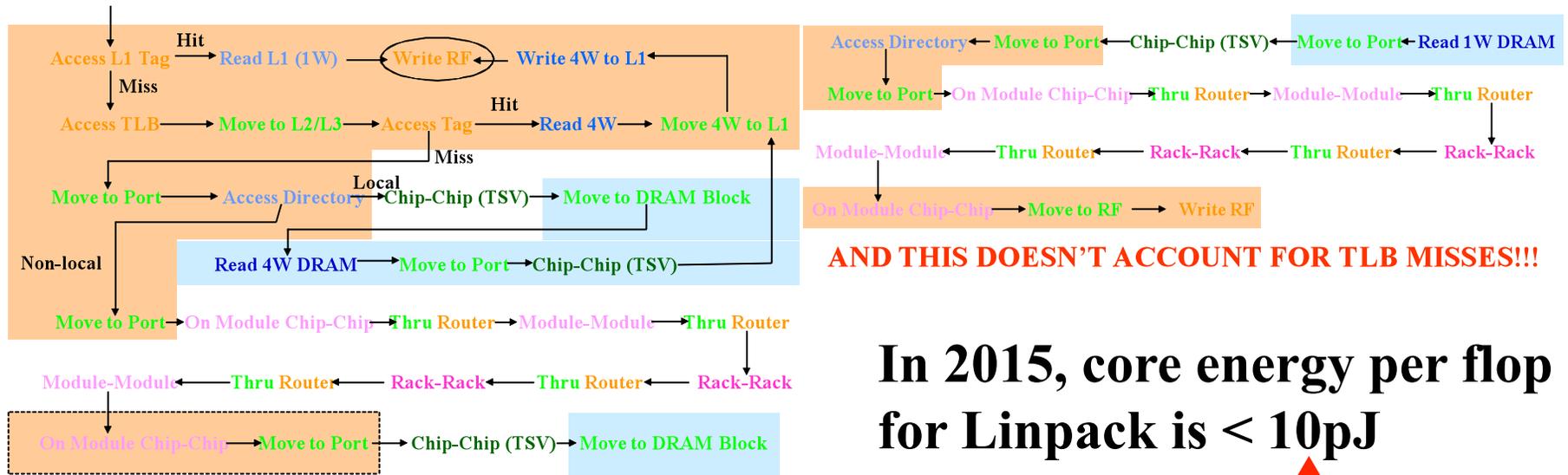
UNIVERSITY OF
NOTRE DAME

ATPESC July 29, 2013

ENABLING
INNOVATION



Relook at Exascale Strawman



In 2015, core energy per flop for Linpack is < 10pJ

Operation	Energy (pJ/bit)
Register File Access	0.16
SRAM Access	0.23
DRAM Access	1
On-chip movement	0.0187
Thru Silicon Vias (TSV)	0.011
Chip-to-Board	2
Chip-to-optical	10
Router on-chip	2

Step	Target	pJ	#Occurrences	Total pJ	% of Total
Read Alphas	Remote	13,819	4	55,276	16.5%
Read pivot row	Remote	13,819	4	55,276	16.5%
Read 1st Y[i]	Local	1,380	88	121,440	36.5%
Read Other Y[i]s	L1	39	264	10,116	3.1%
Write Y's	L1	39	352	13,900	4.2%
Flush Y's	Local	891	88	78,380	23.4%
Total				334,656	
Ave per Flop				475	

50X

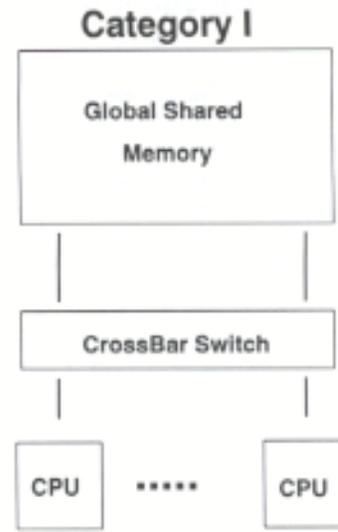
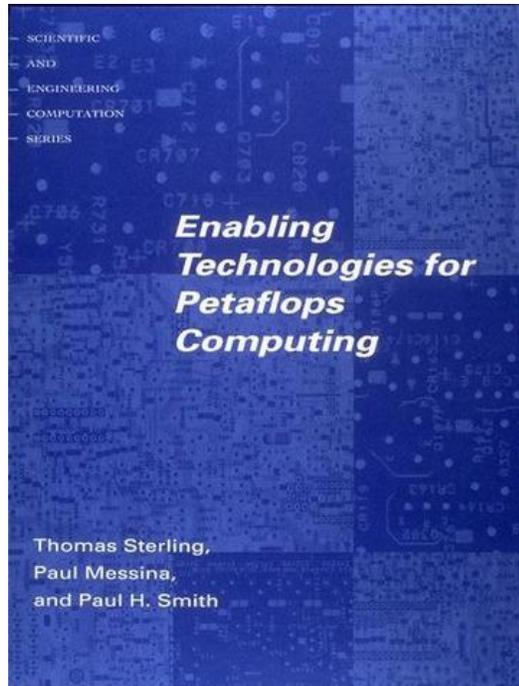
If this is true, 1 EF/s = 0.5 GW!



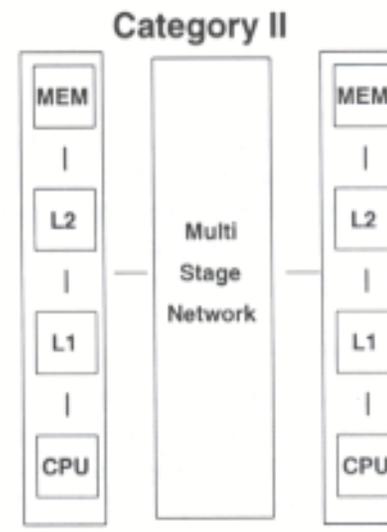
Processing In Memory



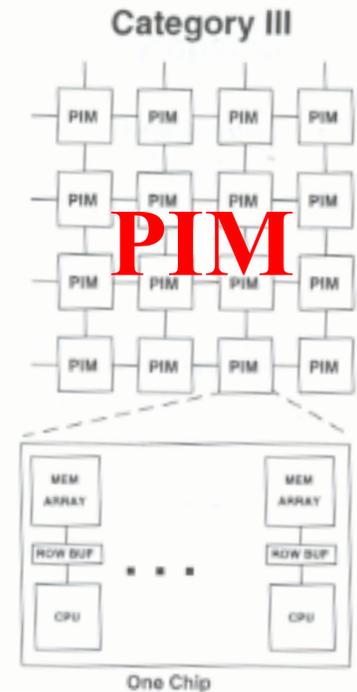
The 3 1994 Approaches to Petaflops



Seymour Cray



Thomas Sterling



Peter Kogge

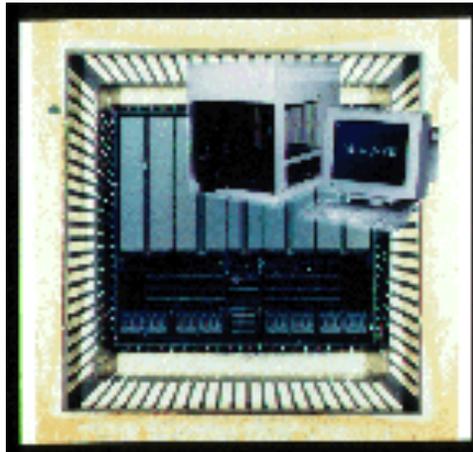
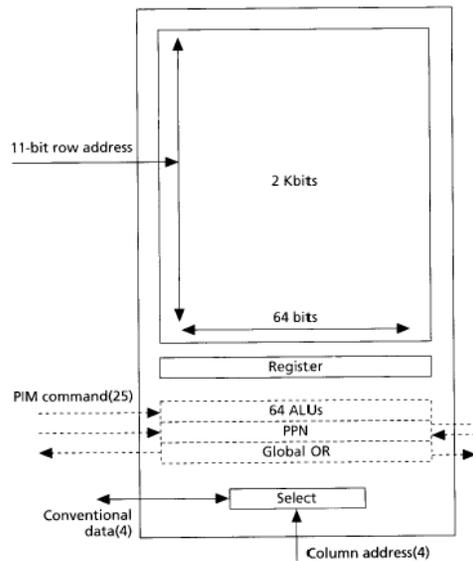


PIM

- Only way to get a lot of memory is *a lot of memory!*
- Current memory *wastes* 98% of actual data fetched within DRAM chip
- Bulk of energy costs on
 - shipping small piece of requested data off chip
 - transporting it up and down cache hierarchy
 - over long on-chip distances
- Obvious solution: place cores on memory
- But still permit large multi-chip systems



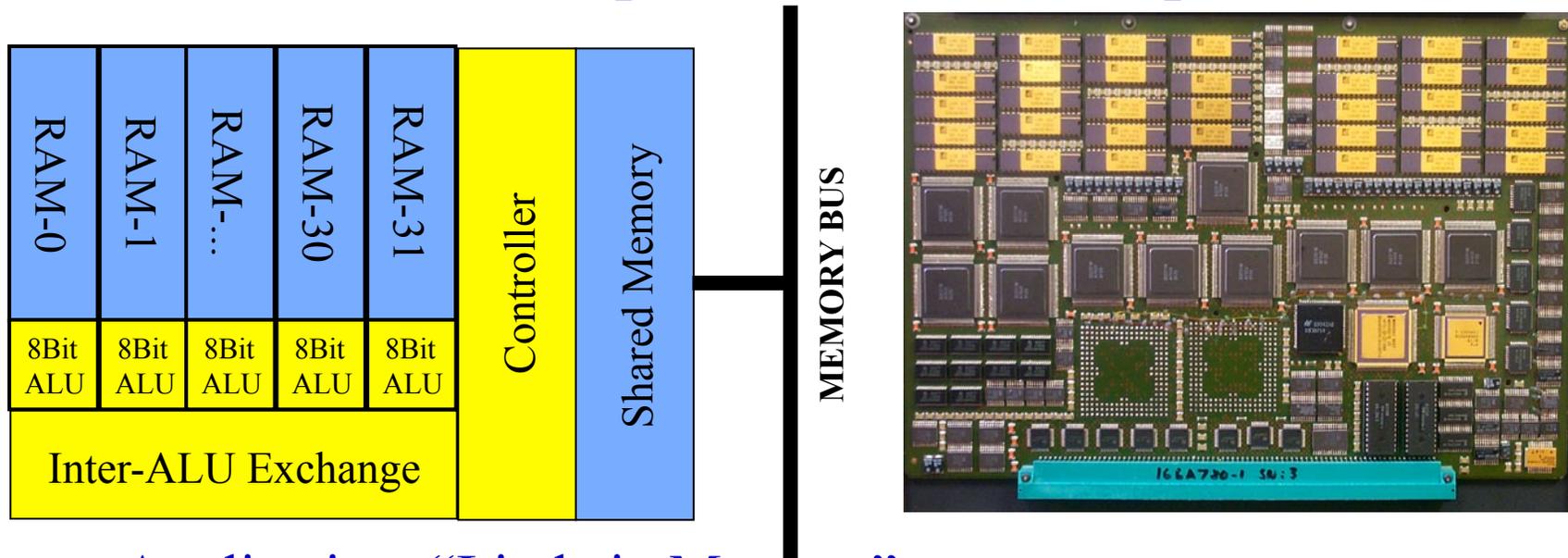
TERASYS SIMD PIM (circa 1993)



- Memory part for CRAY-3
- “Looked like” SRAM memory
 - With extra command port
- 128K SRAM bits (2k x 64)
- 64 1 bit ALUs
- SIMD ISA
- Fabled by National
- Also built into workstation with 64K processors
 - 5-48X Y-MP on 9 NSA benchmarks



RTAIS: Search In Memory (circa 1993)

