Reading the Tea-Leaves: How Architecture Will Evolve

Peter M. Kogge
McCourtney Prof. of CS & Engr.
University of Notre Dame
IBM Fellow (retired)

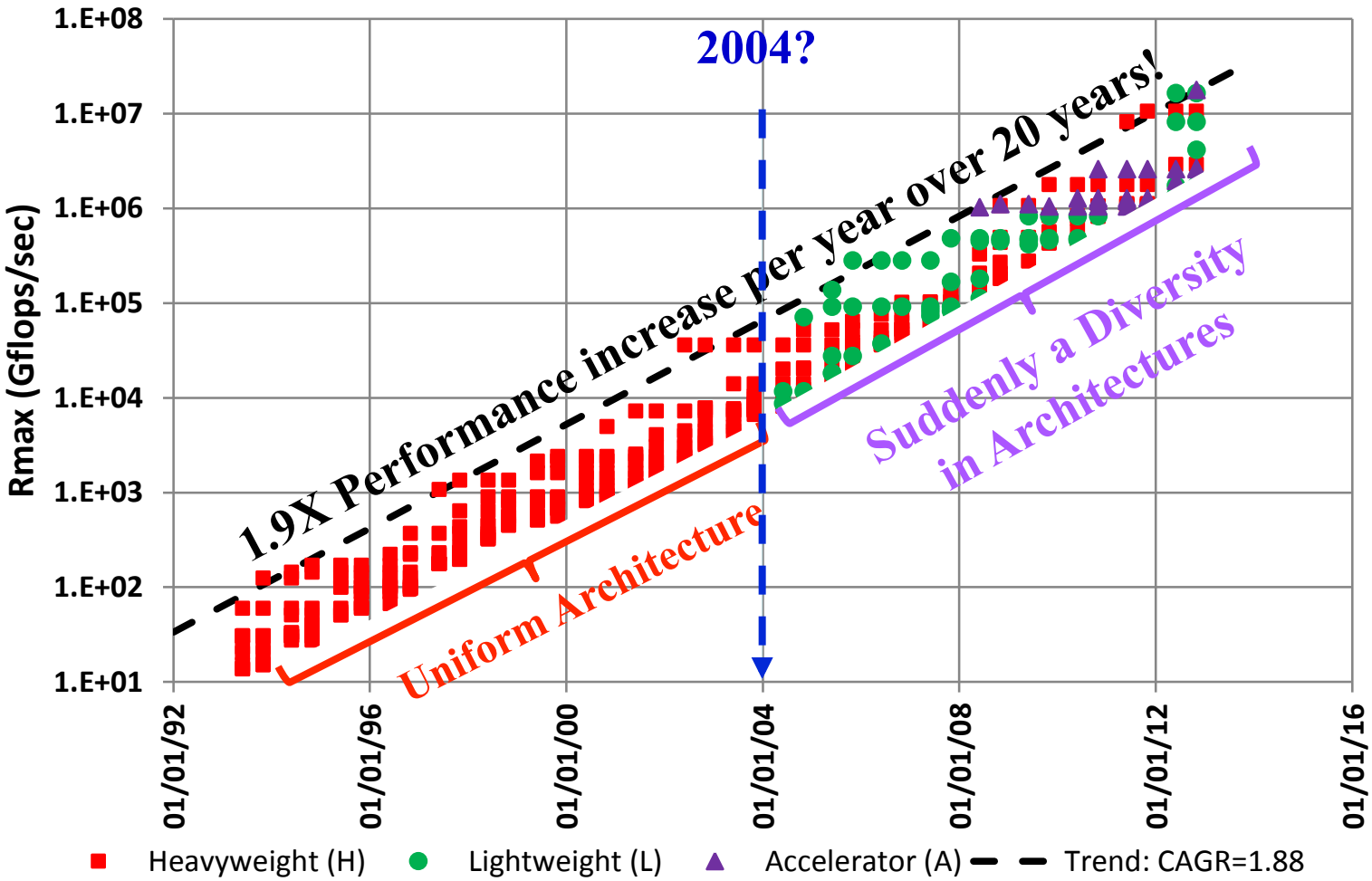
Acknowledgement: This work was funded in part by the US Dept. of Energy, Sandia National Labs, as part of their XGC project.

Topics

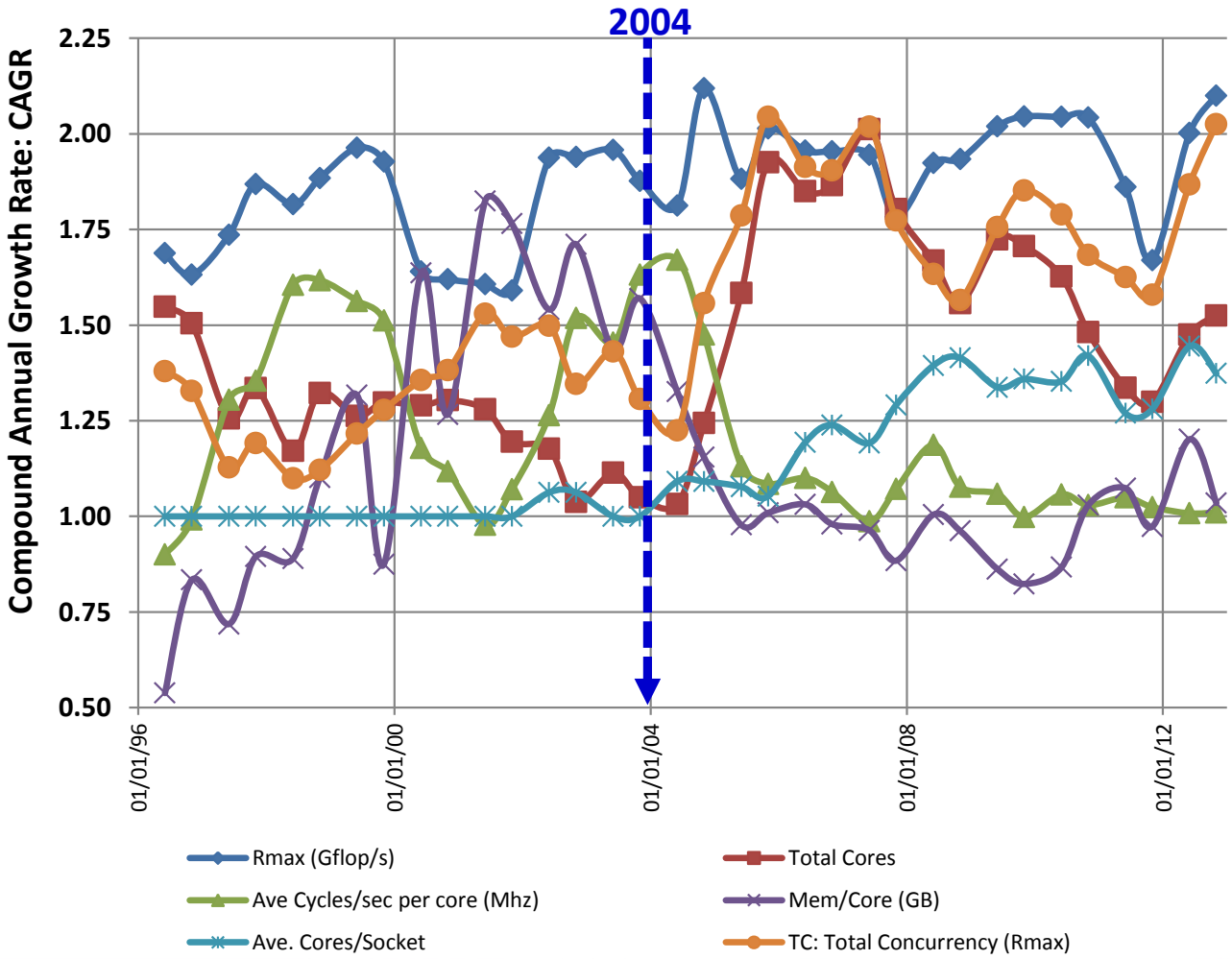
- 2004: Technology Ceilings Force Architecture Changes
- Our Current Architecture Spectrum
- TOP500 Properties
- GRAPH500 Properties
- An Emerging Alternative: 3D

2004: Architecture's Perfect Storm

We All Know The Story: Unbroken Growth in Rmax



But When We Look Deeper Something *Did* Happen in 2004!



Flops/cycle increased even faster

Total Cores increased even faster

Cores/socket increased

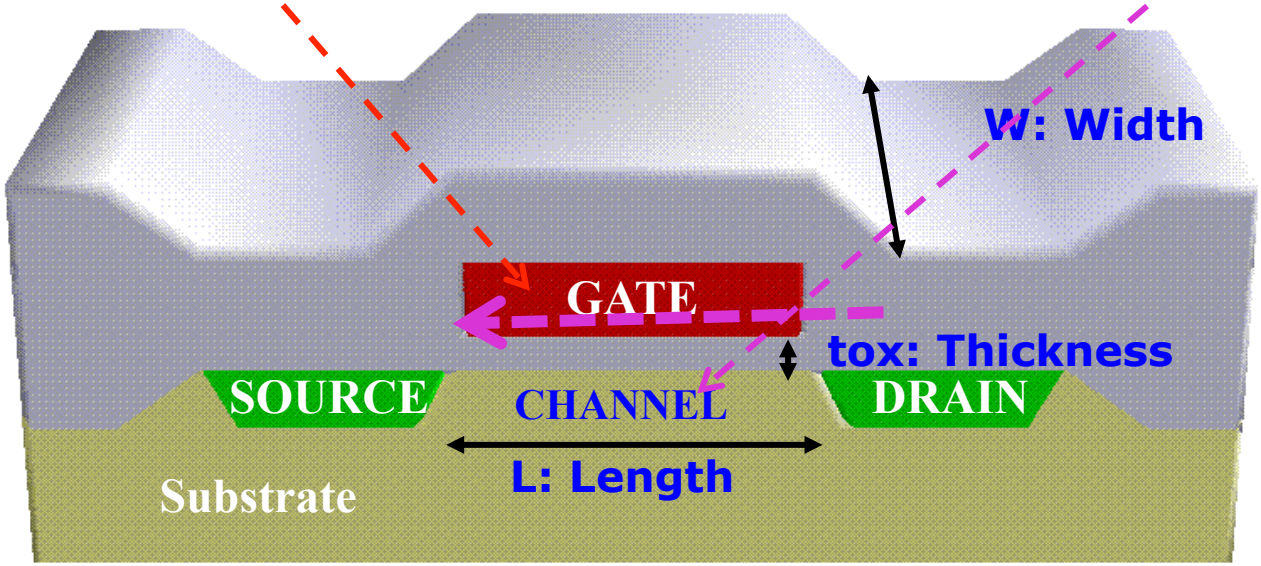
Memory/core went flat

Clock rates went flat

Let's Start with a CMOS Transistor

A voltage here (relative to substrate)

Enables a current here

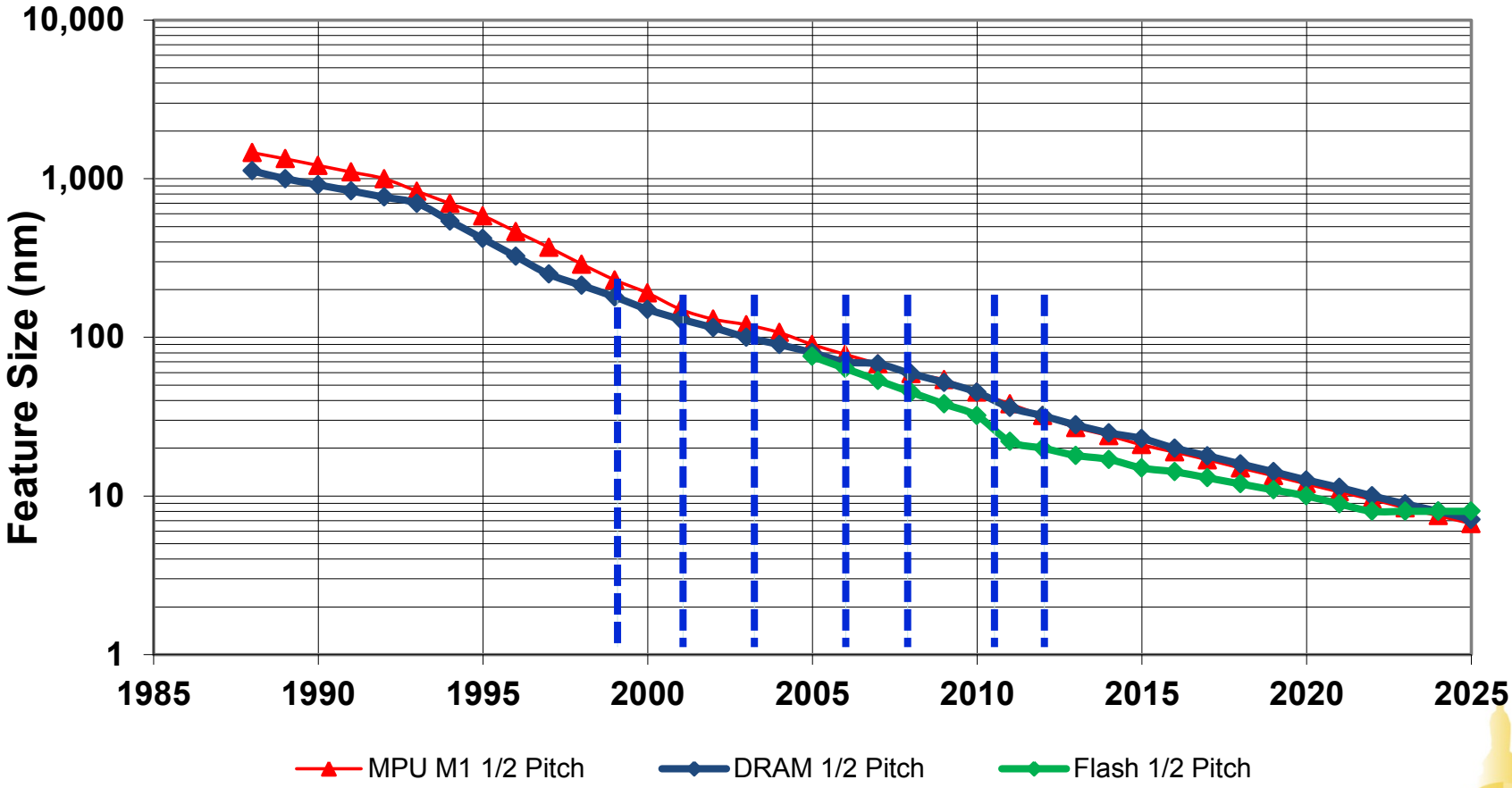


Key physical parameters:

- Length (L): how far do electrons have to go (longer reduces flow)
- Width (W): how many "lanes" are there for electrons (wider increases flow)
- Thickness (tox): how far is gate from substrate (thicker requires higher turn-on voltage)

"Feature Size:" related to minimum linear dimension - Channel

Moore's Law Feature Size



Technology Generation: Feature Size Reduction by "S" ~ √2

Evolution in Moore's Law

- **1965**: # of transistors on chip doubles every year
- **1975**: # per unit area doubles every 1.5 – 2 years
  - Corresponds to **S = √2** per "Generation"
- **Late 1990s**: single core processor "performance" grows ***more than*** S every generation
  - If we keep chip size constant, more room for
    - More transistors in caches
    - More complex cores for higher IPC
  - At same time transistor delay goes down by 1/S
    - Clock rate can increase by S!
- **Now**: back to just growth in transistor density

Why?

The 2004 "Perfect Storm"

2004

1970s

2020

Today

Moore's Law: transistor size and intrinsic delay continue to decrease

Single core microprocessors:
- more capable & faster
- power increase offset by lower voltages

Memory: more memory/chip
- Due to density increase & bigger chips
- Memory latency improves slowly

Interconnect: tracks clock
- Wire driven

The rise of multi-core:
- More, simpler, cores per die
- Slower clocks
- Relatively constant off-die bandwidth

Memory:
- Slow density increase
- Slow grow in off-memory bandwidth

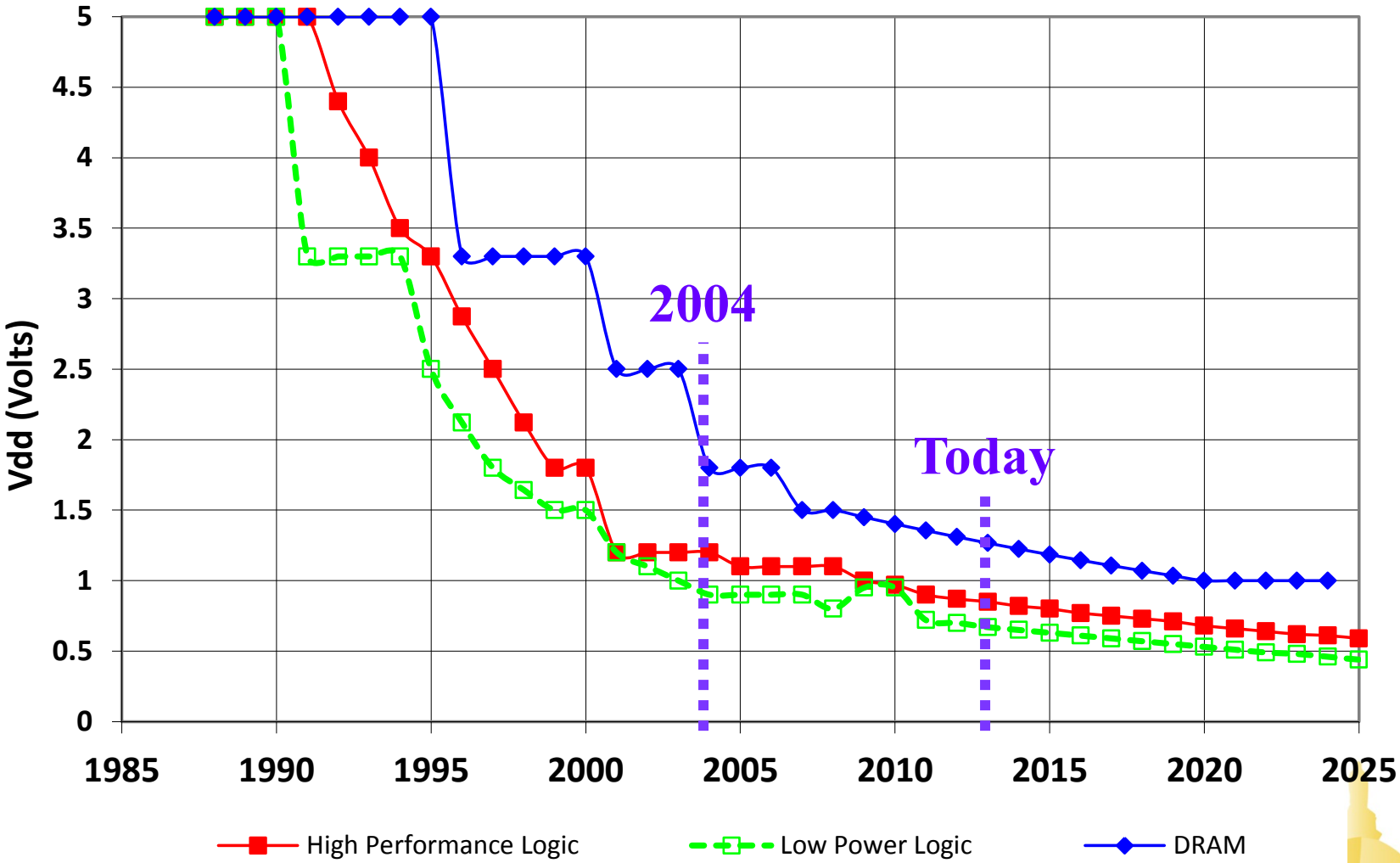
Interconnect:
- Complex, power consuming wire
- Very complex fiber optics

- Operating Voltage stops decreasing
- Chip power exceeds inexpensive cooling
- No more performance gains from most complex cores
- Off-chip I/O maxes out
- Economics of DRAM inhibits bigger chips
- Wire interconnect peaks

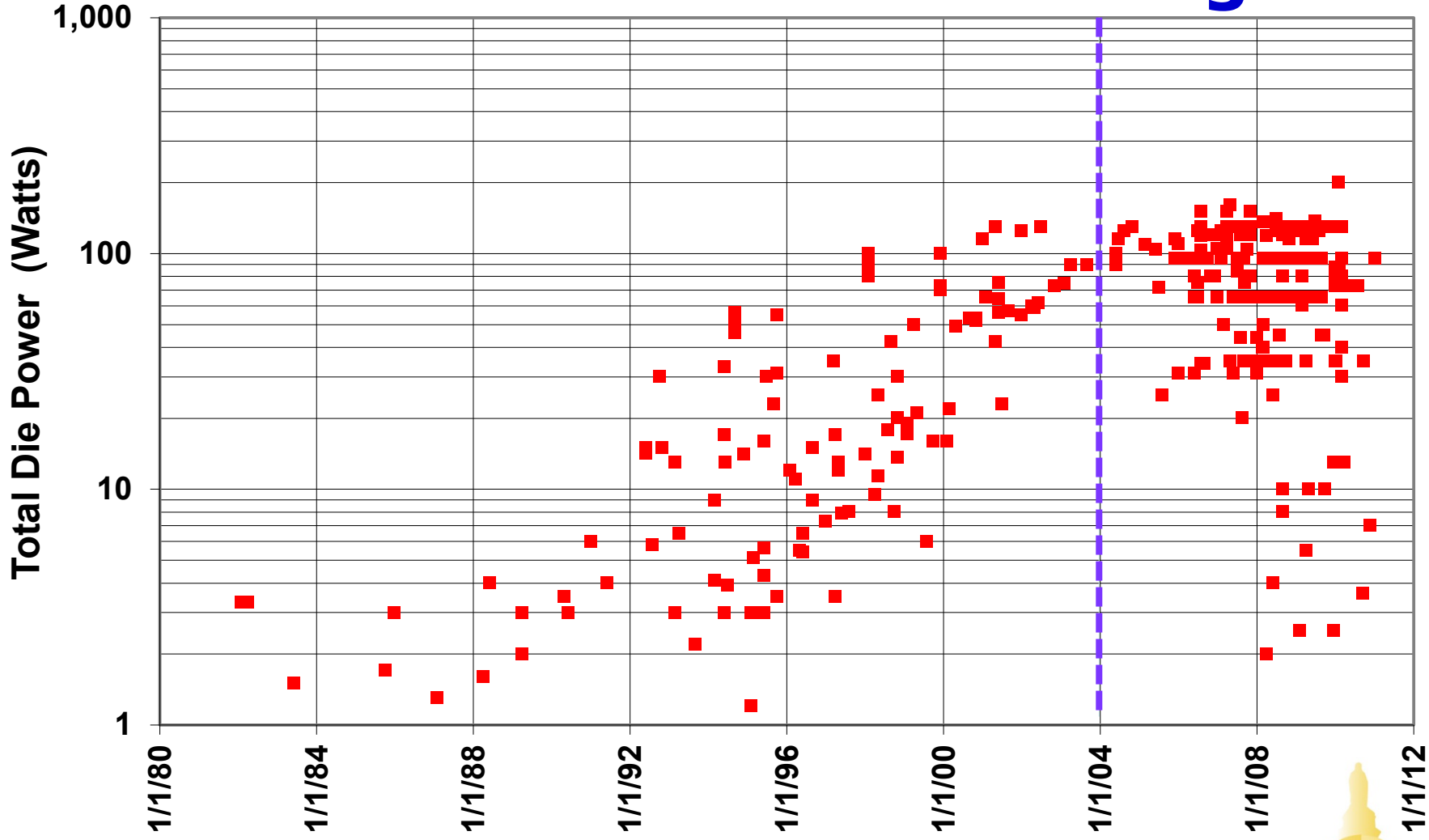
Power: *THE* 1st Design Constraint

- Power in CMOS circuit: αCFV^2
  - αC: average capacitance switched per clock cycle; Scales ≈1/S
  - F: clock rate; Intrinsic peak scales as S
  - V: operating voltage: This is the problem!
- αCF essentially flat for different generations of same circuit
- But we can pack S^2 more circuits on same chip size
- Thus power: $S^2 V^2$
- Before 2004, V declined, reducing increase in power
- After 2004, V has flattened; power can *run wild*!
  - And better cooling is too expensive

Operating Voltage

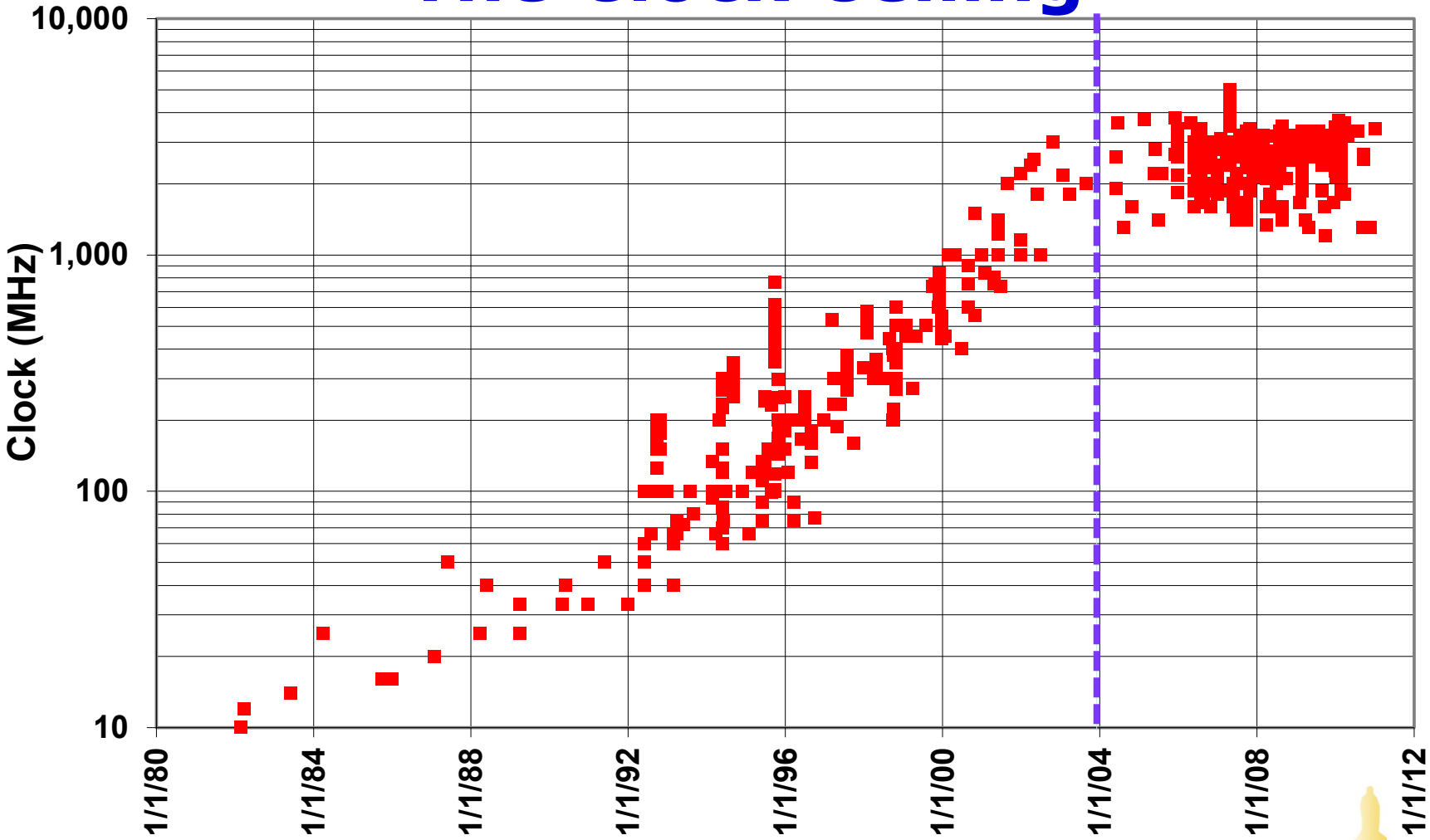


We've Hit a Power Ceiling



data from www.cpudb.stanford.edu

The Clock Ceiling



data from www.cpudb.stanford.edu

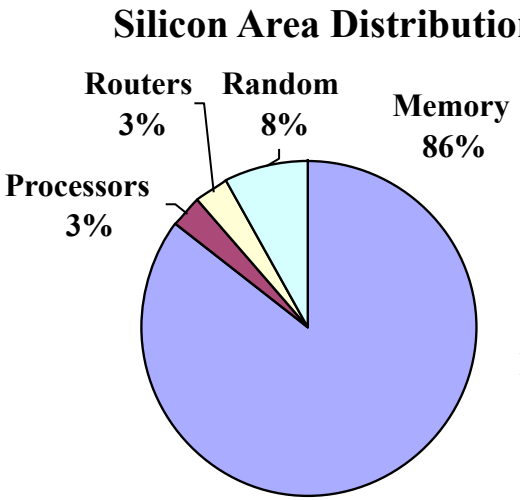
Today's Architectures

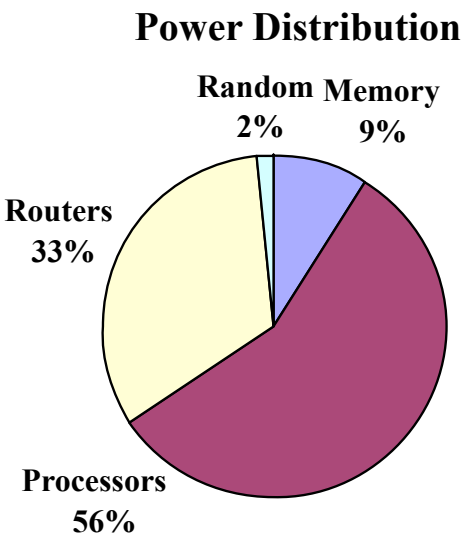
Architecture Classes

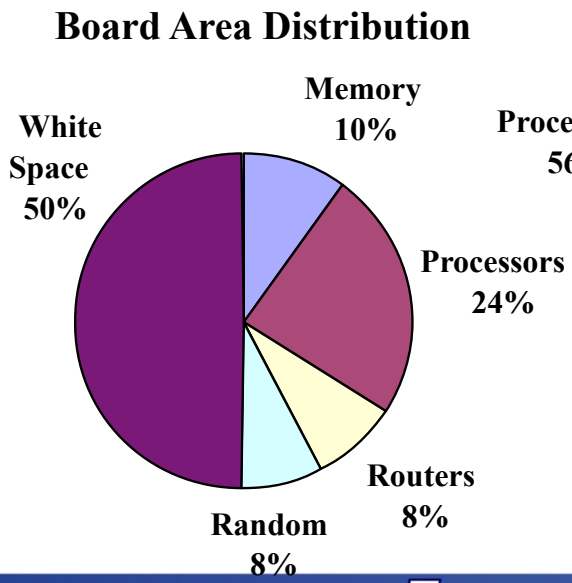
- **Heavyweight**: traditional 100+W multi-core
  - Often requires supporting chip set
- **Lightweight**: lower power single chip system
  - Lower performance but denser packaging
- GPU: specialized high FPU multicore socket
  - Also very high power
- **Hybrid/Heterogeneous**: Heavyweight/GPU combination

- Two **Other** architectures important to GRAPH500
  - XMT: massive shared memory multithreaded
  - Convey: FPGA based

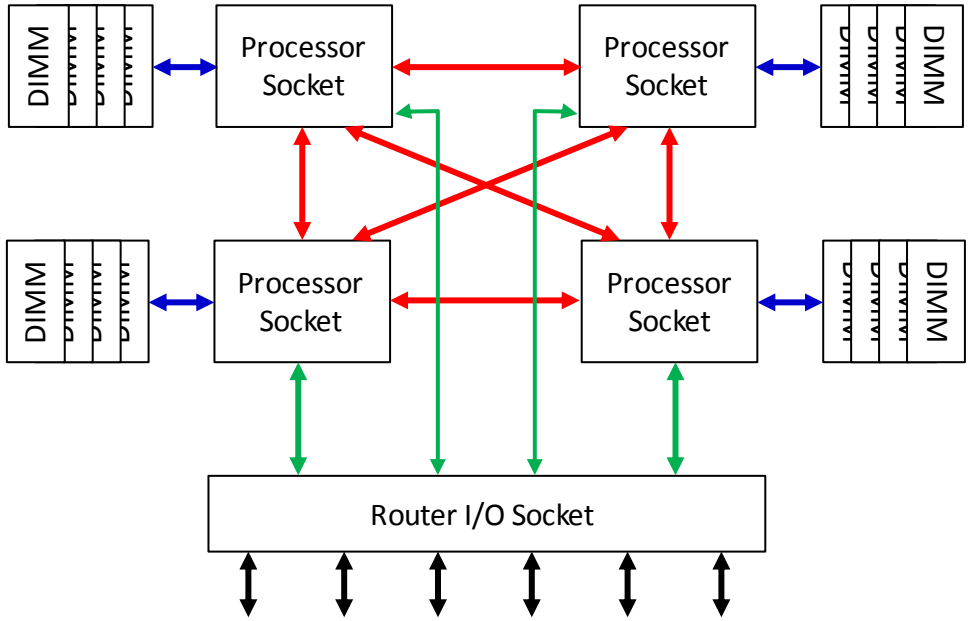
Classical *Heavyweight* Architectures







Today's Heavyweight

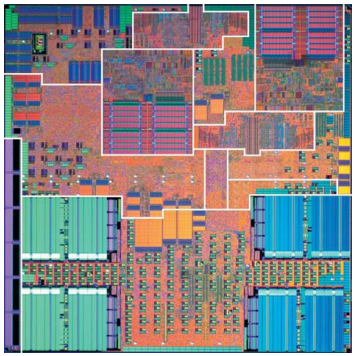


A Power 7 Drawer

LightWeight Architectures Starting with BlueGene L

System Architecture:

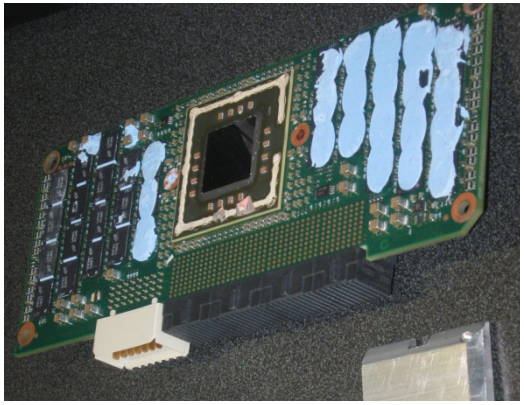
- Multiple Identical Boards/Rack
- Each board holds multiple Compute Cards
- "Nothing Else"

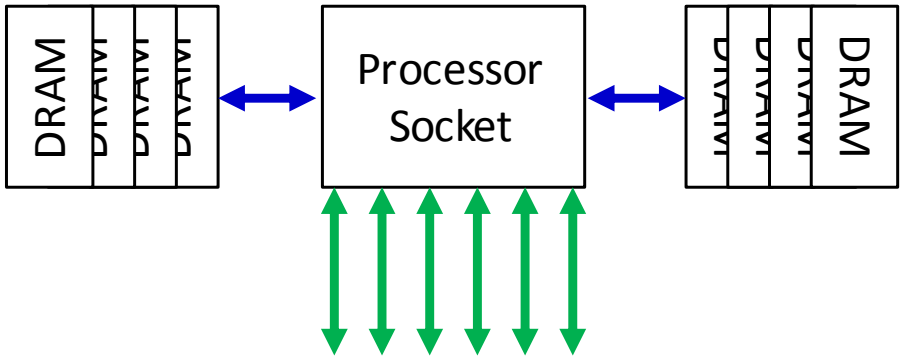


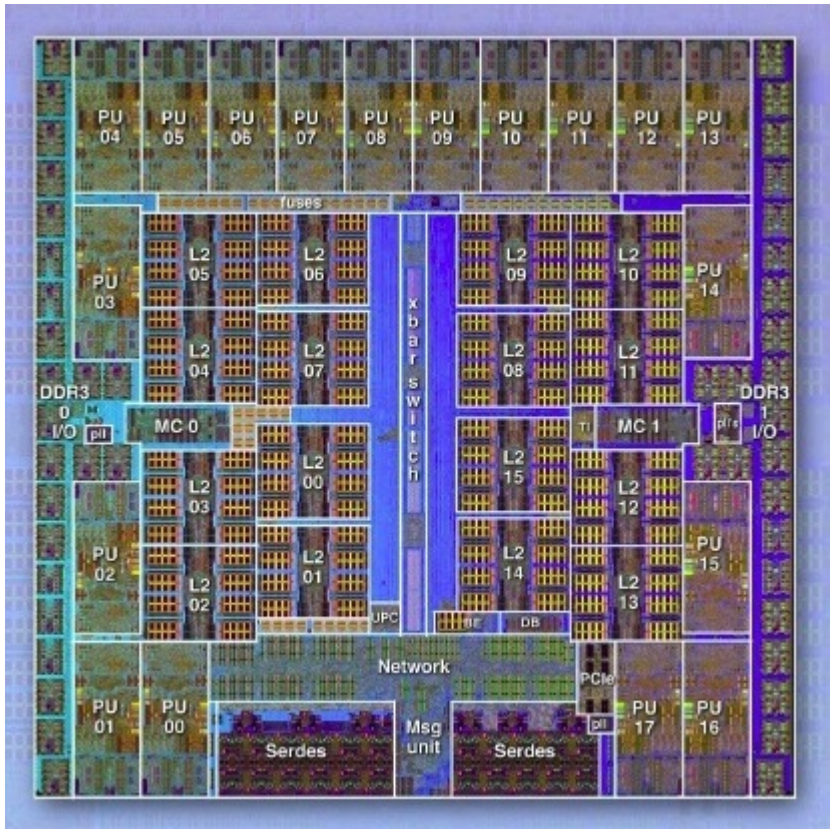
- 2 simple dual issue cores
- Each with dual FPUs
- Memory controller
- Large eDRAM L3
- 3D message interface
- Collective interface
- All at subGHz clock

"Packaging the Blue Gene/L supercomputer," IBM J. R&D, March/May 2005
"Blue Gene/L compute chip: Synthesis, timing, and physical design," IBM J. R&D, March/May 2005