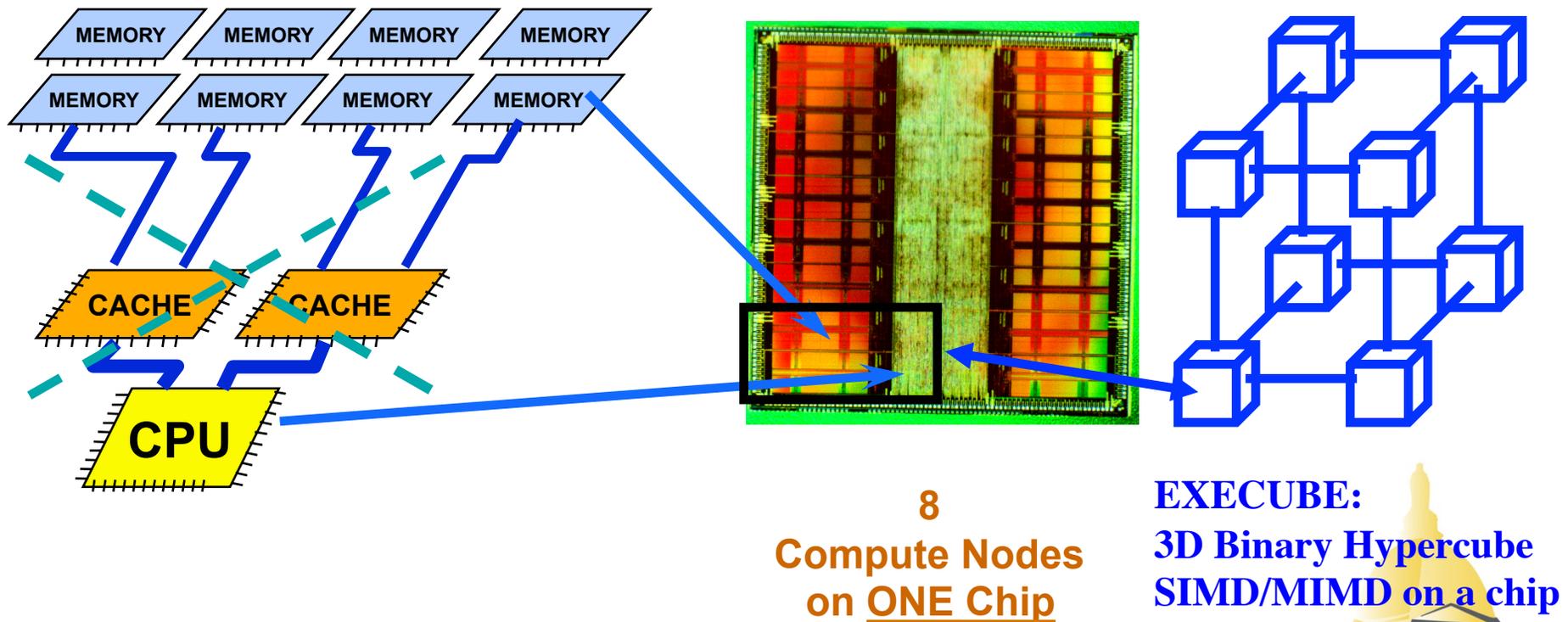


- Application: “Linda in Memory”
- Designed from onset to perform wide ops “at the sense amps”
- More than SIMD: flexible mix of VLIW
- “Object oriented” multi-threaded memory interface
- Result: 1 card 60X faster than state-of-art R3000 card

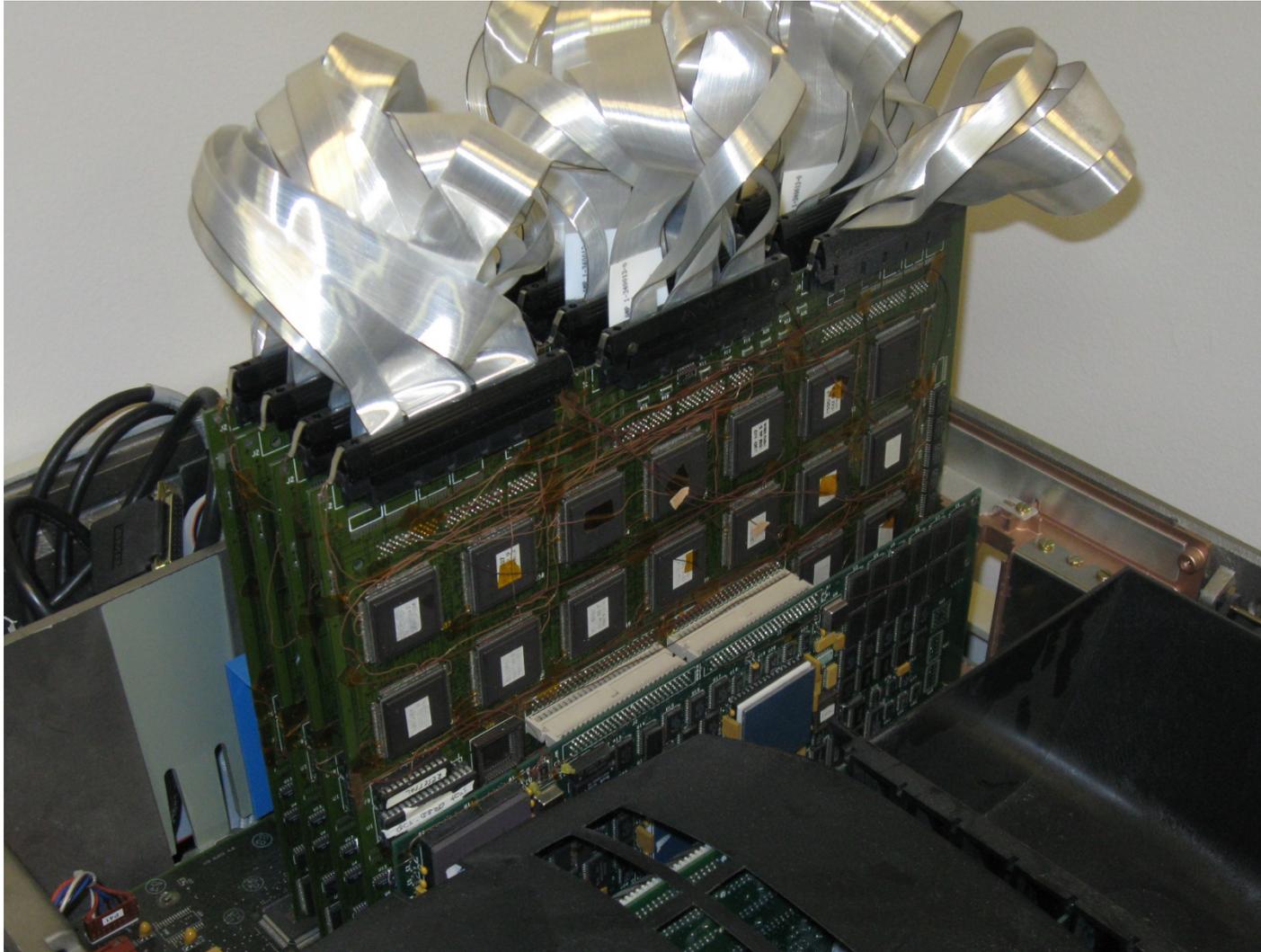


EXECUBE: SPMD on Chip (1993)

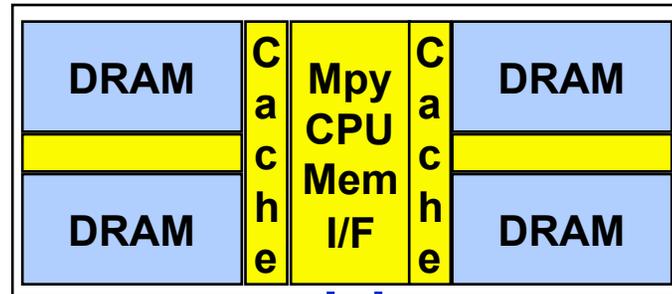
- First DRAM-based Multi-core on a Chip
- *Designed from onset for "glueless" one-part-type scalability*



An Array of EXECUBEs



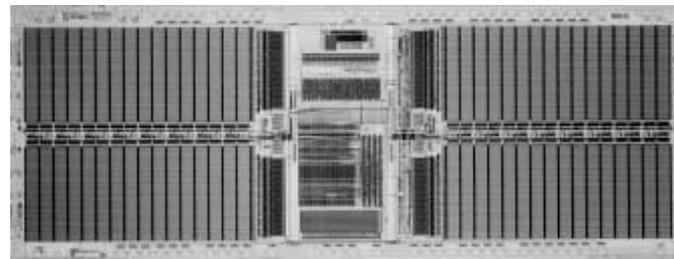
Mitsubishi M32R/D (circa 1997)



Also two 1-bit I/Os

16 bit data bus

24 bit address bus



- 32-bit fixed point CPU + 2 MB DRAM
- “Memory-like” Interface
- Utilize wide word I/F from DRAM macro for cache line



Linden DAAM Chip (1998)

- Designed for in-memory text search
- 16 Mbit DRAM divided into 64 blocks
- 64 1-bit Processing Elements per block
 - 4K PEs/chip

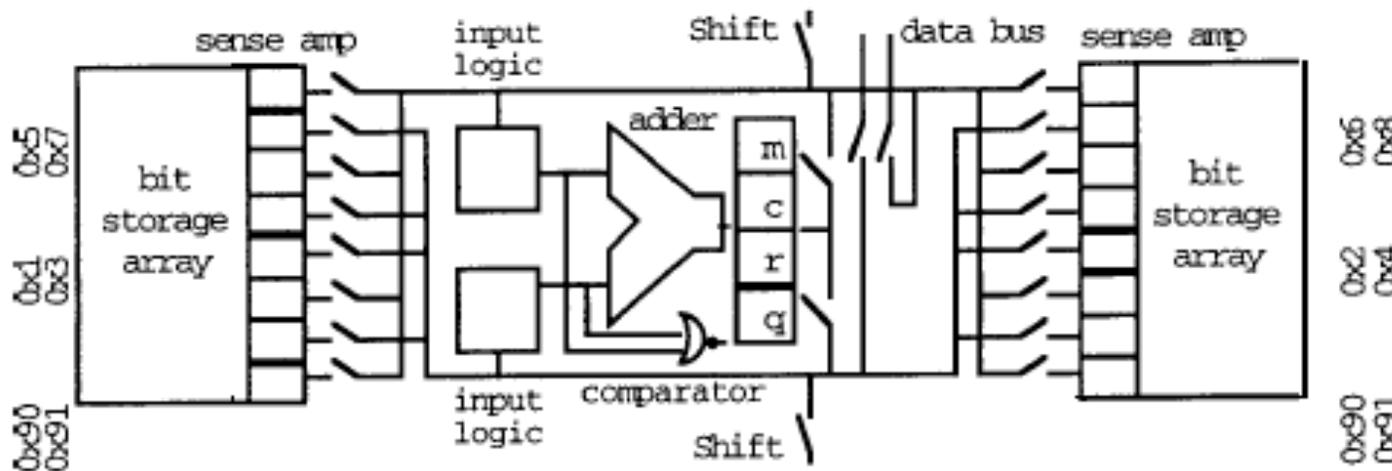
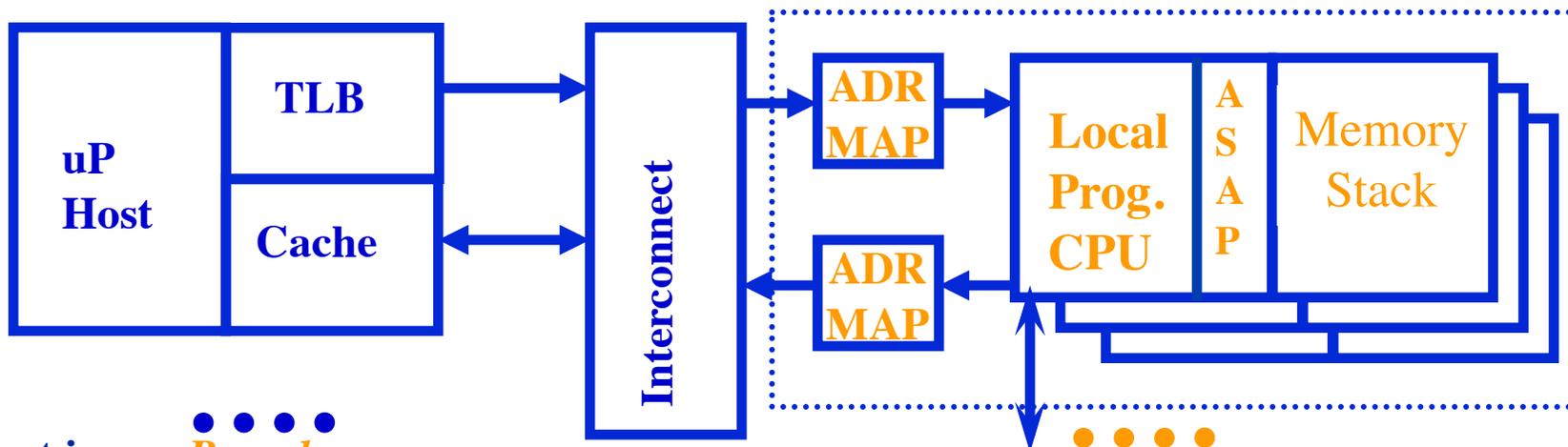


Fig. 1 from Lipovski and Yu, “The Dynamic Associative Access Memory Chip and its Application to SIMD Processing and Full-text Database Retrieval”, 1999

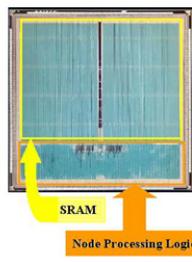
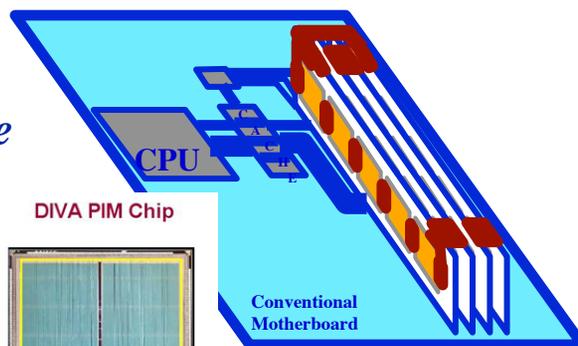


DIVA: Smart DIMMs for Irregular Data Structures



Host issues *Parcels*

- Generalized “Loads & Stores”
- Treat memory as *Active* Object-oriented store



- 1 CPU + 2MB
- MIPS + “Wide Word”

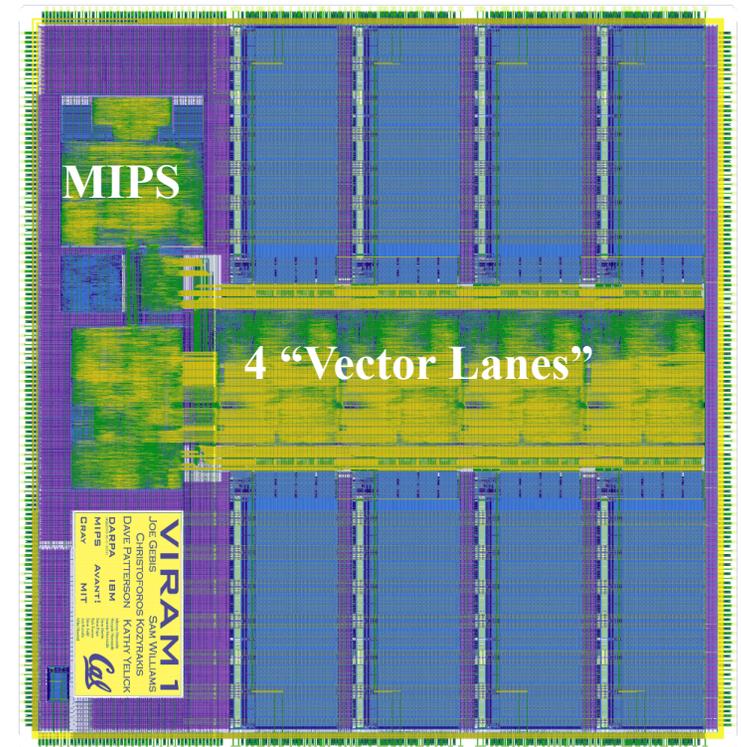
DIVA Functions:

- Prefix operators
- Dereferencing & pointer chasing
- Compiled methods
- Multi-threaded
- May generate parcels



Berkeley VIRAM

- System Architecture: single chip media processing
- ISA: MIPS Core + Vectors + DSP ops
- 13 MB DRAM in 8 banks
- Includes flt pt
- 2 Watts @ 200 MHz, 1.6GFlops



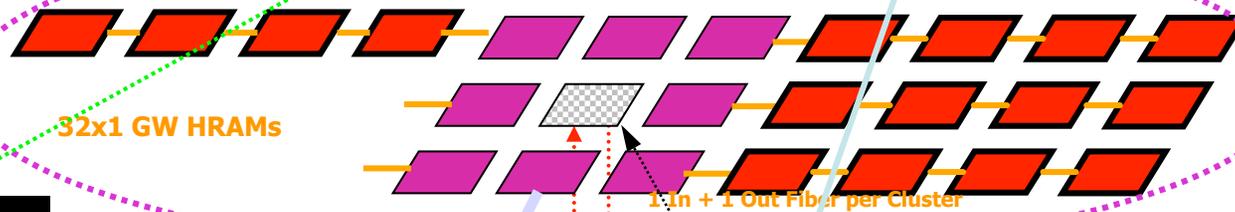
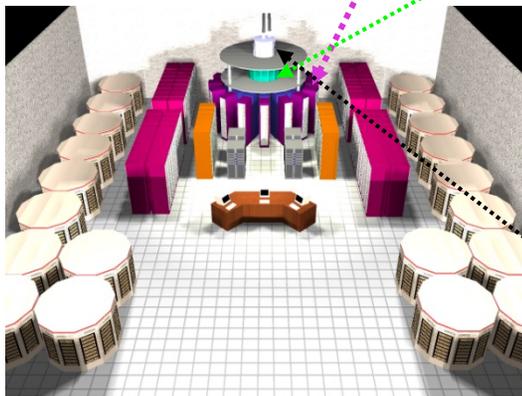
HTMT: The Original Petaflops Initiative (circa 2000)

DRAM PIM Cluster

- 8 PIM chips/cluster
- 16 nodes/chip
- 32 MB, 2 GF/node
- 4 GB, 256 GF/cluster

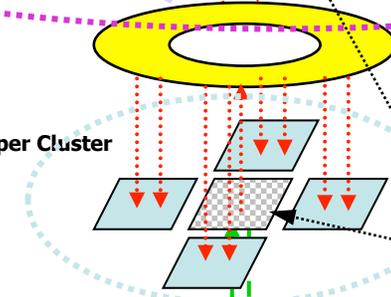
SRAM PIM Cluster

- 4 PIM chips/cluster
- 16 nodes/chip
- 4 MB, 4 GF/node
- 256 MB, 256 GF/cluster



DATA VORTEX

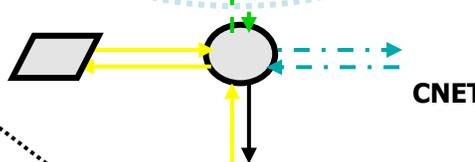
8 In + 1 Out Fiber per Cluster



- 1 GW/s Wire
- ⋯ 10 GW/s Fiber
- - - 20 GW/s RSFQ
- - - 64 GW/s Wire
- ← 256 GW/s RSFQ

Fiber Modulator Chip

128 KW CRAM



Replicated
4096 times

256 GFLOPS SPELL



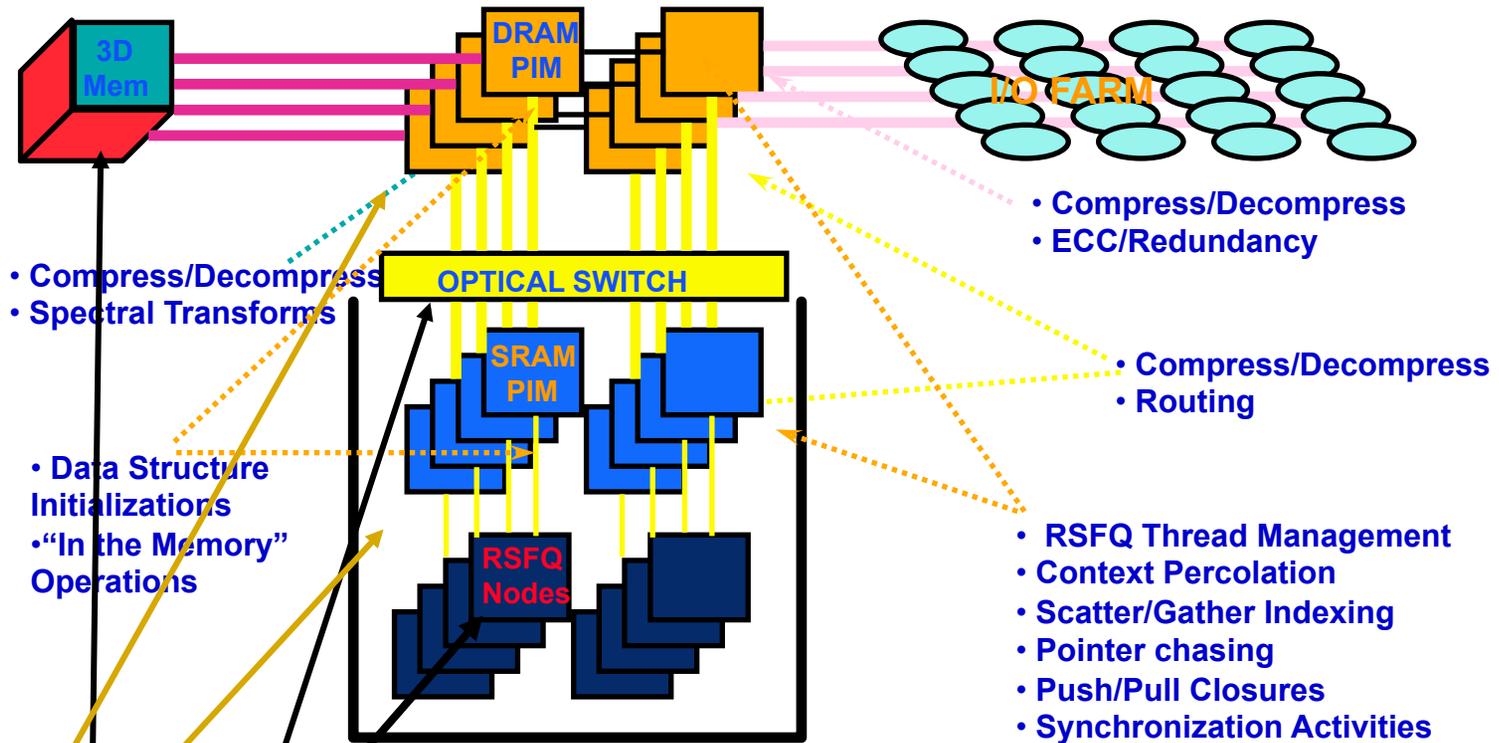
UNIVERSITY OF
NOTRE DAME

ATPESC July 29, 2013

ENABLING
INNOVATION



The HTMT Architecture & PIM Functions



New Technologies:

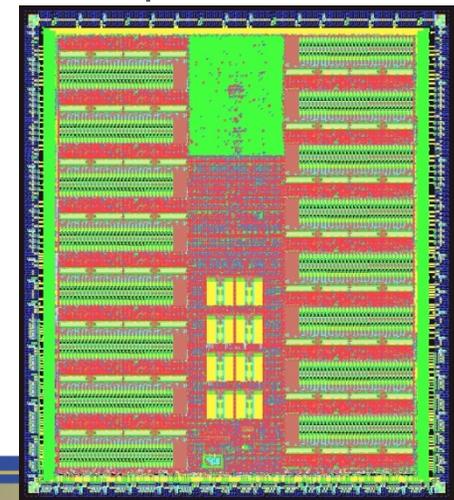
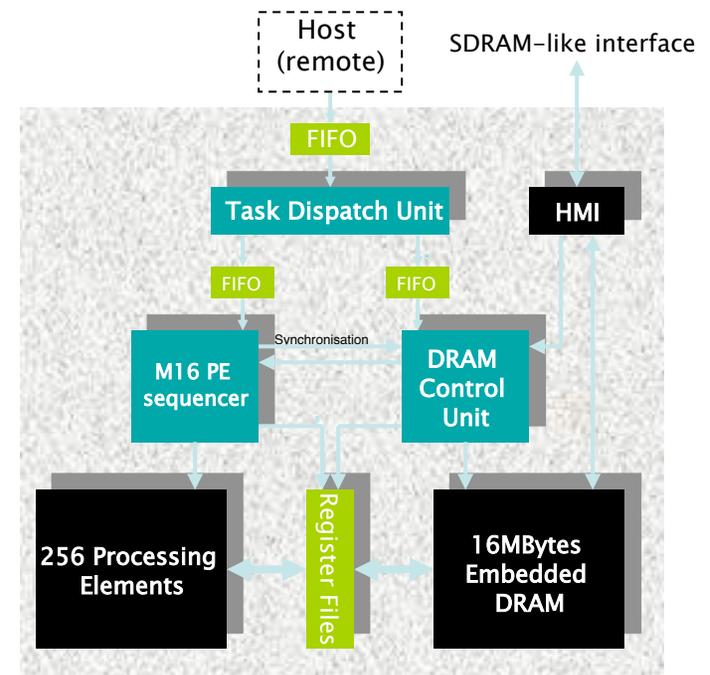
- Rapid Single Flux Quantum (RSFQ) devices for 100 GHz CPU nodes
- WDM all optical network for petabit/sec bi-section bandwidth
- Holographic 3D crystals for Petabytes of on-line RAM
- PIM for active memories to manage latency