BlueGene/Q





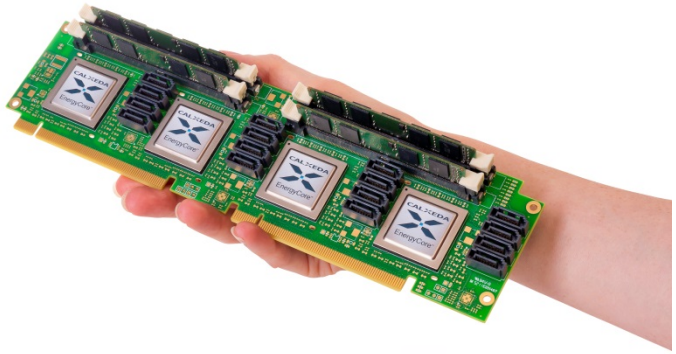


Integrated

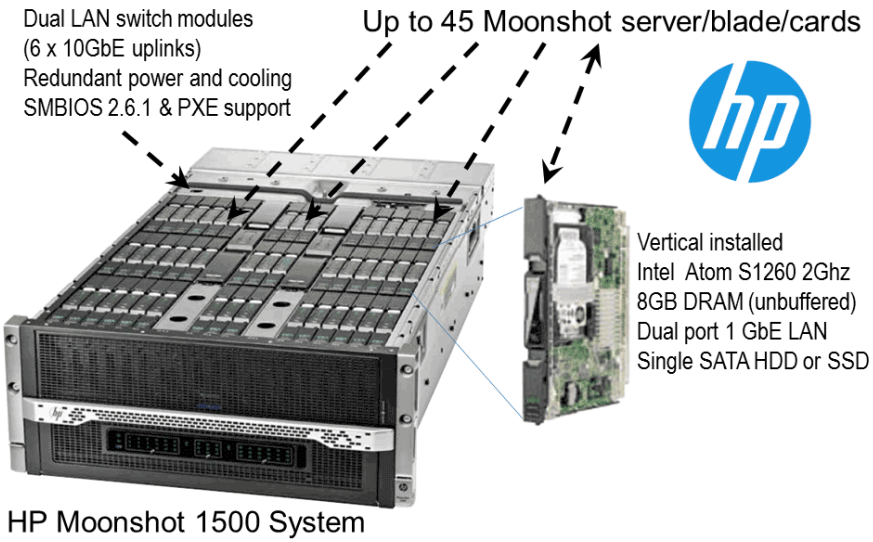
- **NIC**
- **Memory controllers**

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

Other Lightweight Systems Emerging



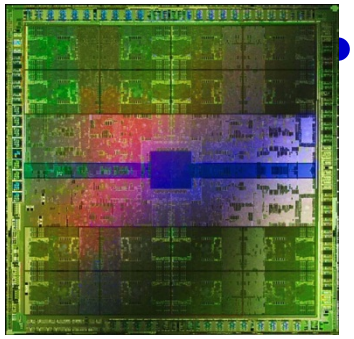
Calxeda quad-socket, quad-core ARMs



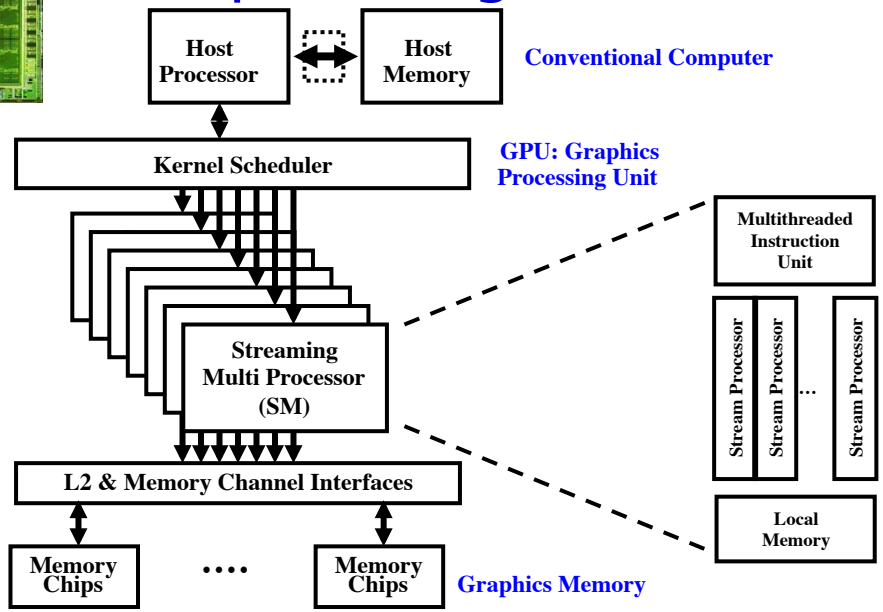
HP Moonshot

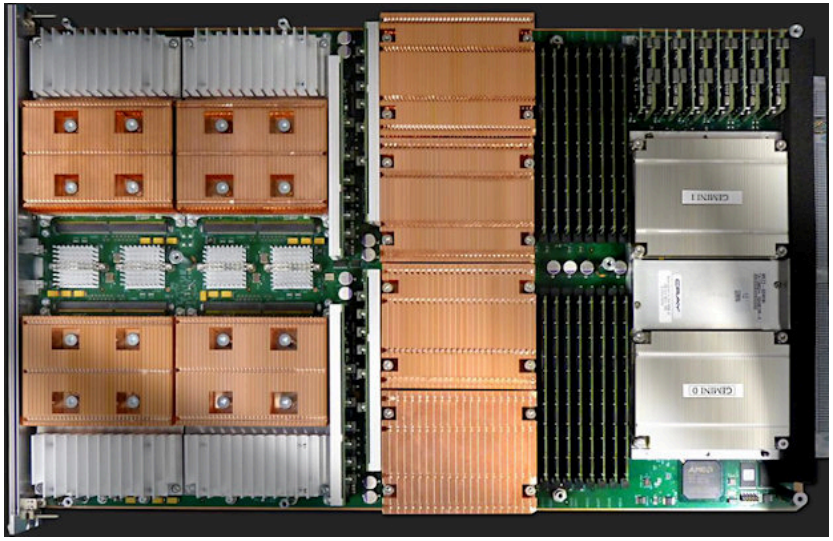
Heterogeneous Architectures

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

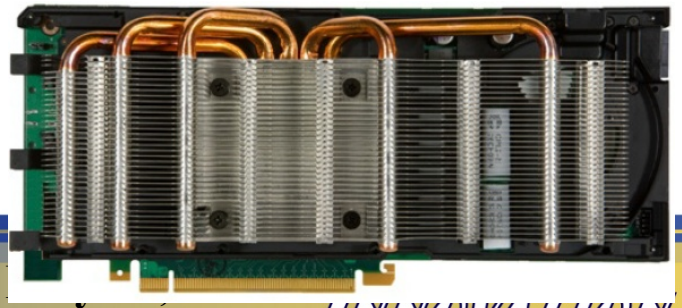


- Mix of heavyweight masters and GPU compute engines



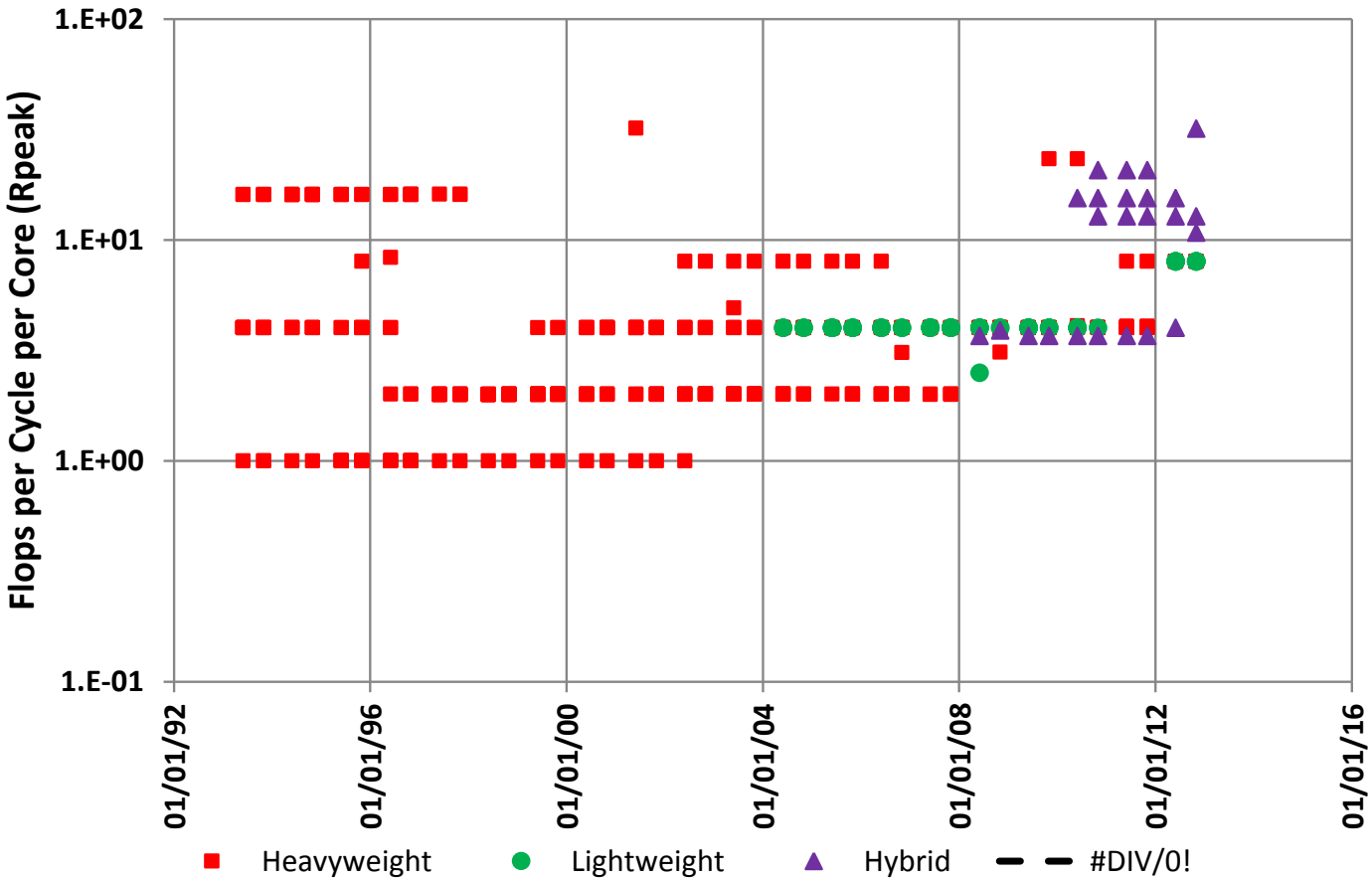


A Titan Blade



http://www.nvidia.com/docs/IO/43395/BD-05238-001_v03.pdf

What's the Difference?