PIMs in Charge



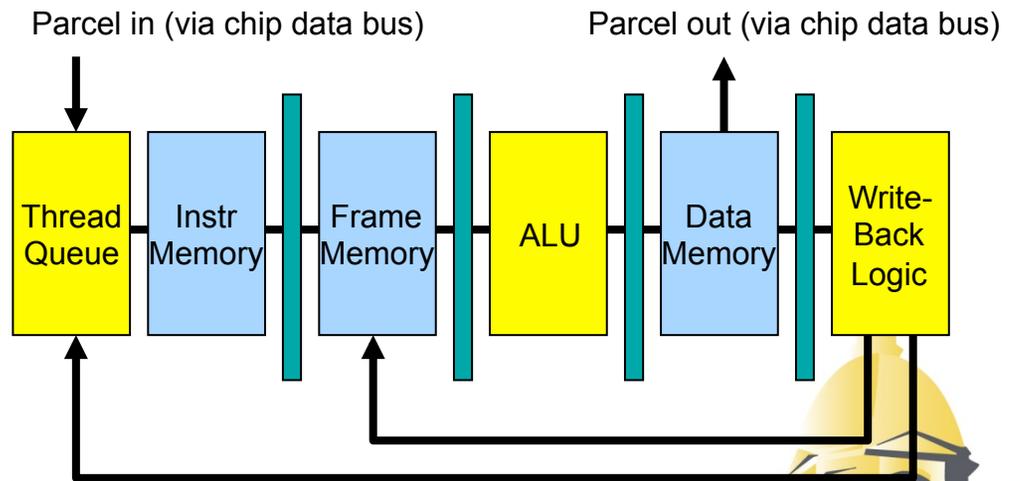
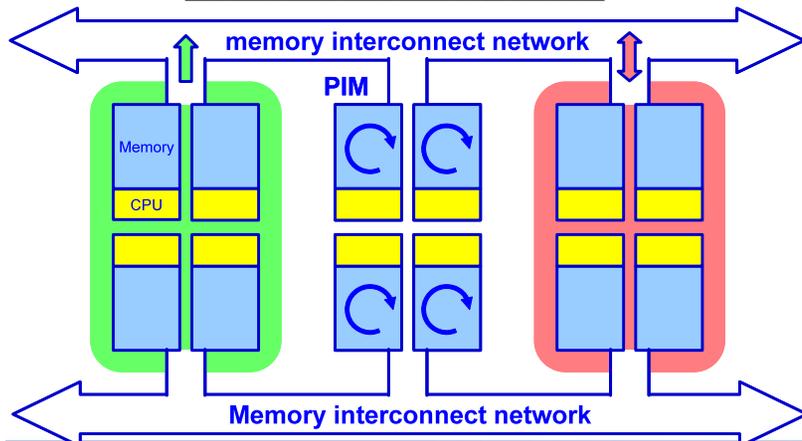
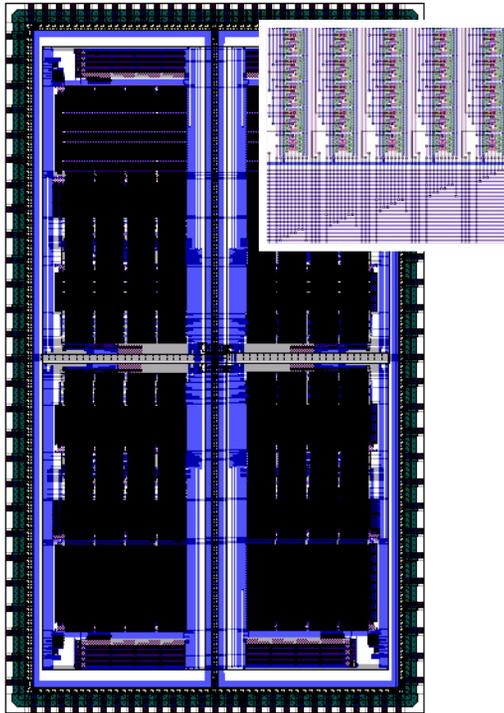
Micron Yukon

- 0.15 μm eDRAM/ 0.18 μm logic process
- 128Mbits DRAM
 - 2048 data bits per access
- 256 8-bit integer processors
 - Configurable in multiple topologies
- On-chip programmable controller
- Operates like an SDRAM

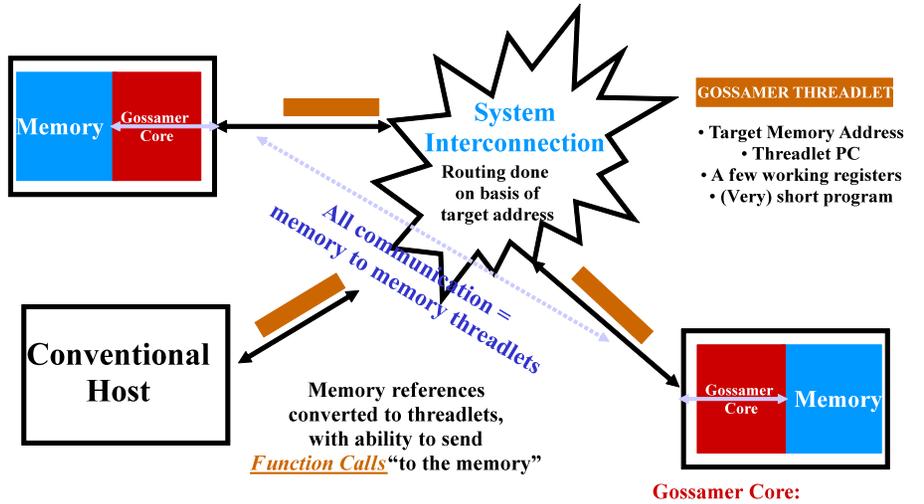


PIM Lite

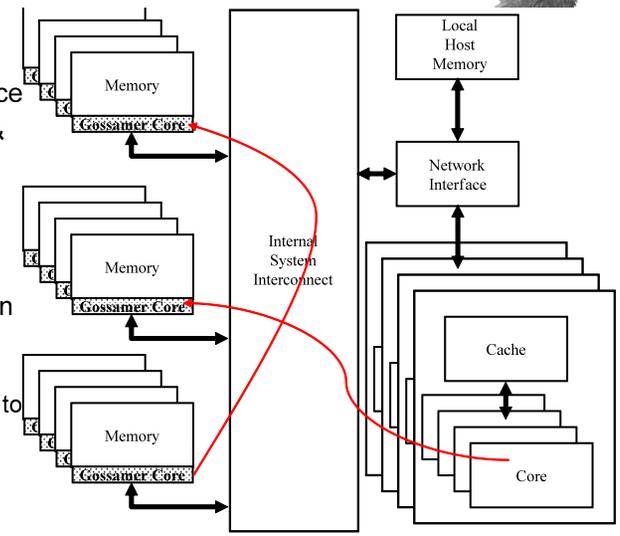
- “Looks like memory” at Interfaces
- ISA: 16-bit multithreaded/SIMD
 - “Thread” = IP/FP pair
 - “Registers” = wide words in frames
- Multiple nodes per chip
- 1 node logic area ~ 10.3 KB SRAM (comparable to MIPS R3000)
- TSMC 0.18u 1-node in fab now
- 3.2 million transistors (4-node)



Traveling Threadlets

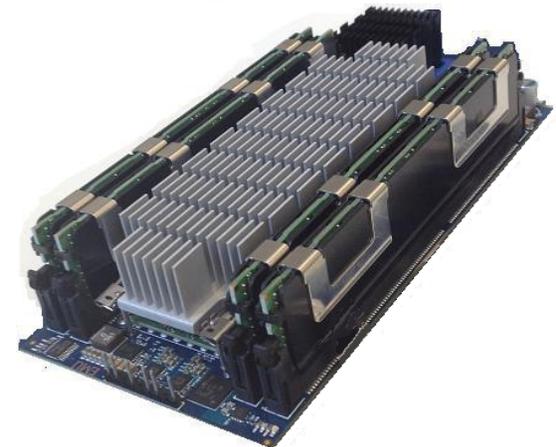


- Single Address Space
- Visible to all Hosts & Gossamer Cores
- Hosts can issue
 - Reads and Writes
 - **Threadlets**
- Gossamer Cores can
 - Spawn new **threadlets**
 - Migrate **threadlets** to other cores



Kogge, "Of Piglets and Threadlets: Architectures for Self-Contained, Mobile, Memory Programming, IWIA, Maui, HI, Jan. 2004

- Single Address Space Visible to all Hosts & Gossamer Cores
- Hosts can launch:
 - Reads and Writes of Memory
 - Threadlets for execution on Gossamer core
- Gossamer Cores can
 - Spawn new threadlets
 - Migrate threadlets to other cores



Processing Near Memory



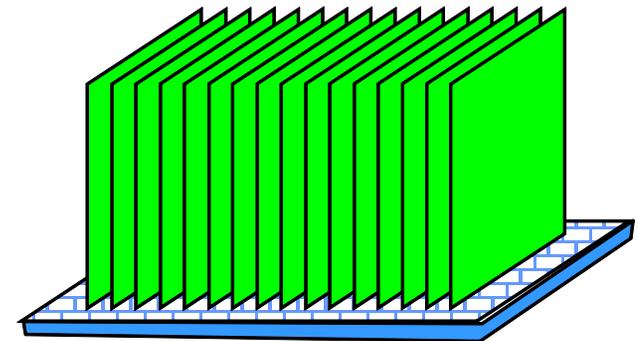
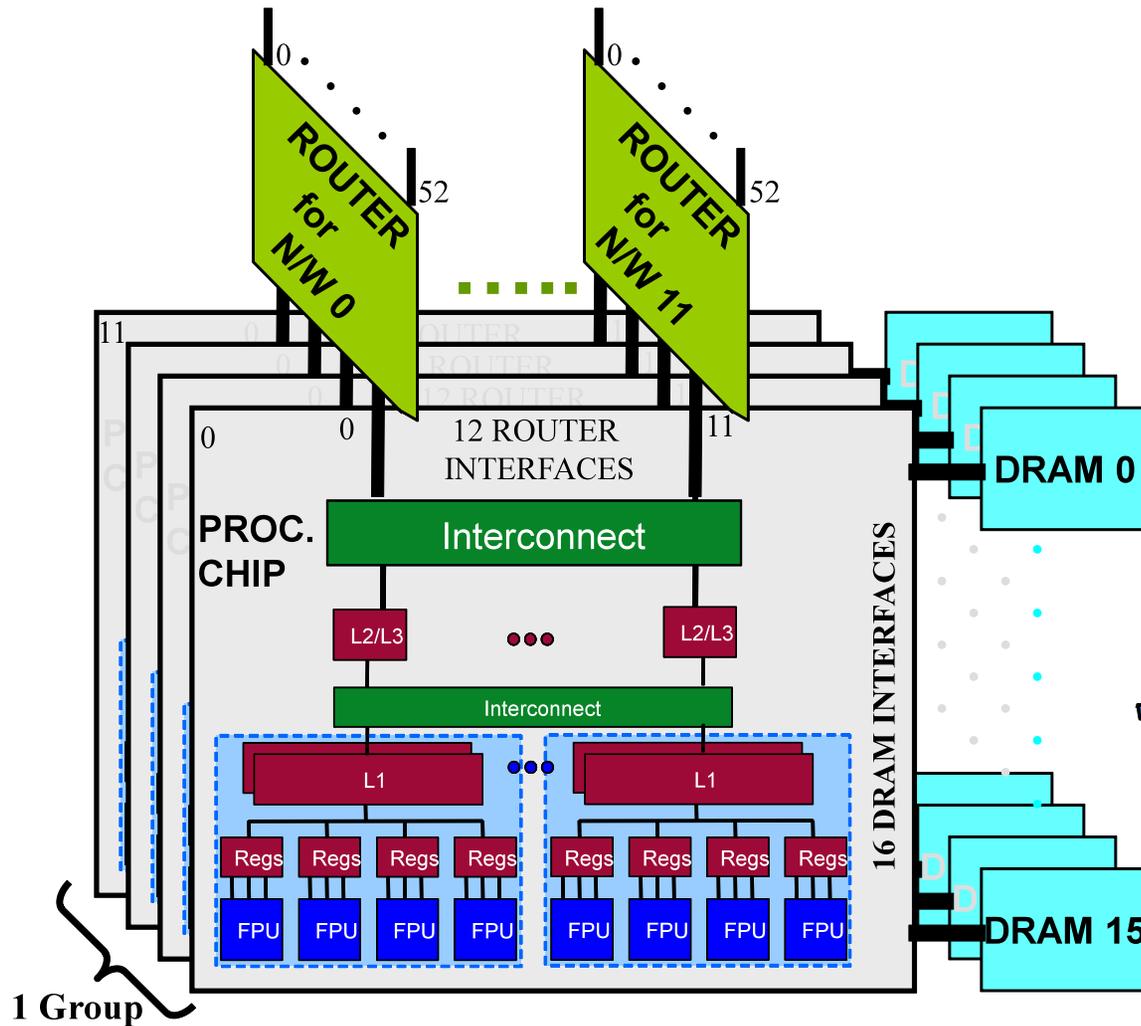
UNIVERSITY OF
NOTRE DAME

ATPESC July 29, 2013

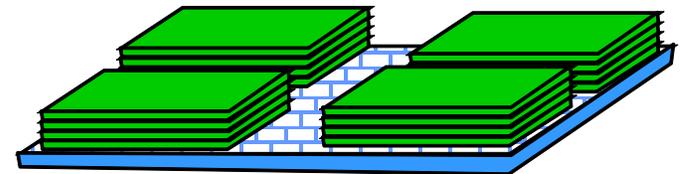
*ENABLING
INNOVATION*



The Exascale Strawman



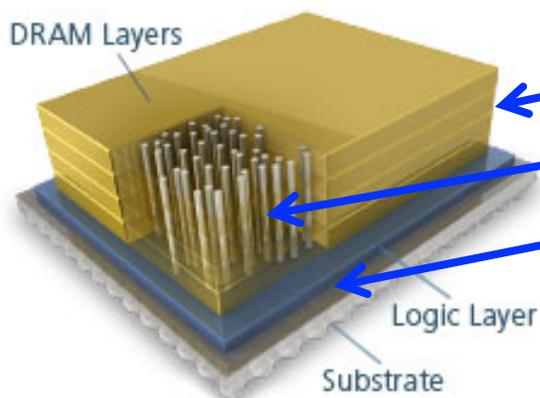
(a) Quilt Packaging



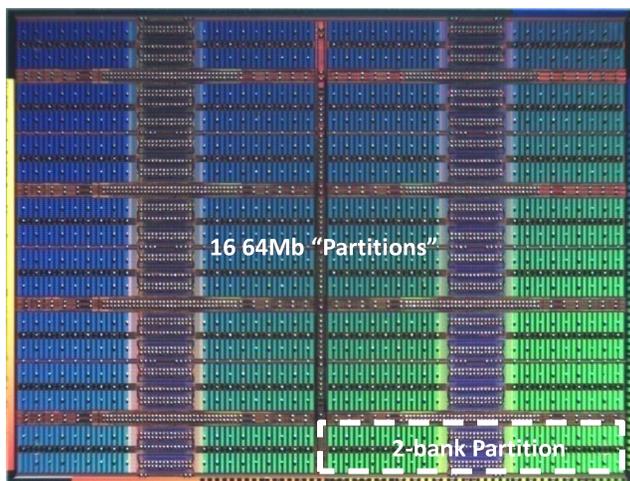
(b) Thru via chip stack

1 Cabinet Contains 32 Groups on 12 Networks

“Please Sir, I want more” The Emergence of Hybrid 3D Memory



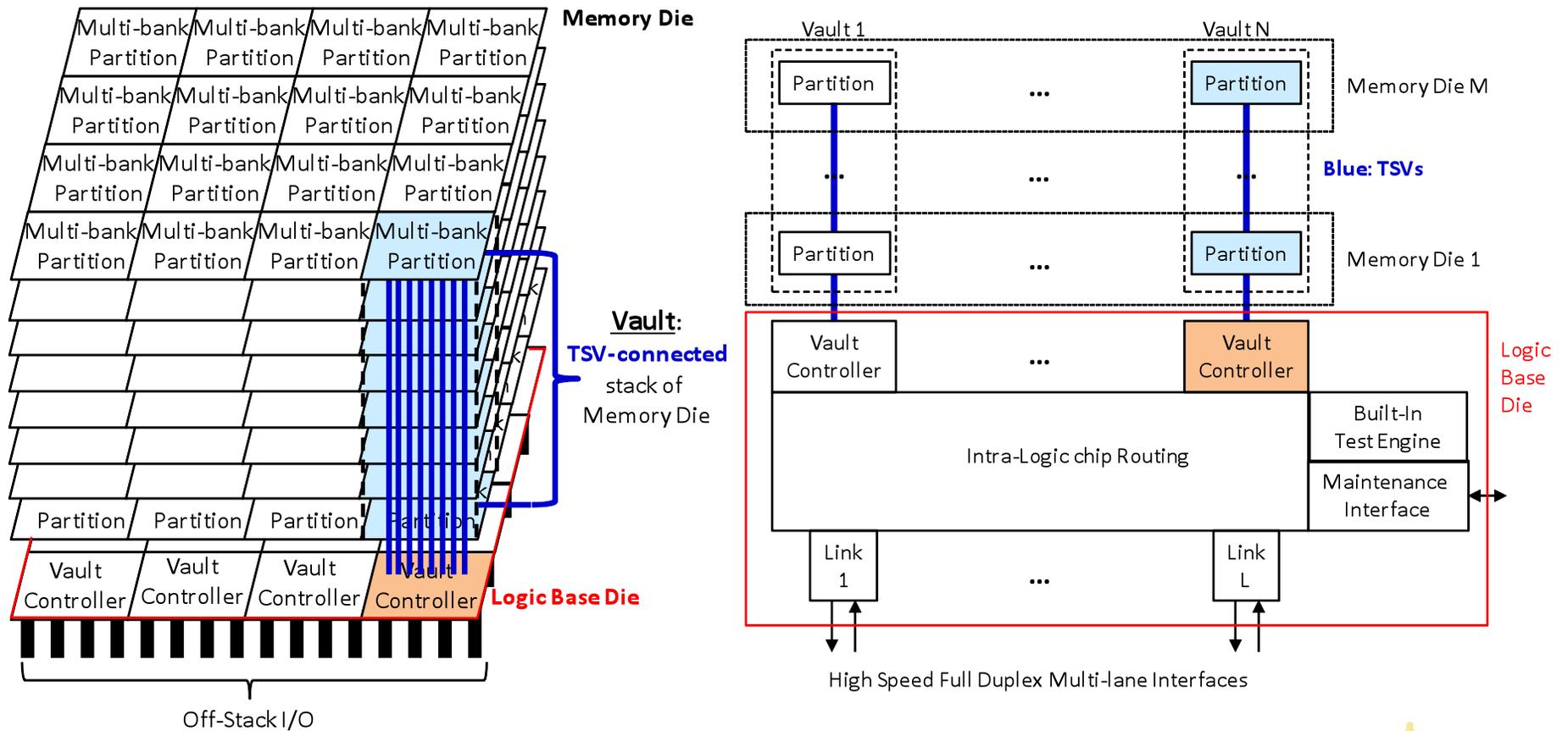
<http://www.micron.com/products/hybrid-memory-cube>



- Stackable memory chips (no cores)
- “Through Silicon Vias” (TSVs)
- Logic chip on bottom
 - Multiple memory controllers
 - More sophisticated off-stack interfaces than DDR
- Prototype demonstrated in 2011
- 1st Product expected in 2015 timeframe
 - Spec:<http://www.hybridmemorycube.org>
 - Capacity: up to 8GB: 8X single chip
 - Bandwidth: up to 480GB/s: 40X
 - Lots of room on logic chip
- **Bottom Line: Huge increase in**
 - **Memory density**
 - **Bandwidth**



The HMC Architecture



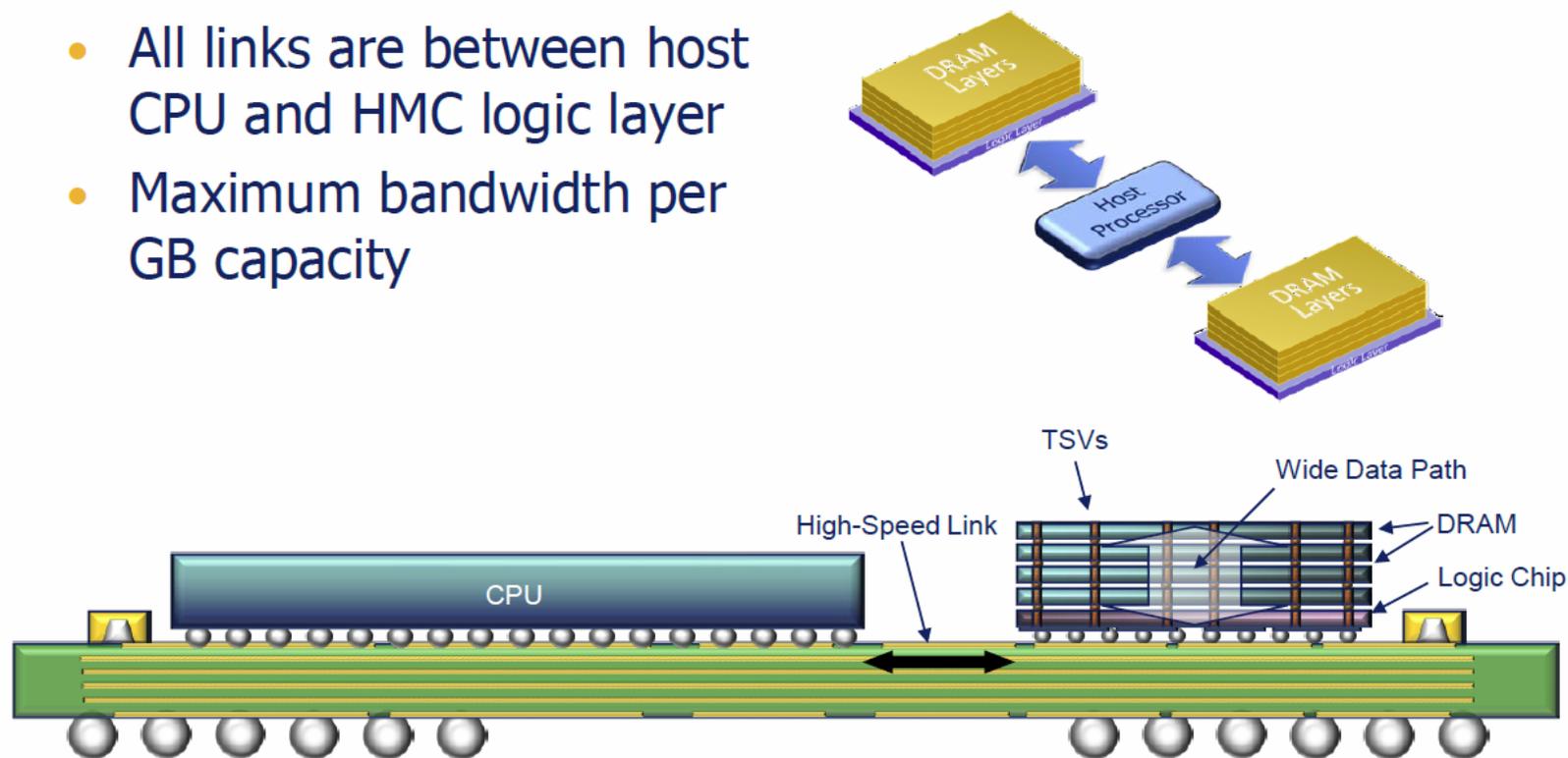
(Shared by All Vault Controllers via Distributed Routing)

- All vaults run independently
- Each vault looks like a set of M dual independent banks



HMC Near Memory – MCM Configuration

- All links are between host CPU and HMC logic layer
- Maximum bandwidth per GB capacity