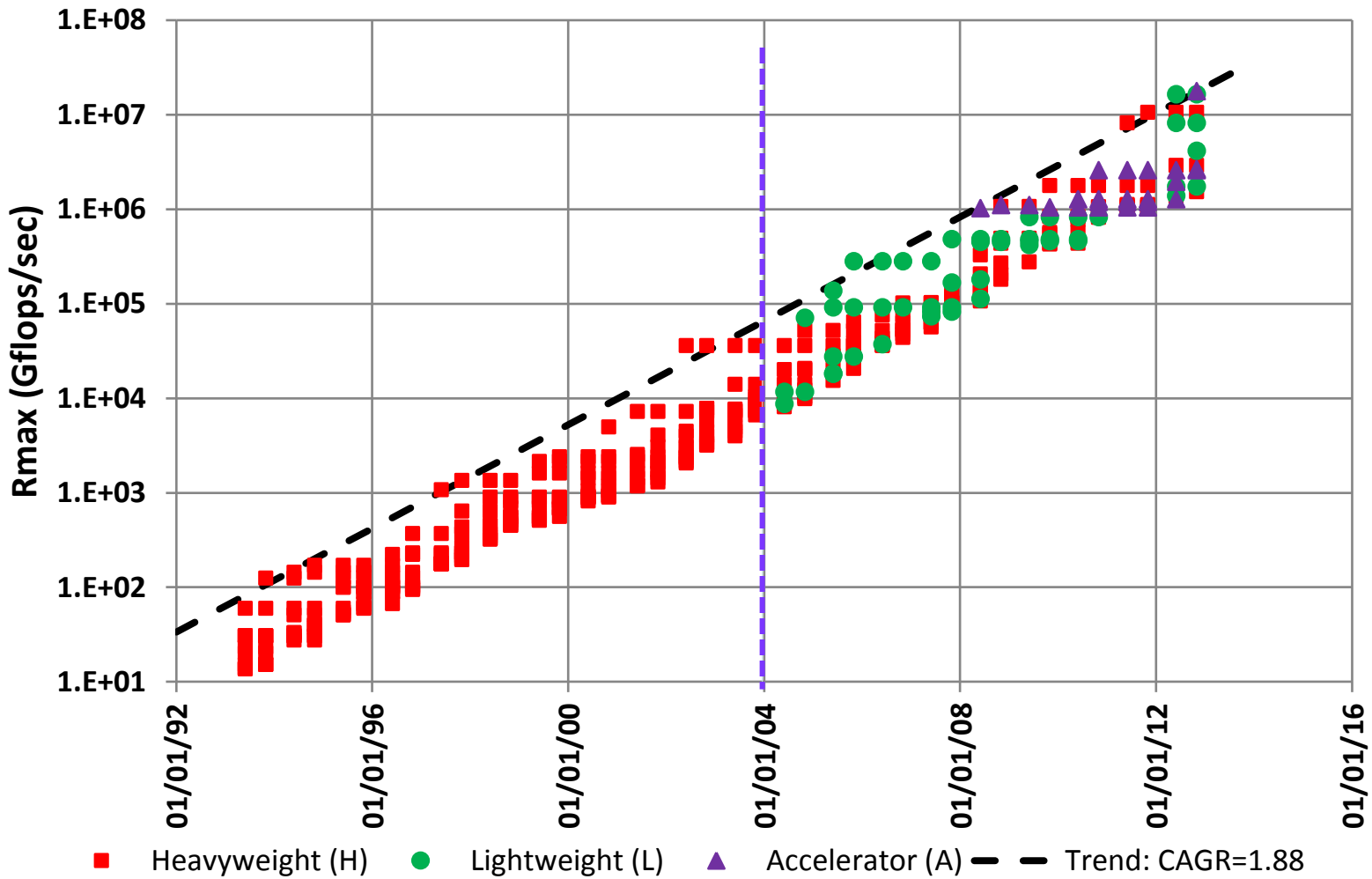
Comparison against Heavyweight

Lightweight: same flops/cycle; lower clock but more cores per rack

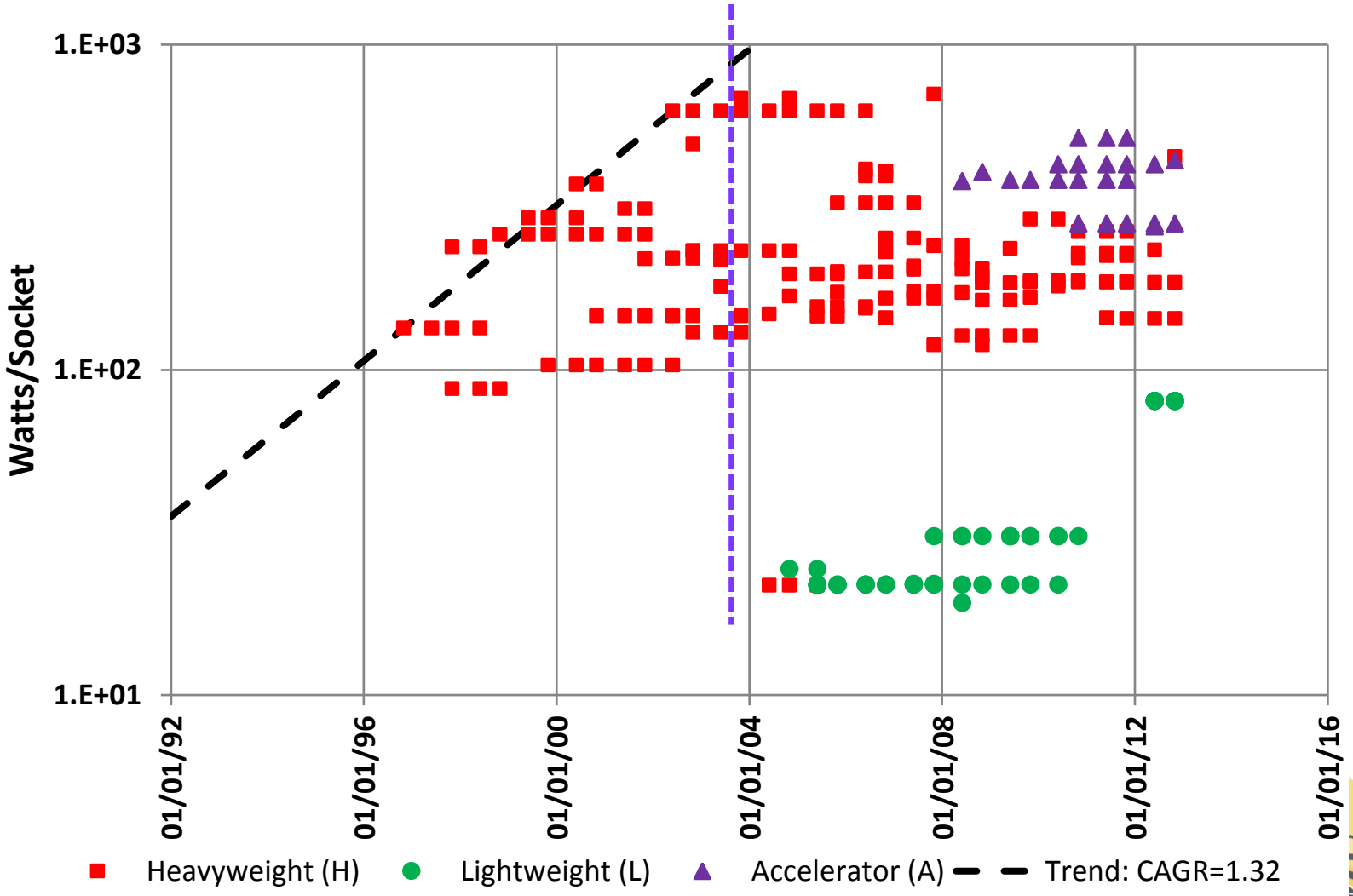
Hetergeneous: many more flops/cycle per core; many more cores

TOP500 Details
(Update to Exascale Report)

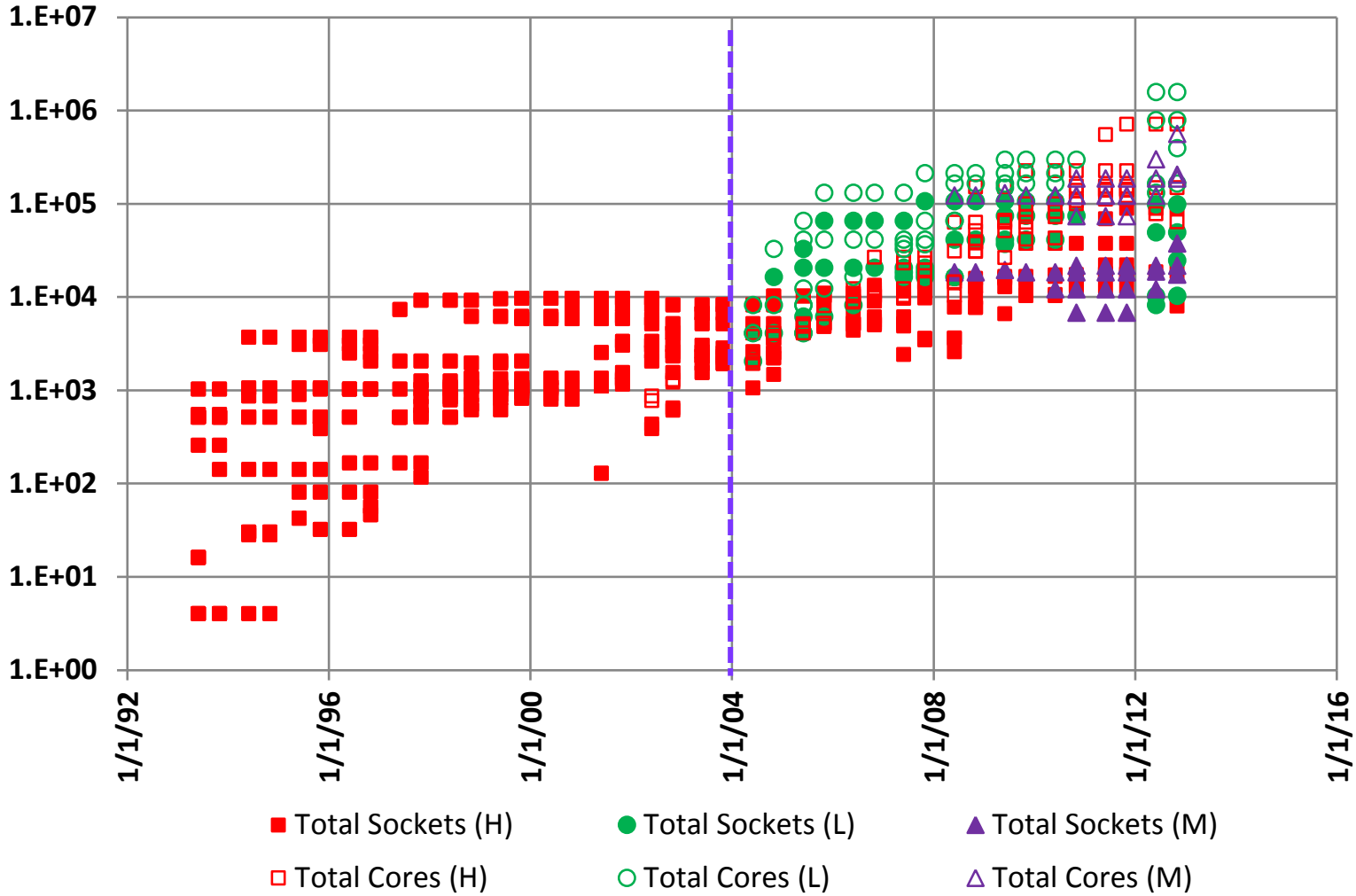
Emergence of Lightweight, Hybrid



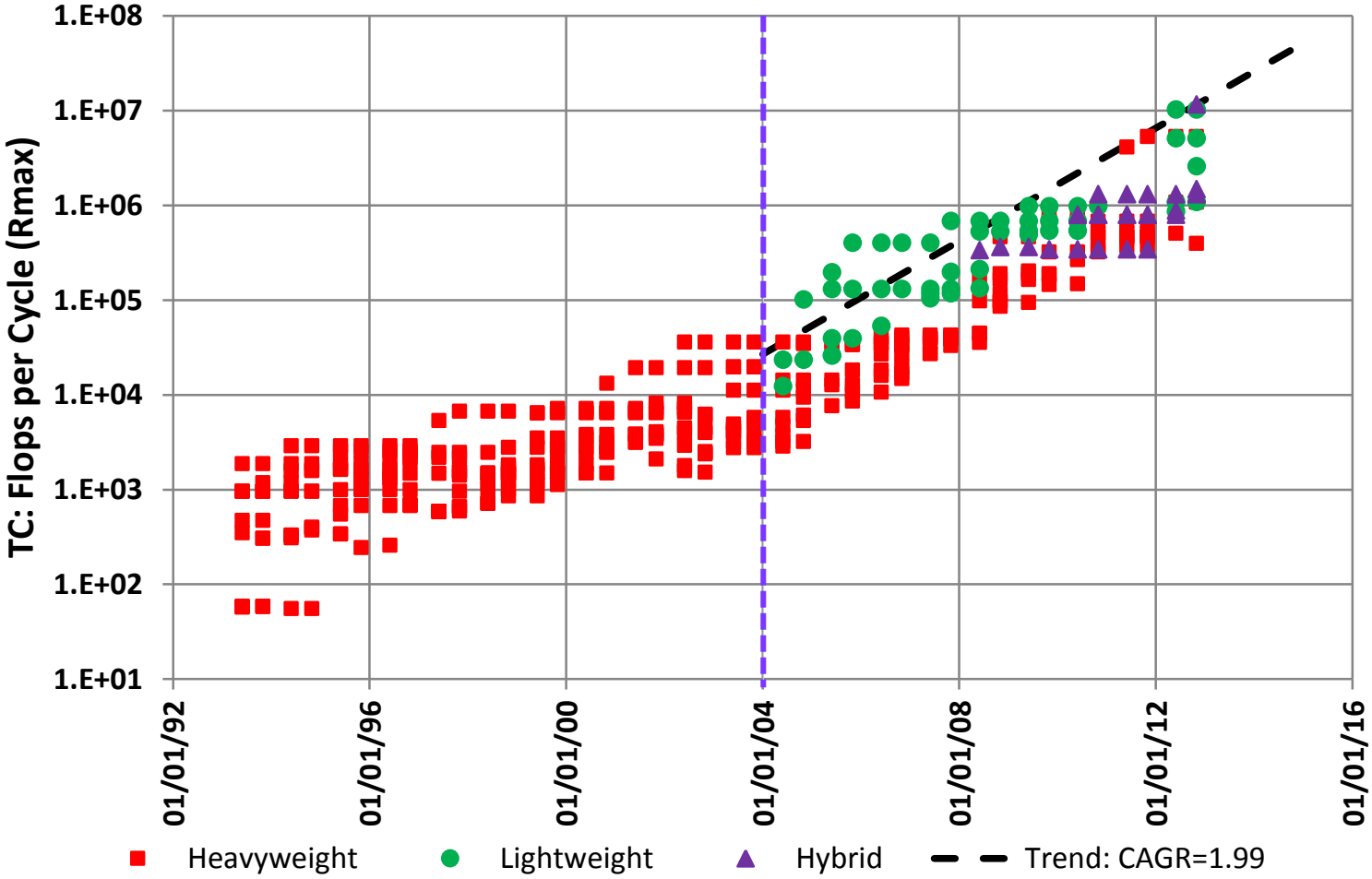
Ceiling on System Power "per Socket"



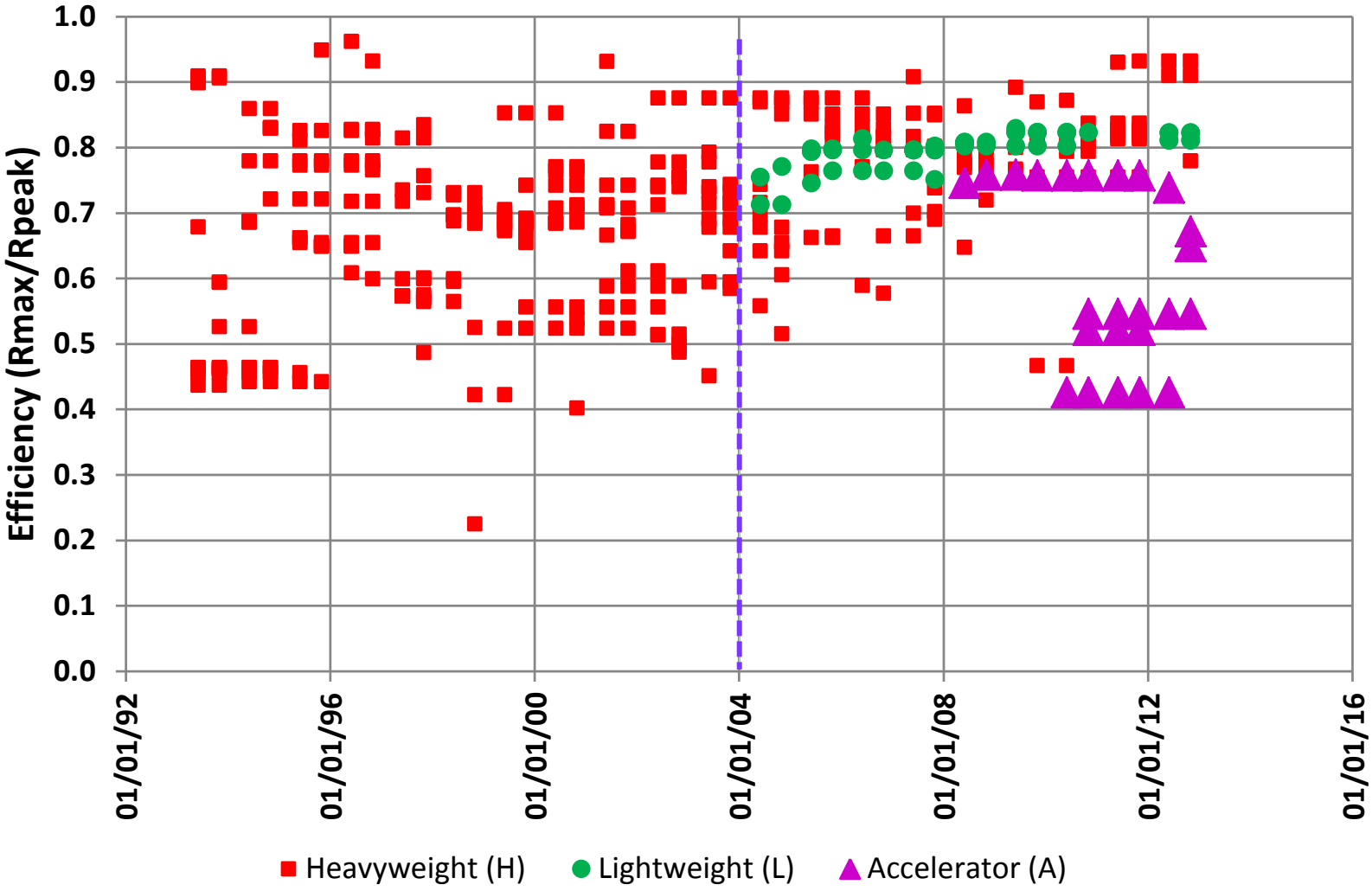
Sockets and Cores Growing



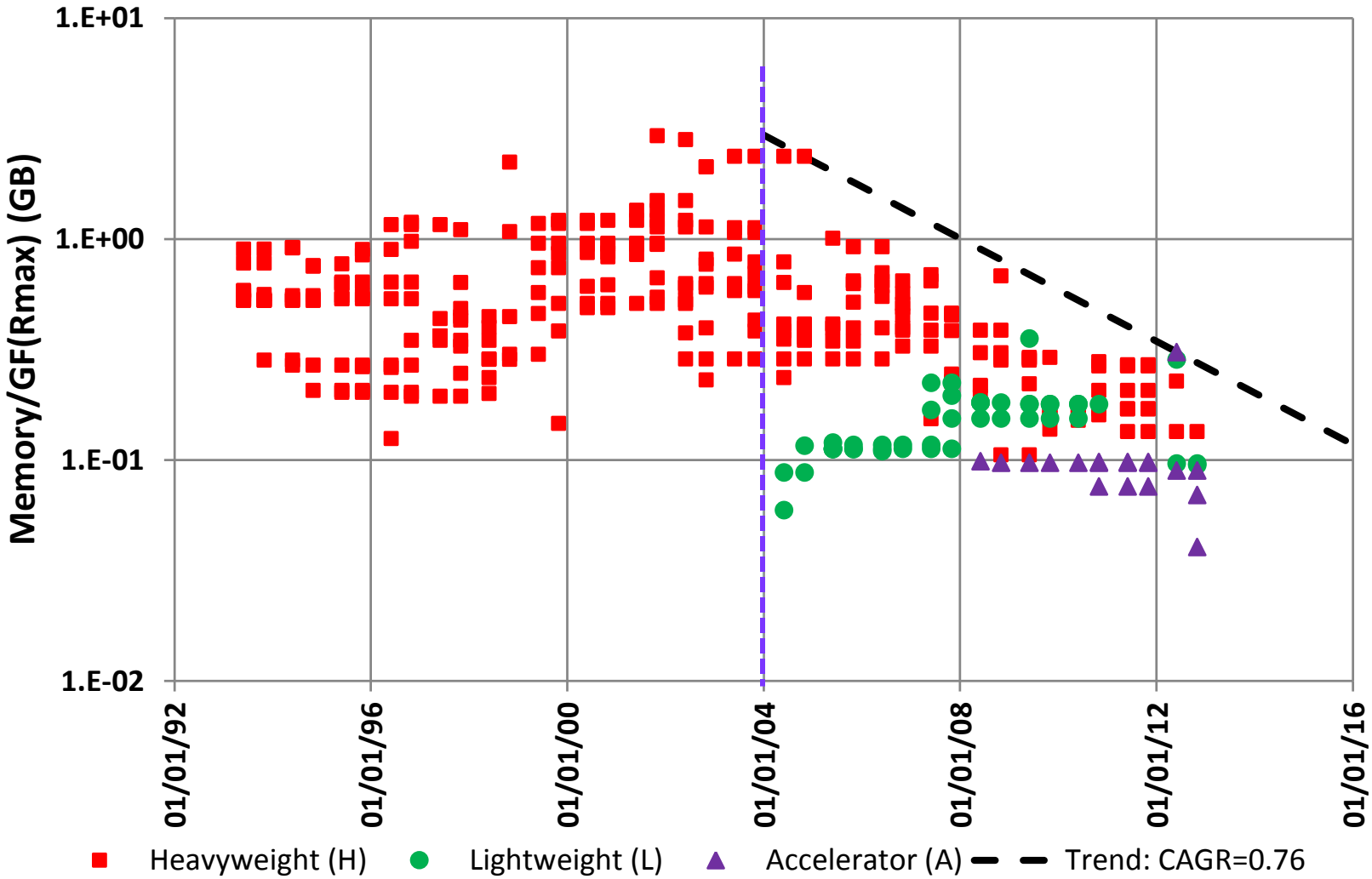
Flops/Cycle: This Drives Programming Complexity