Notes: MCM = multi-chip module
Illustrative purposes only; height is exaggerated

Enhancing a Conventional Architecture

- CPU(s) sees sea of memory stacks – all “far”
- True address-based routing

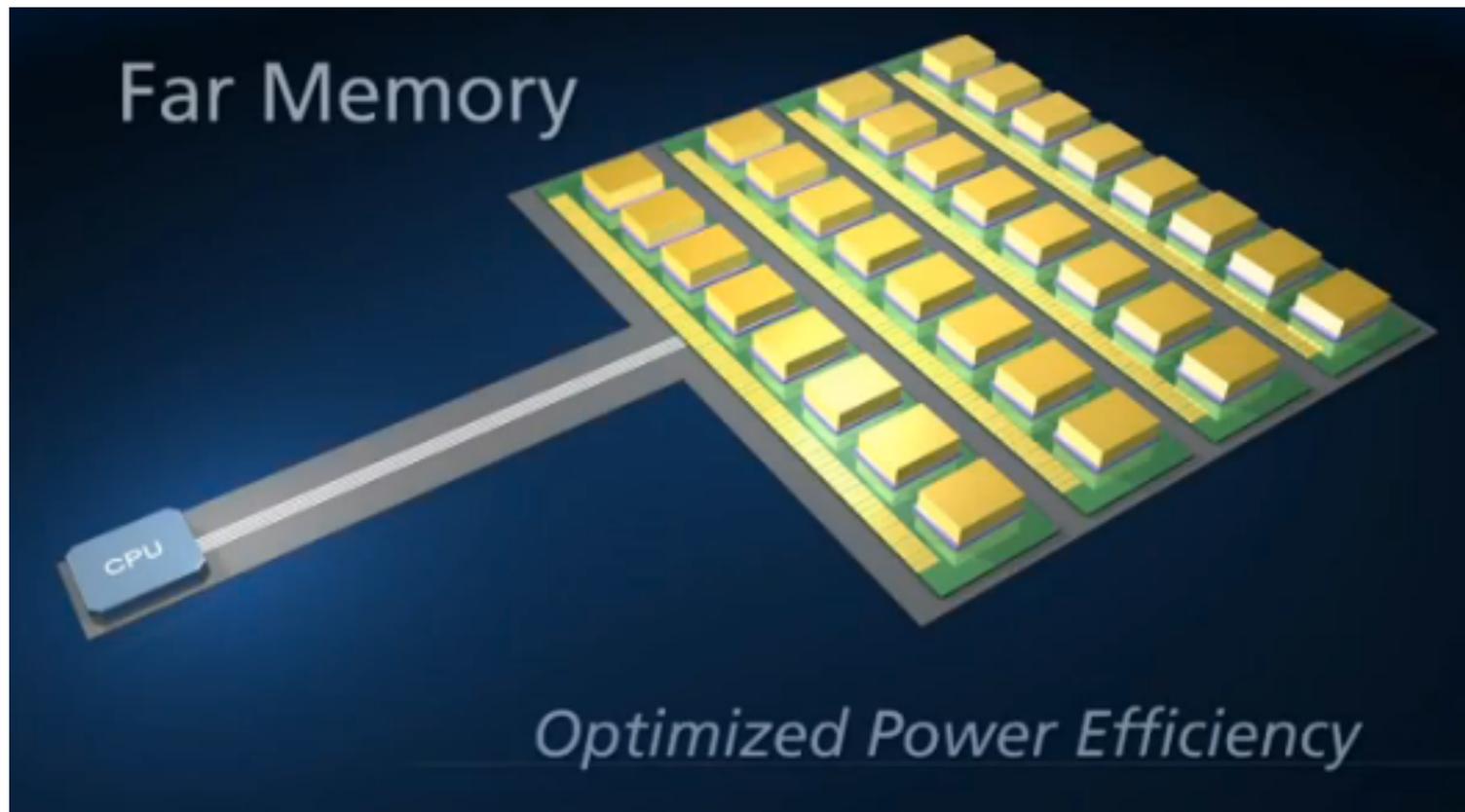
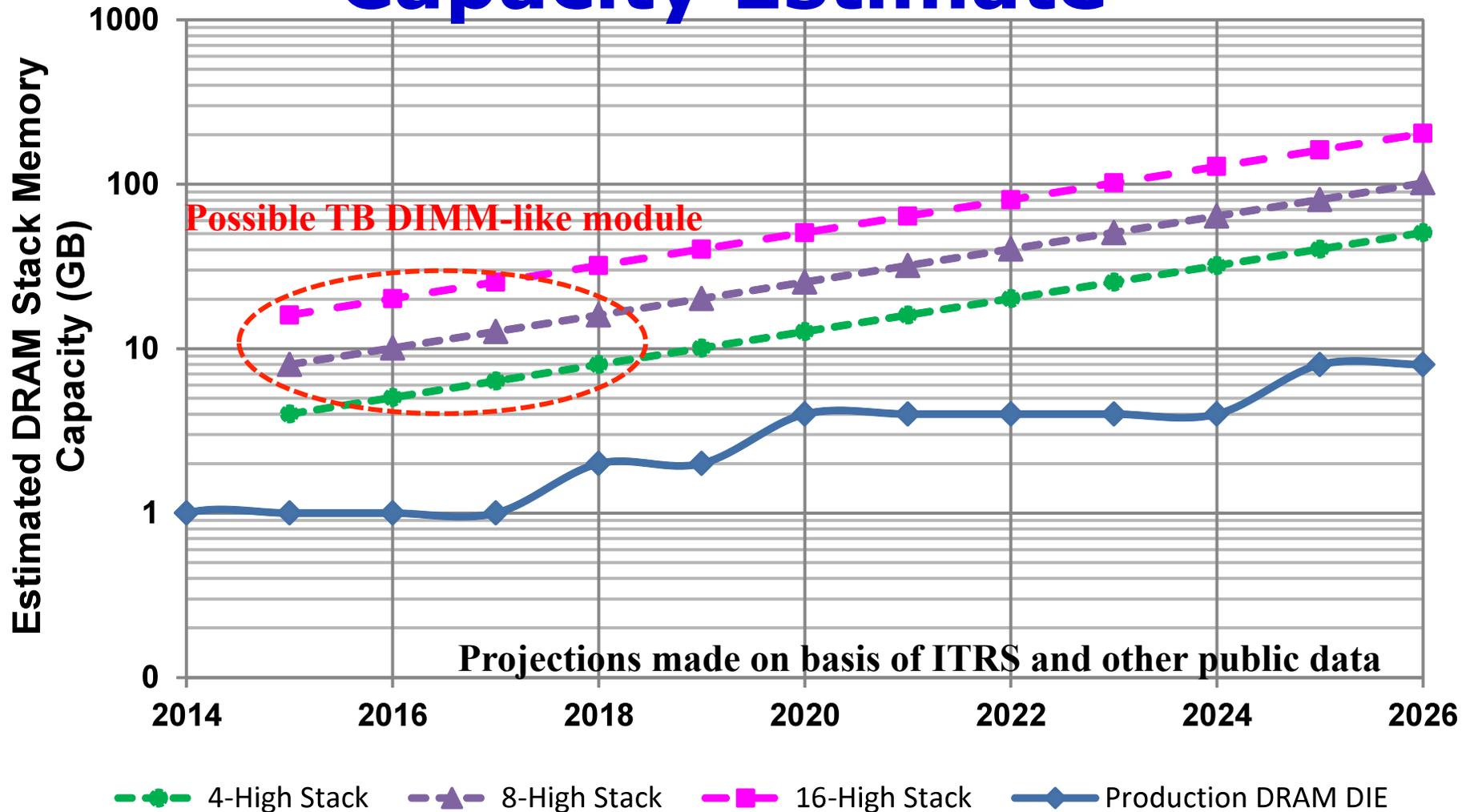


Chart from T. Pawlowski



Capacity Estimate

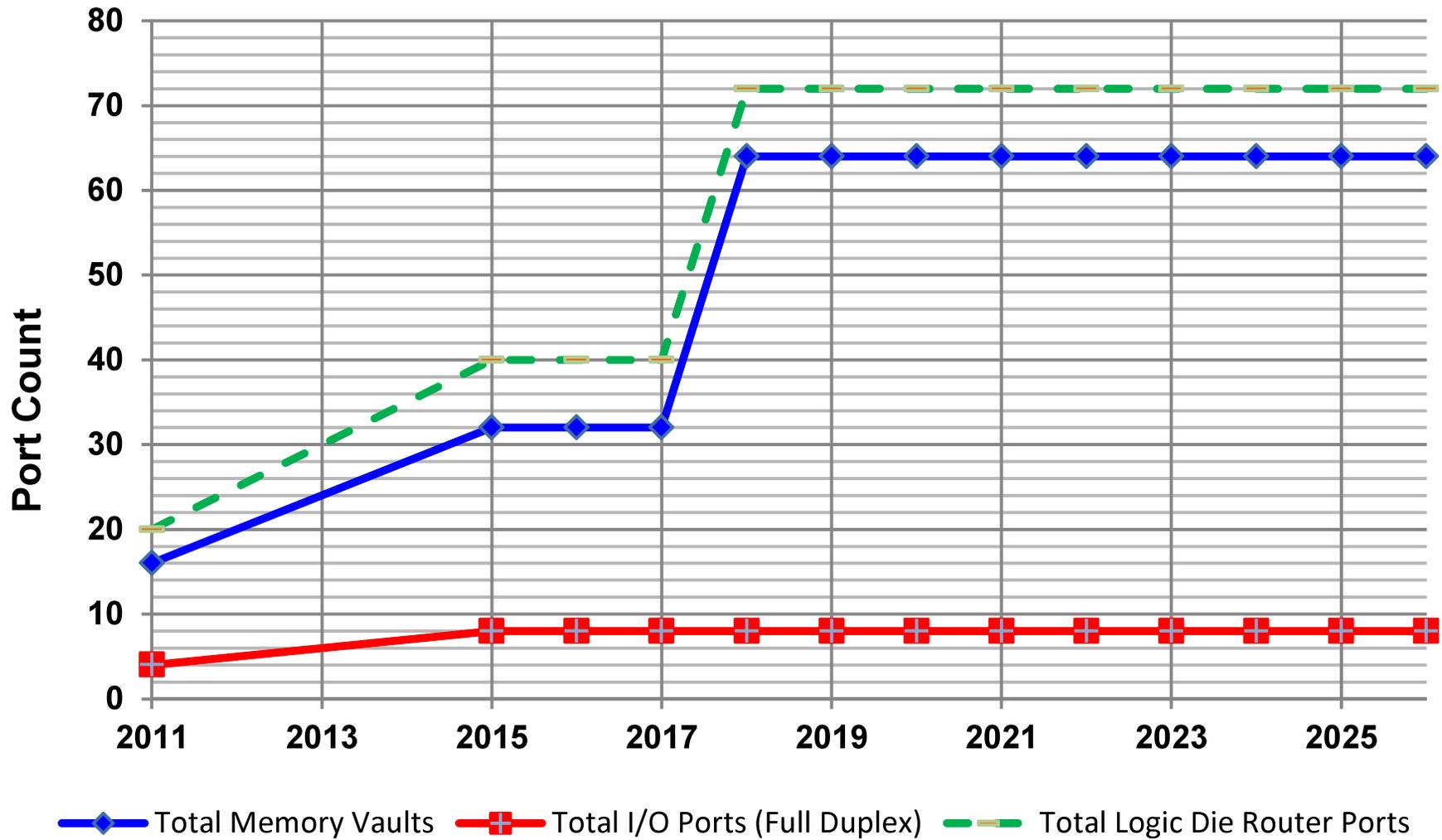


Remember: Stack takes ~ area of a single DRAM die.

Leading edge DDR DIMMs have up to 128 die.