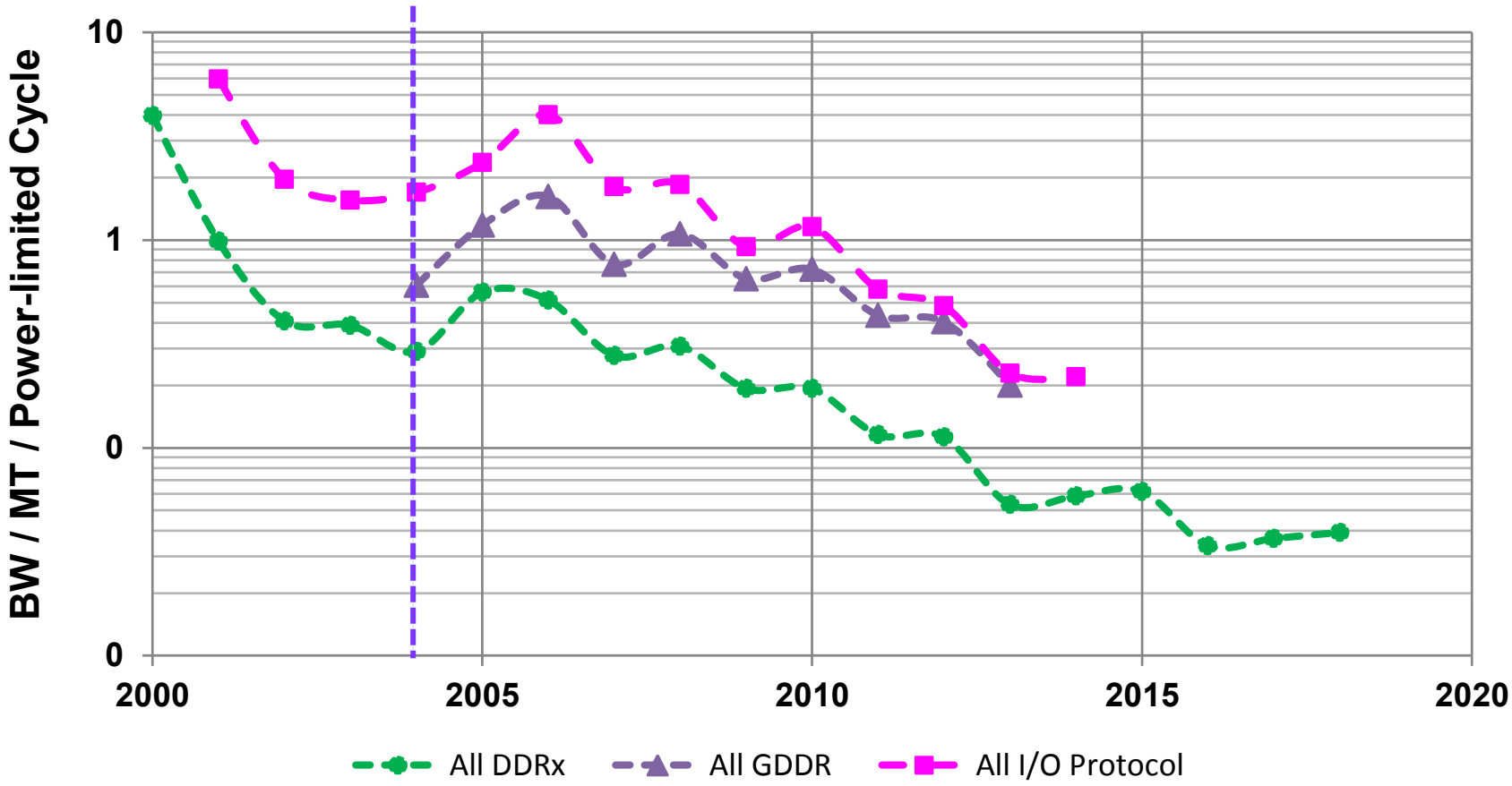
Floating Point Efficiency



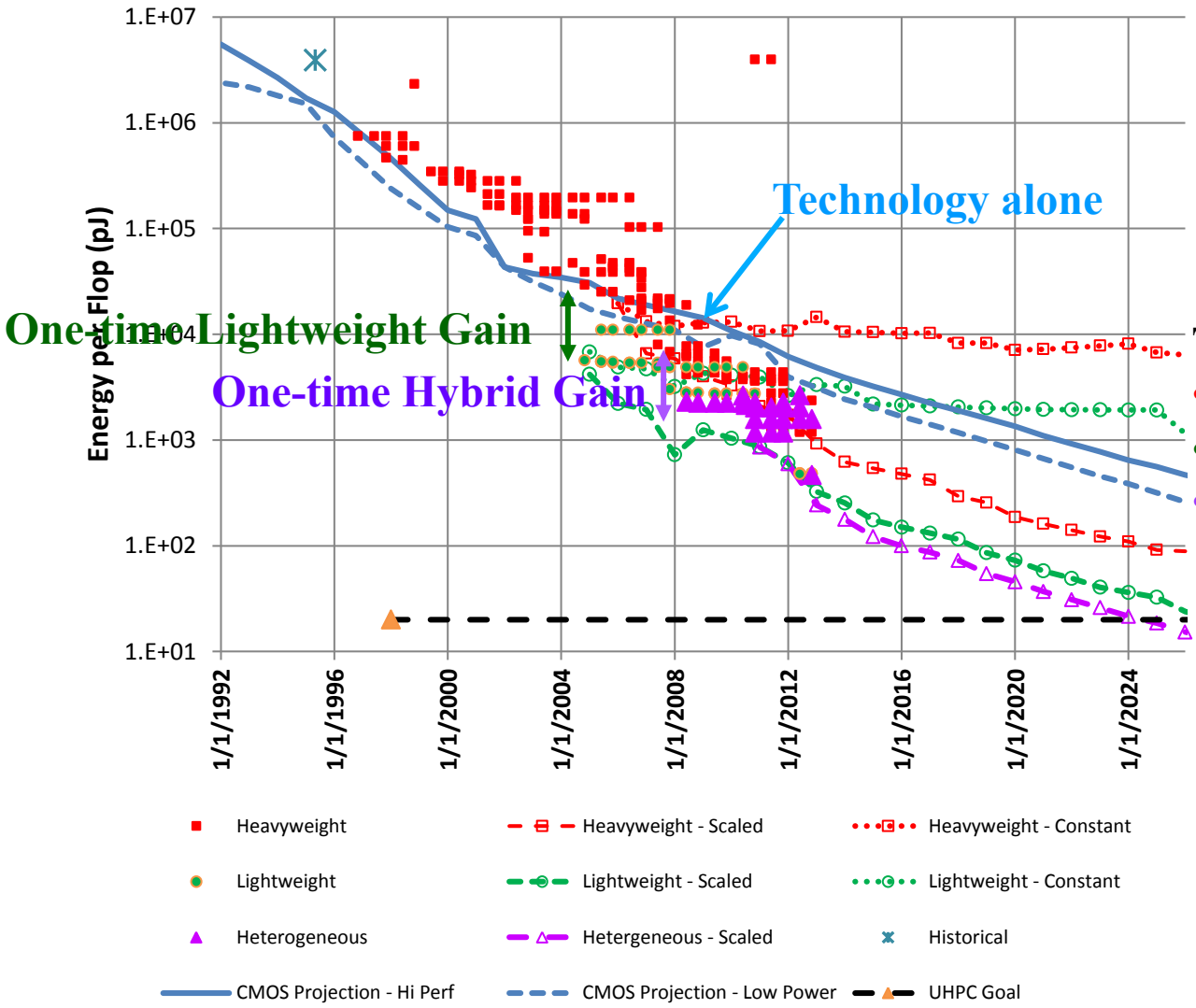
Memory per flop/s is Dropping



Peak Possible Off-chip Bandwidth/ op is Also Dropping



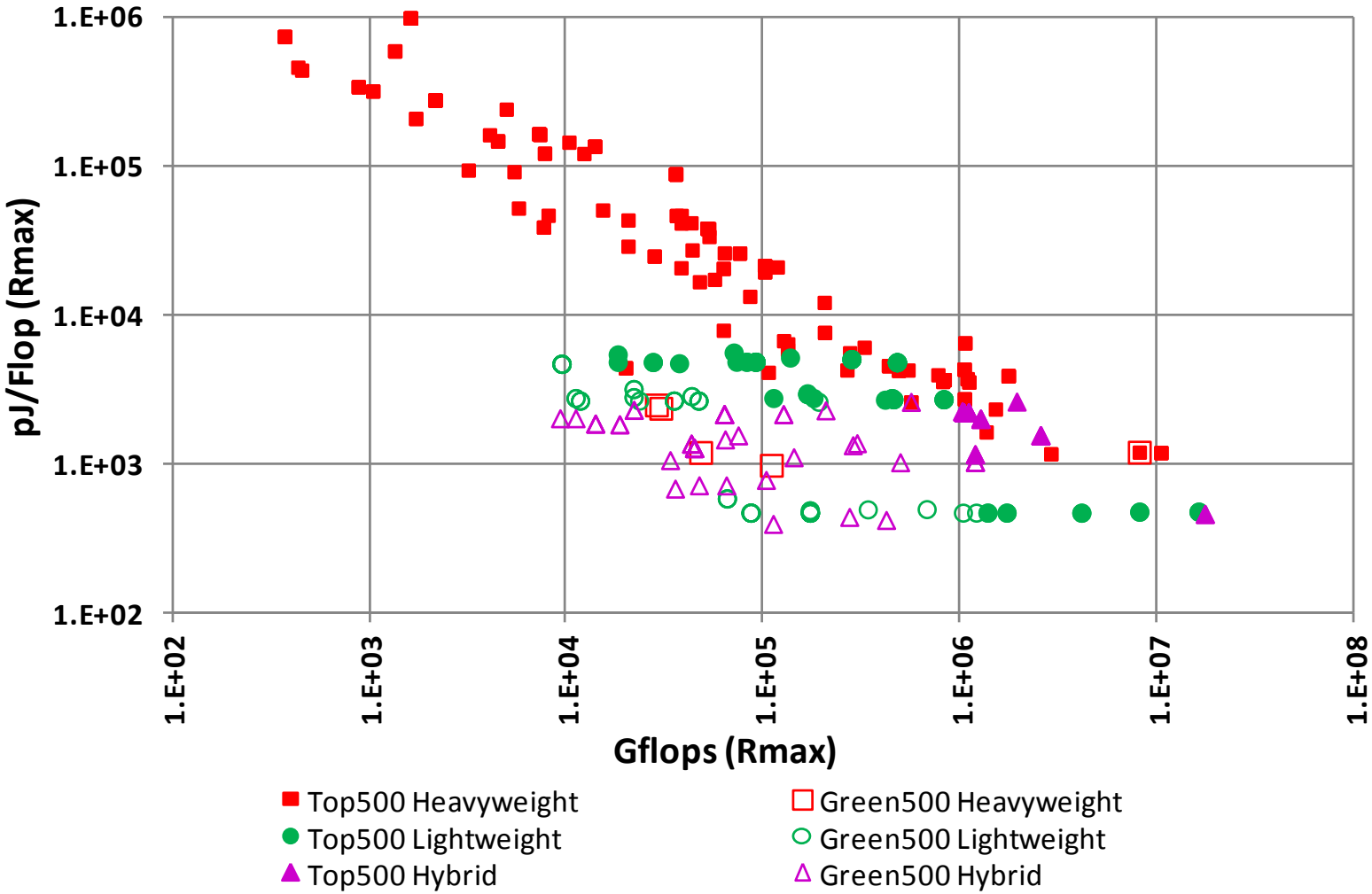
Energy per Flop is Dropping



Trends vs 20pJ/Flop goal

- **Heavyweight: never**
- **Lightweight: close**
- **Hybrid: Possible**

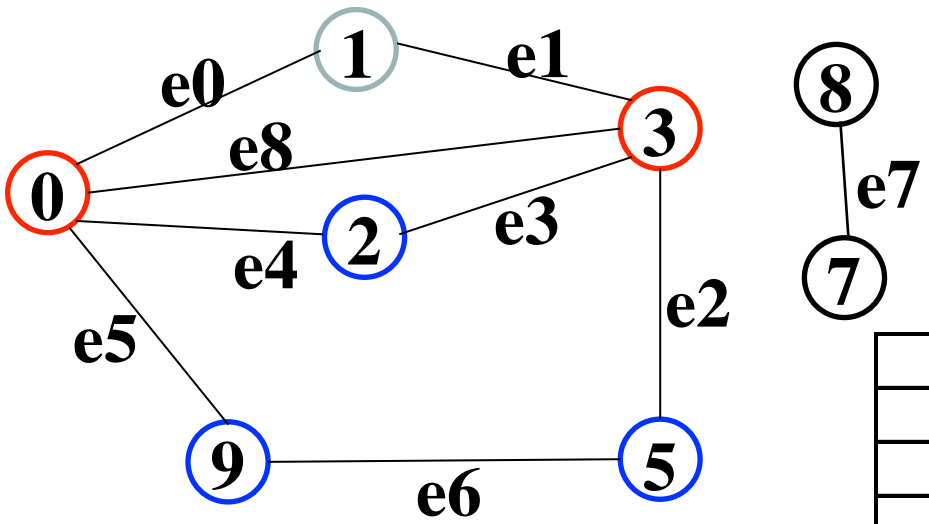
But Energy Efficiency Limits Performance



GRAPH500

Graph 500

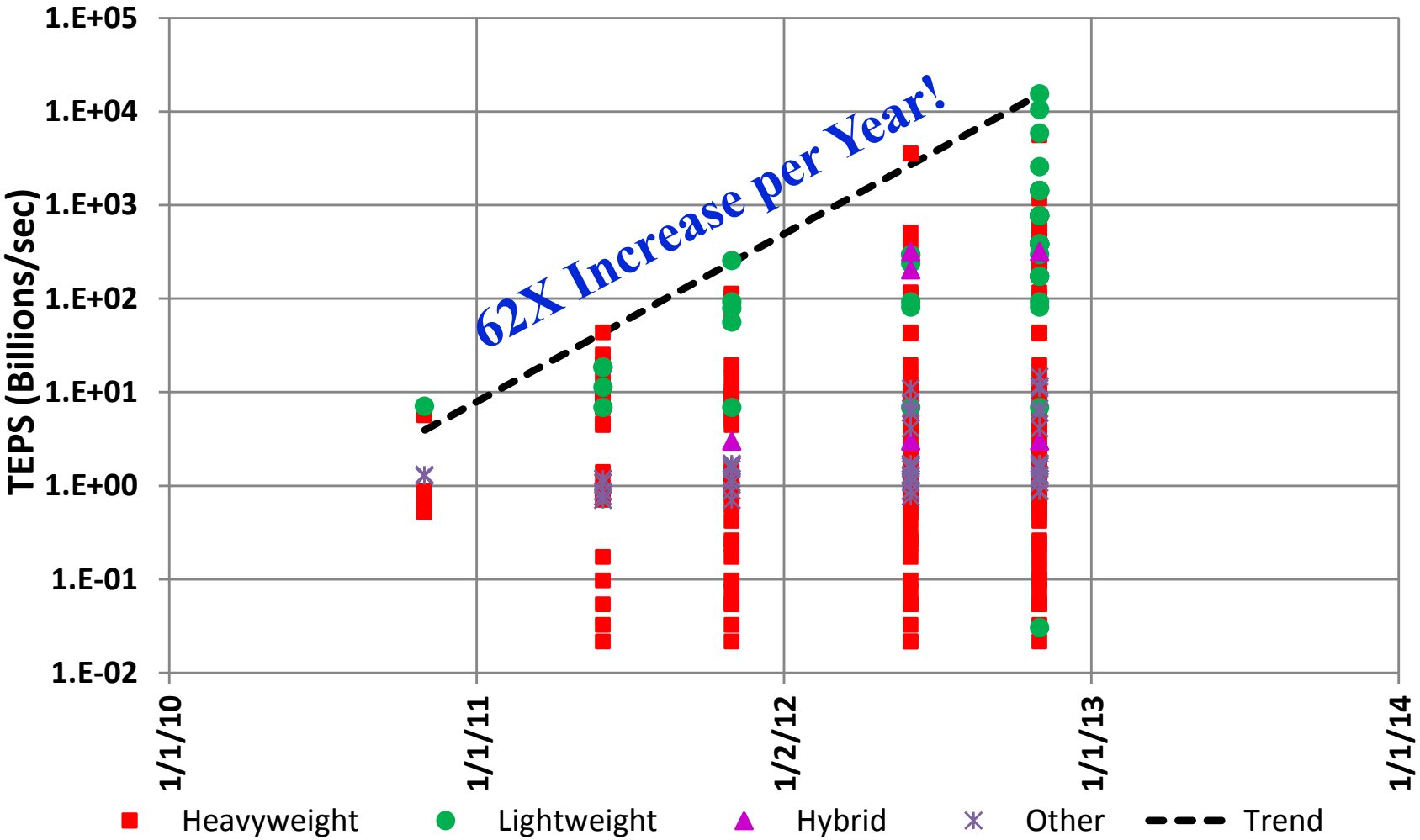
- Start with a root, find reachable nodes
- Simplifications: only 1 kind of edge, no weights
- Performance metric: **TEPS**: Traversed Edges/sec



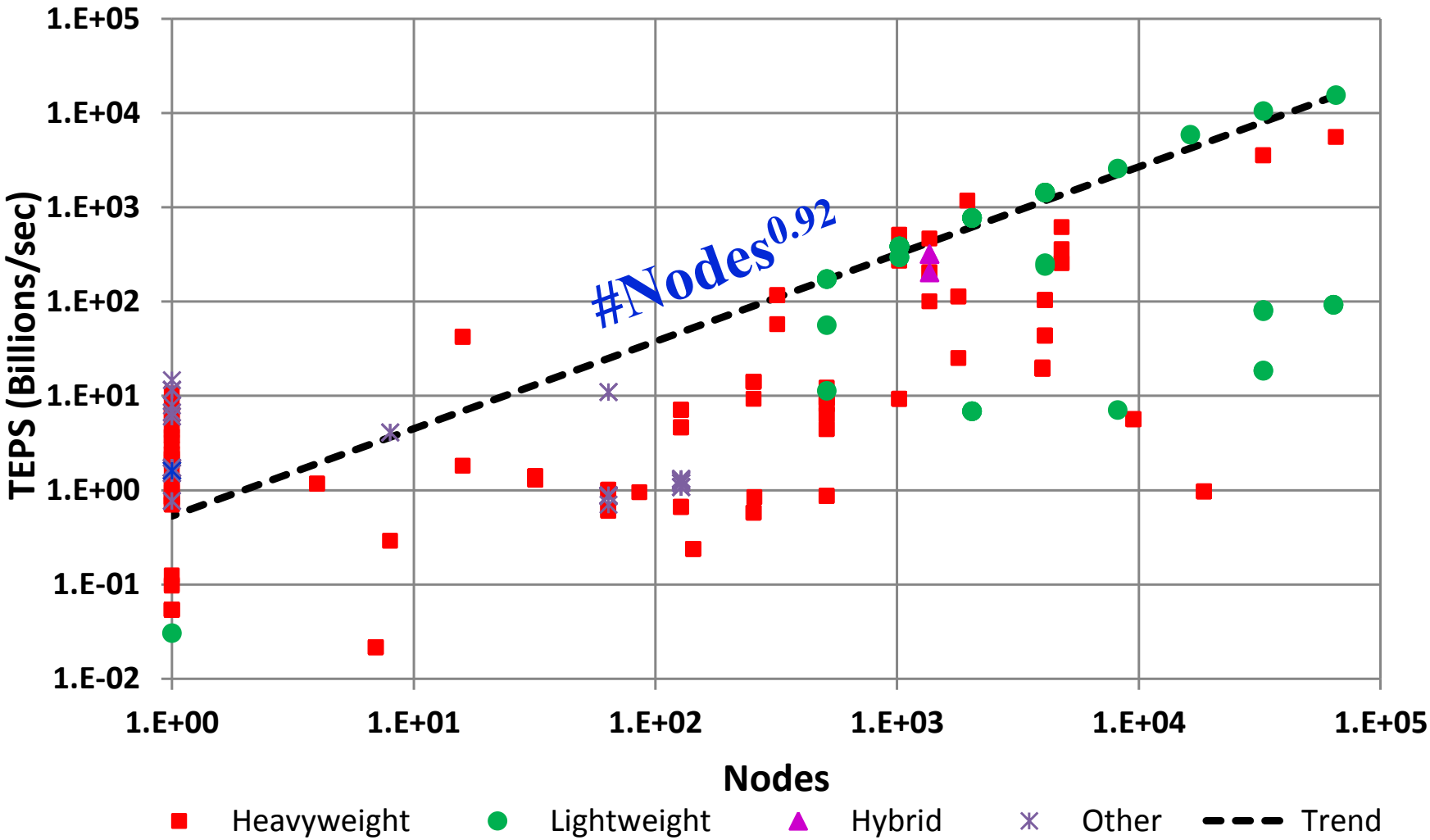
Starting at 1: 1, 0, 3, 2, 9, 5

Level	Scale	Size	GB	Edge(B)
10	26	Toy	17	272
11	29	Mini	140	280
12	32	Small	1024	256
13	36	Medium	17408	272
14	39	Large	143360	280
15	42	Huge	1153434	281.6
			Average	273.6