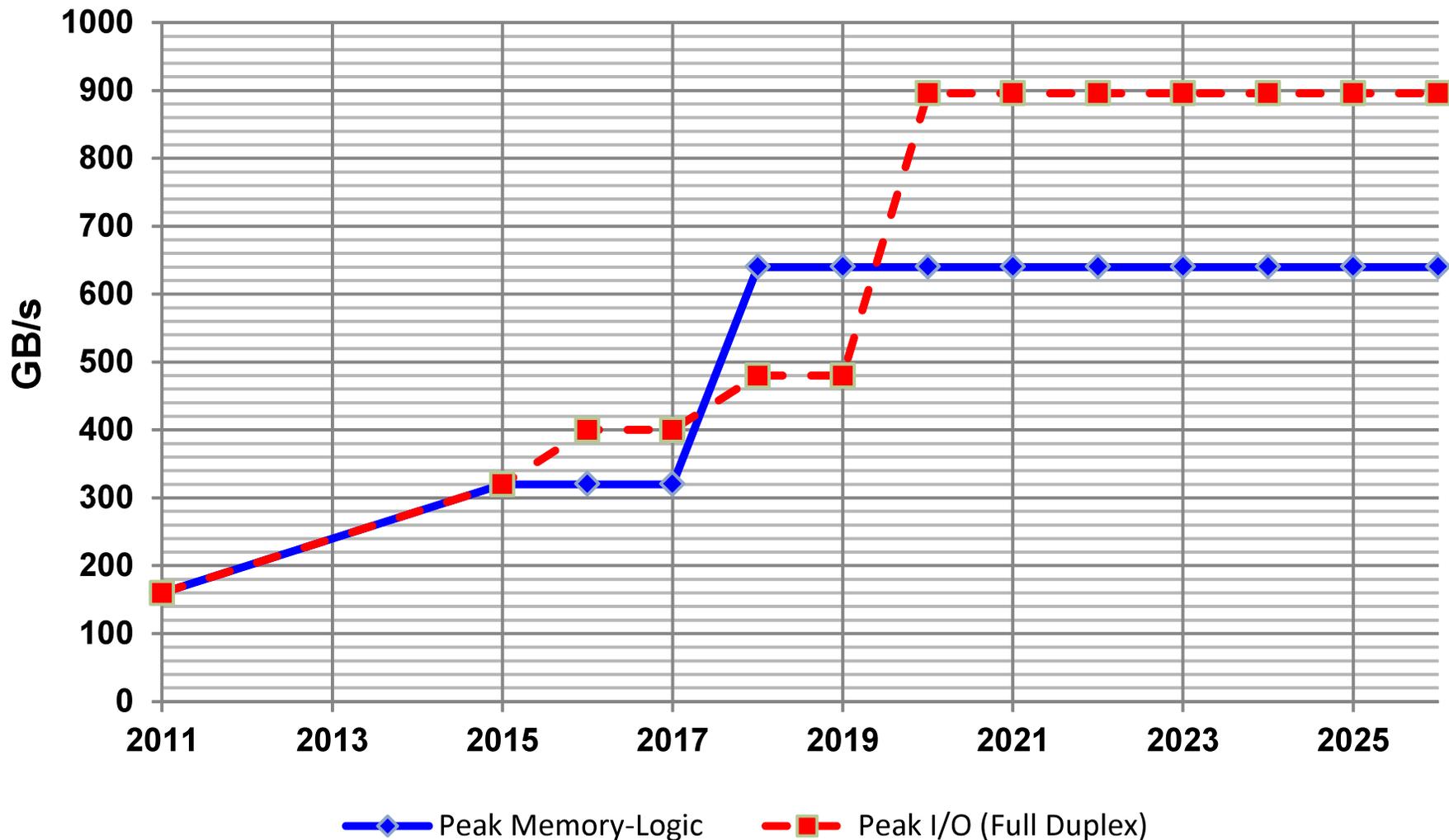


How Many Vaults, Ports per Stack



Projections made on basis of ITRS and other public data

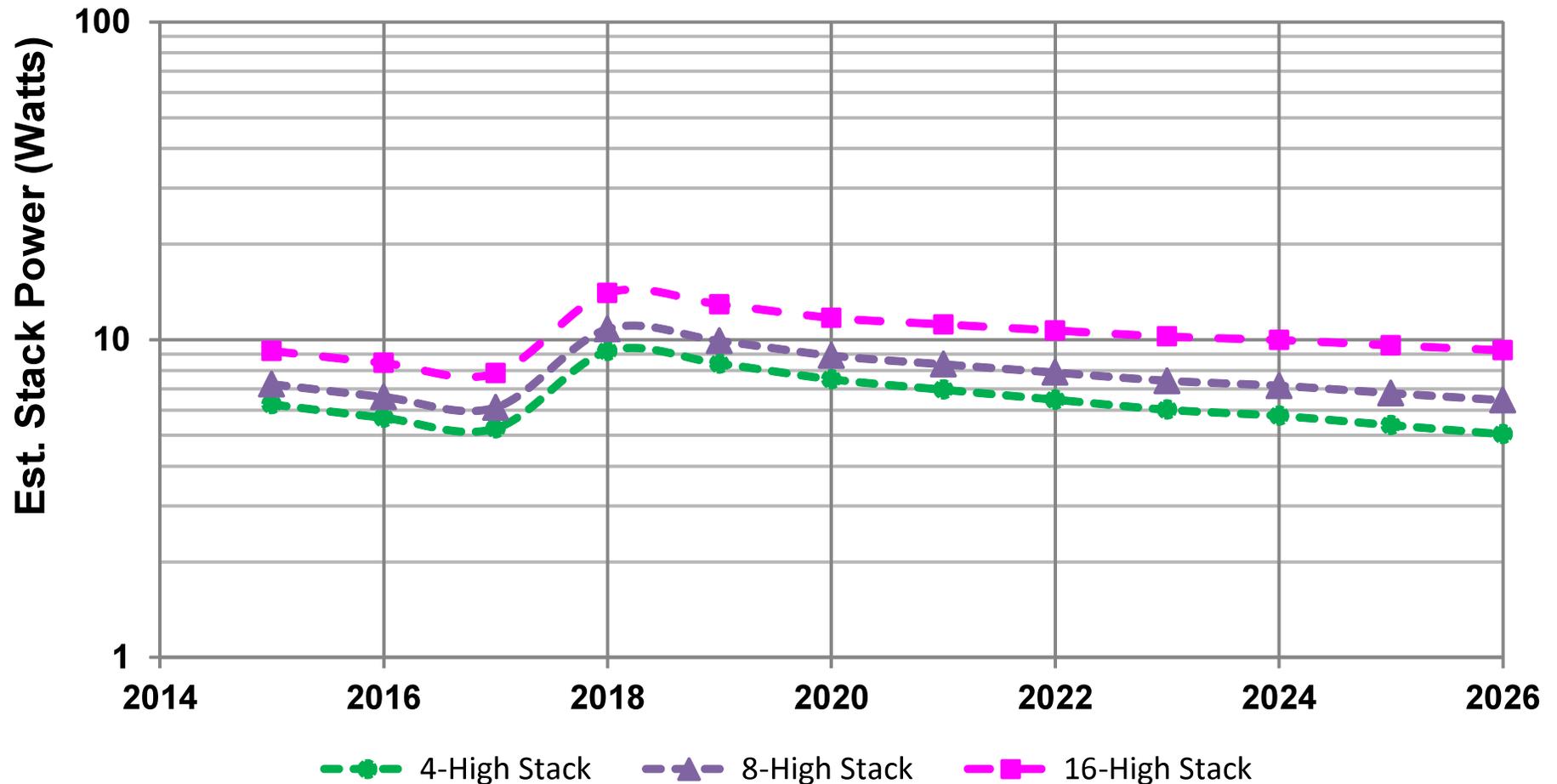
What Might be the Bandwidth/Stack



Projections made on basis of ITRS and other public data

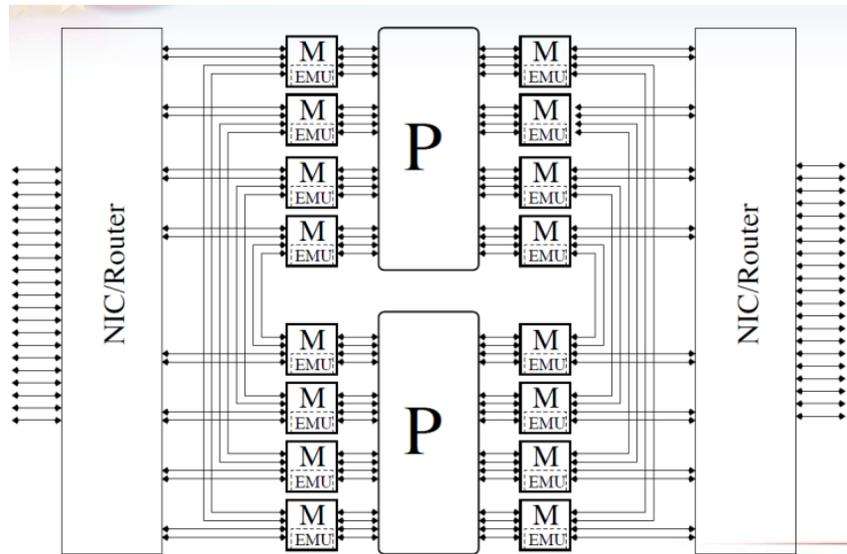


Memory Stack Power Estimates

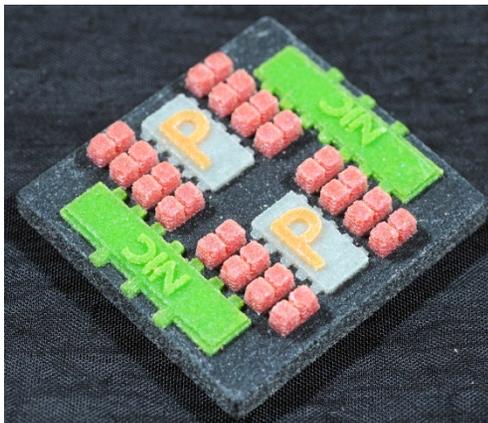


Projections made on basis of ITRS and other public data

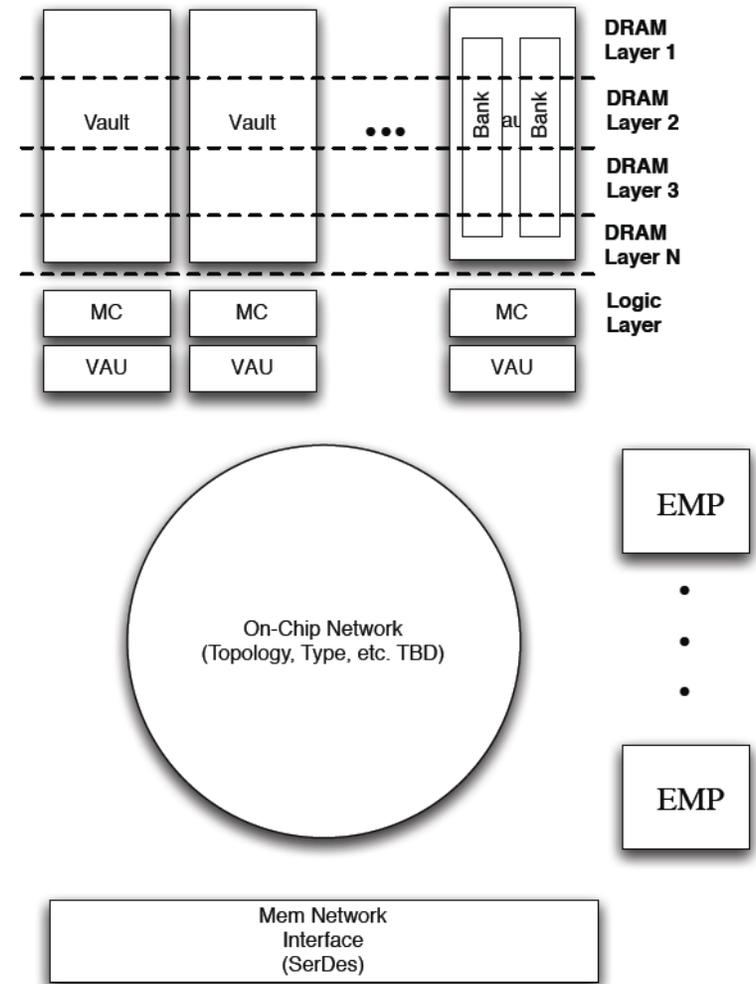
SNL Xcaliber Architecture



(a) X-caliber Node Architecture



(b) X-caliber Node Mockup



(c) X-caliber stack notional architecture



What's In an Exascale Address?

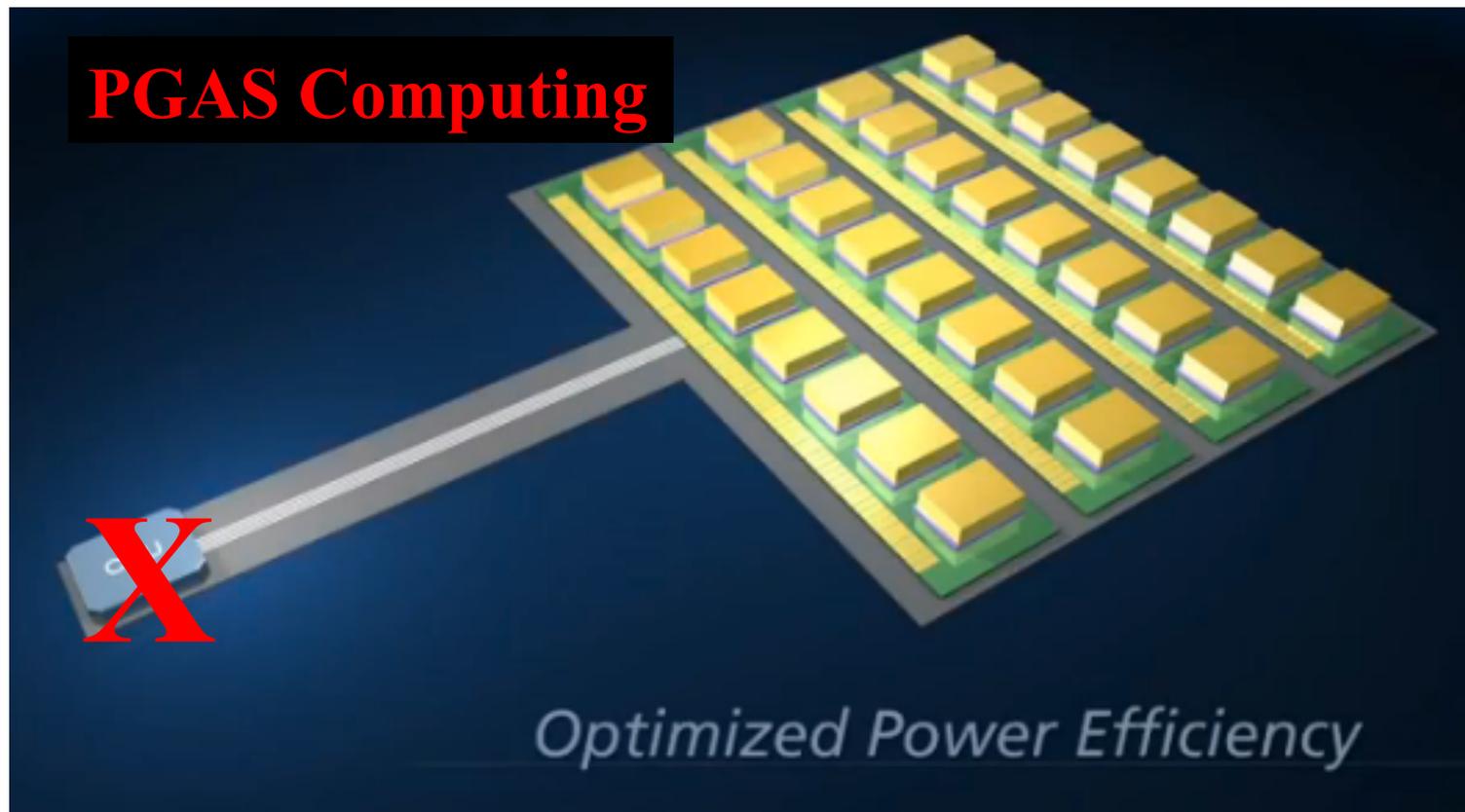
1. Which Node
2. Which Socket
3. Which Channel
4. Which "DIMM"
5. Which Stack
6. Which Vault
7. Which Bank set
8. Which Bank
9. Which Block
10. Which Row
11. Which Word

**Optimal Data Placement
Now a
11-Dimensional Problem
And this doesn't include
Cache Hierarchy**



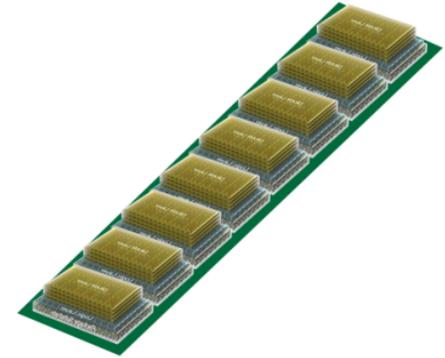
But ...

- What if we add cores to the logic die on each stack?
- Now – opportunity for real PGAS architecture ***in the small!***

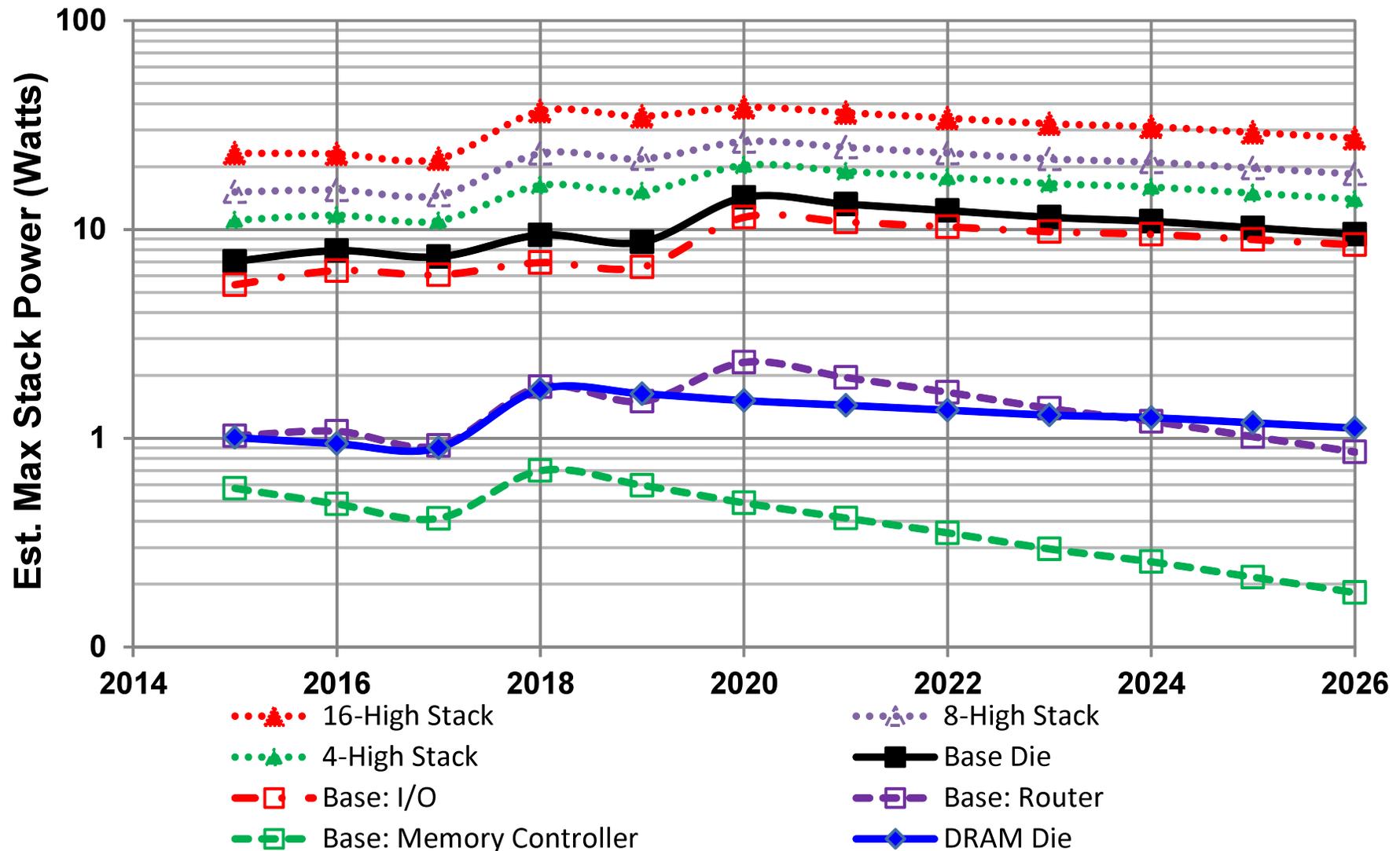


Thought Experiment: Memory Stack Only Version

- Same stack as from X-caliber
 - Multiple DRAM, NVRAM vaults
 - Internal crossbar for full interconnect
 - 8 external ports (still wire)
- Multiple stacks on something like a DIMM
- Remove Processor sockets and NIC chips
- Use stack external ports for all routing
- Keep routing on global address
- And grow up logic chip processing
 - “Conventional Core” per vault

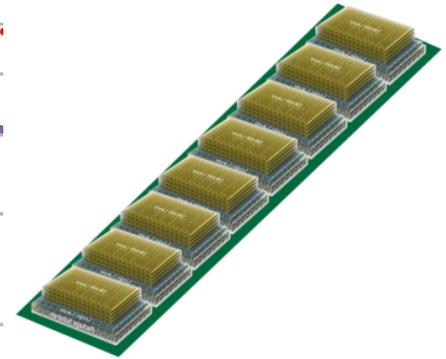
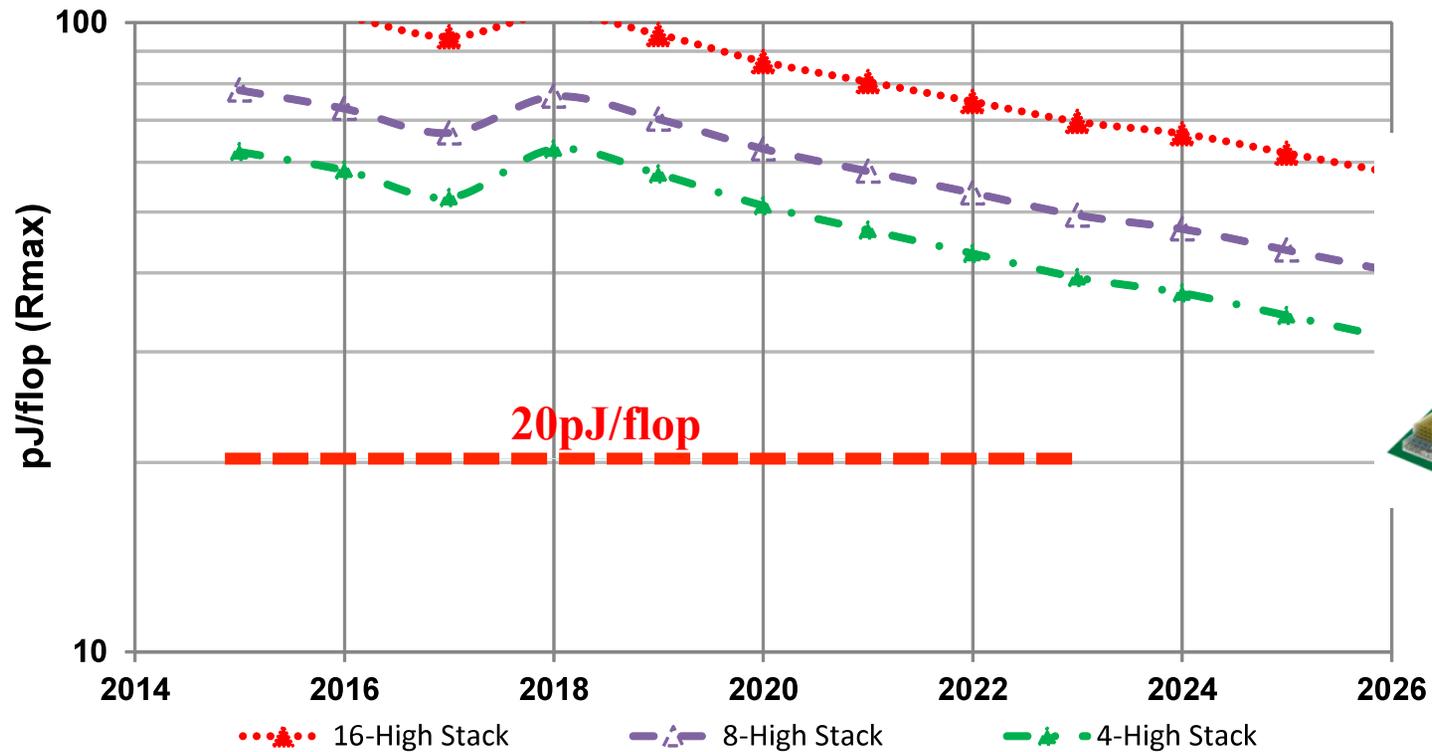


Memory + Processing



Projections made on basis of ITRS and other public data

Energy/Flop Extrapolation



We can see the Goal!

Projections made on basis of ITRS and other public data

Conclusions

- Memory is essential for computing
- But rapidly becoming severe limitation
- Limitations stem from architecture that separates memory from computation
- PIM: attempt to overcome
- 3D stacks will enable massive “Processing Near Memory”

**That have a shot at
useable extreme scale**