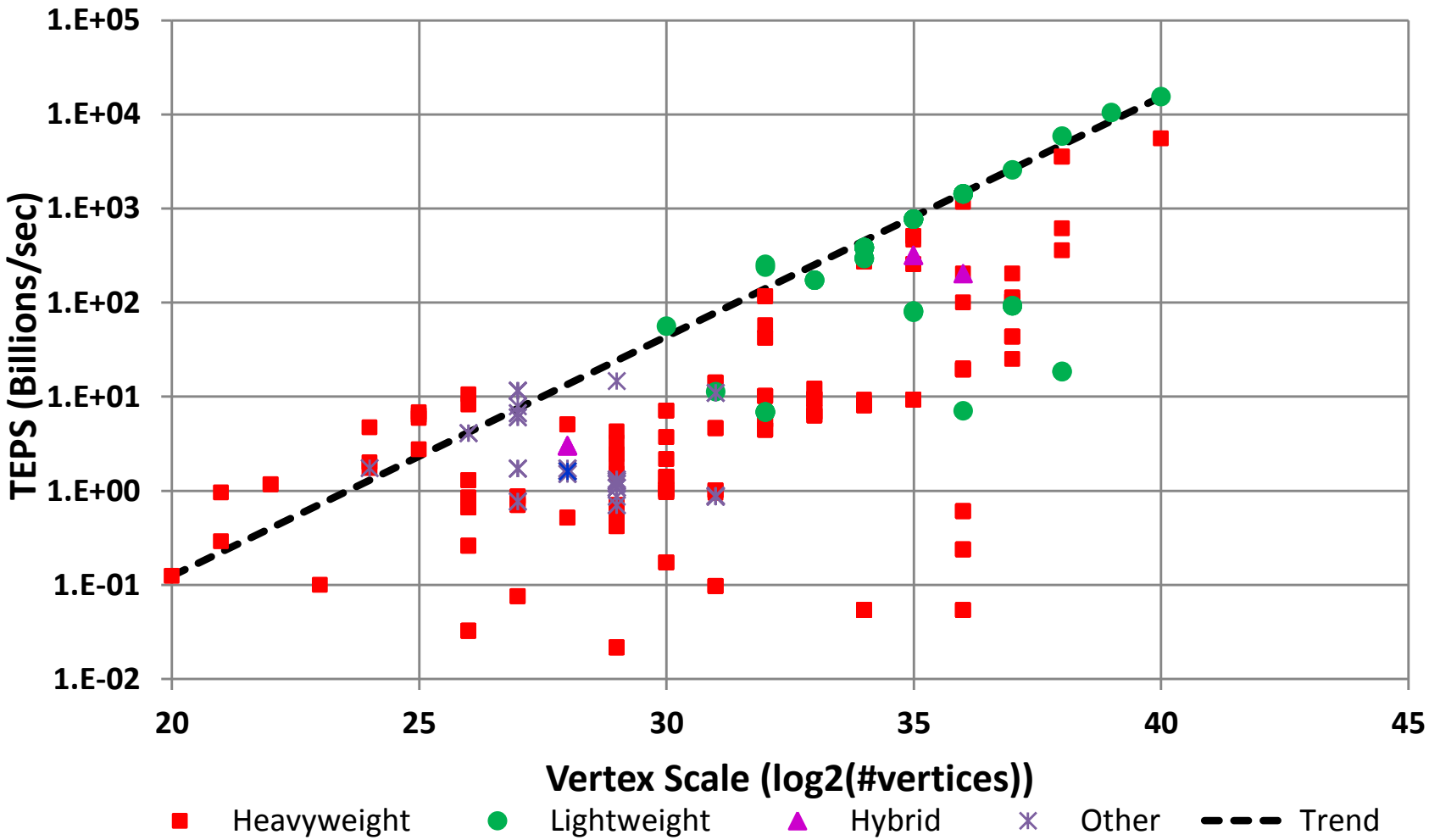
TEPS vs Time


1.E+05
1.E+04
1.E+03
1.E+02
1.E+01
1.E+00
1.E-01
1.E-02
TEPS (Billions/sec)
62X Increase per Year!
1/1/10
1/1/11
1/2/12
1/1/13
1/1/14
Heavyweight
Lightweight
Hybrid
Other
Trend

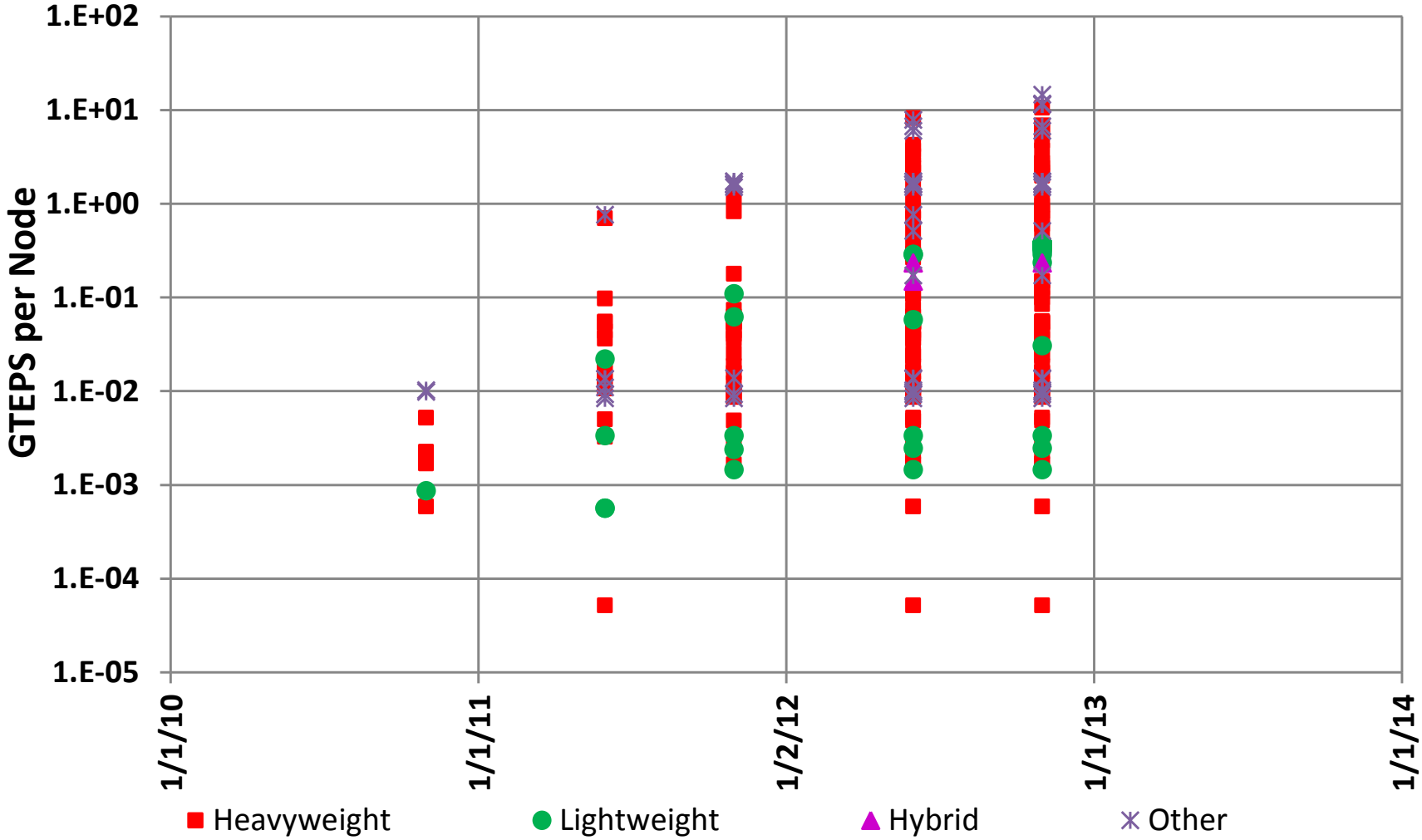
TEPS vs # Nodes



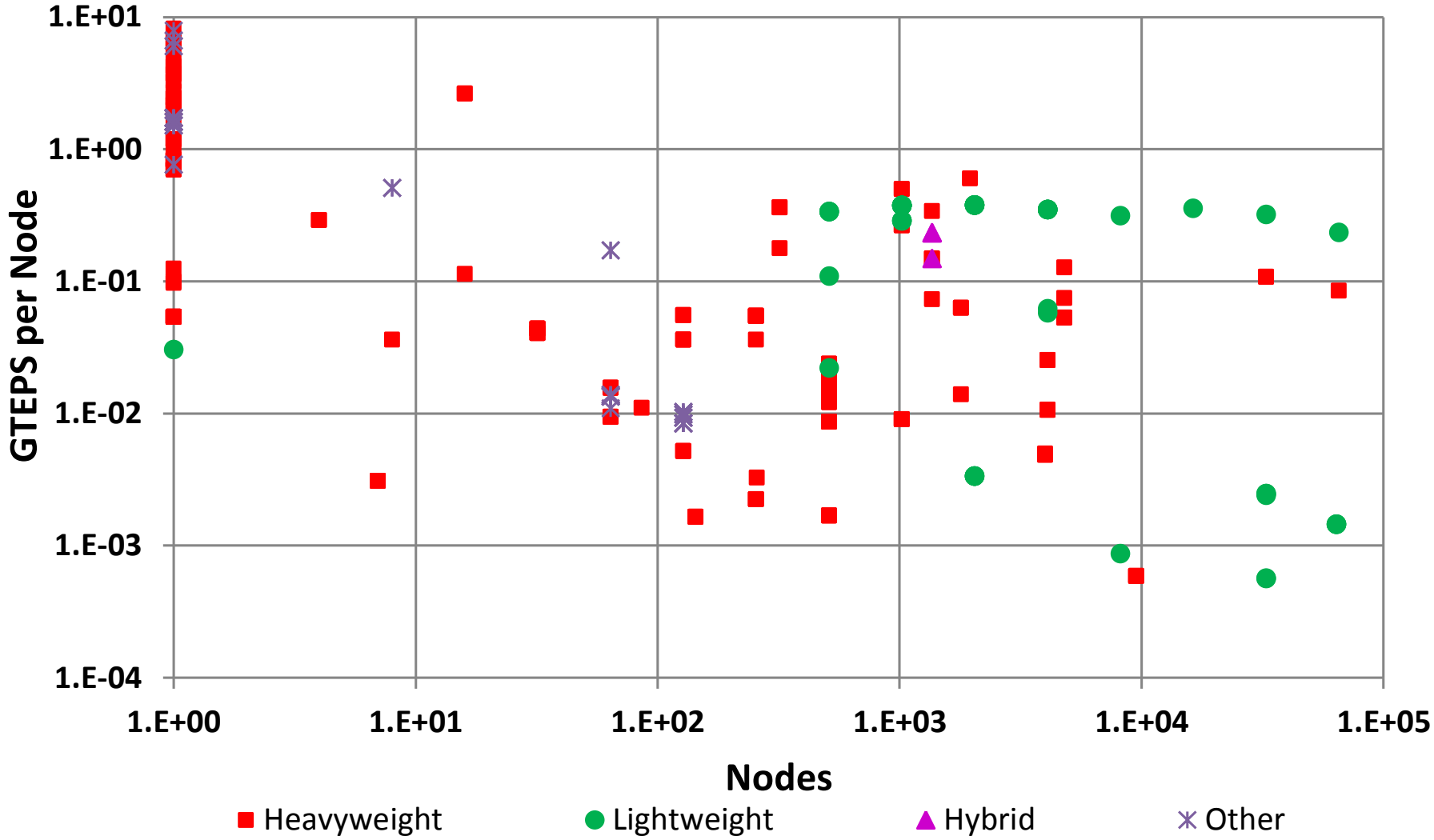
TEPS vs Scale



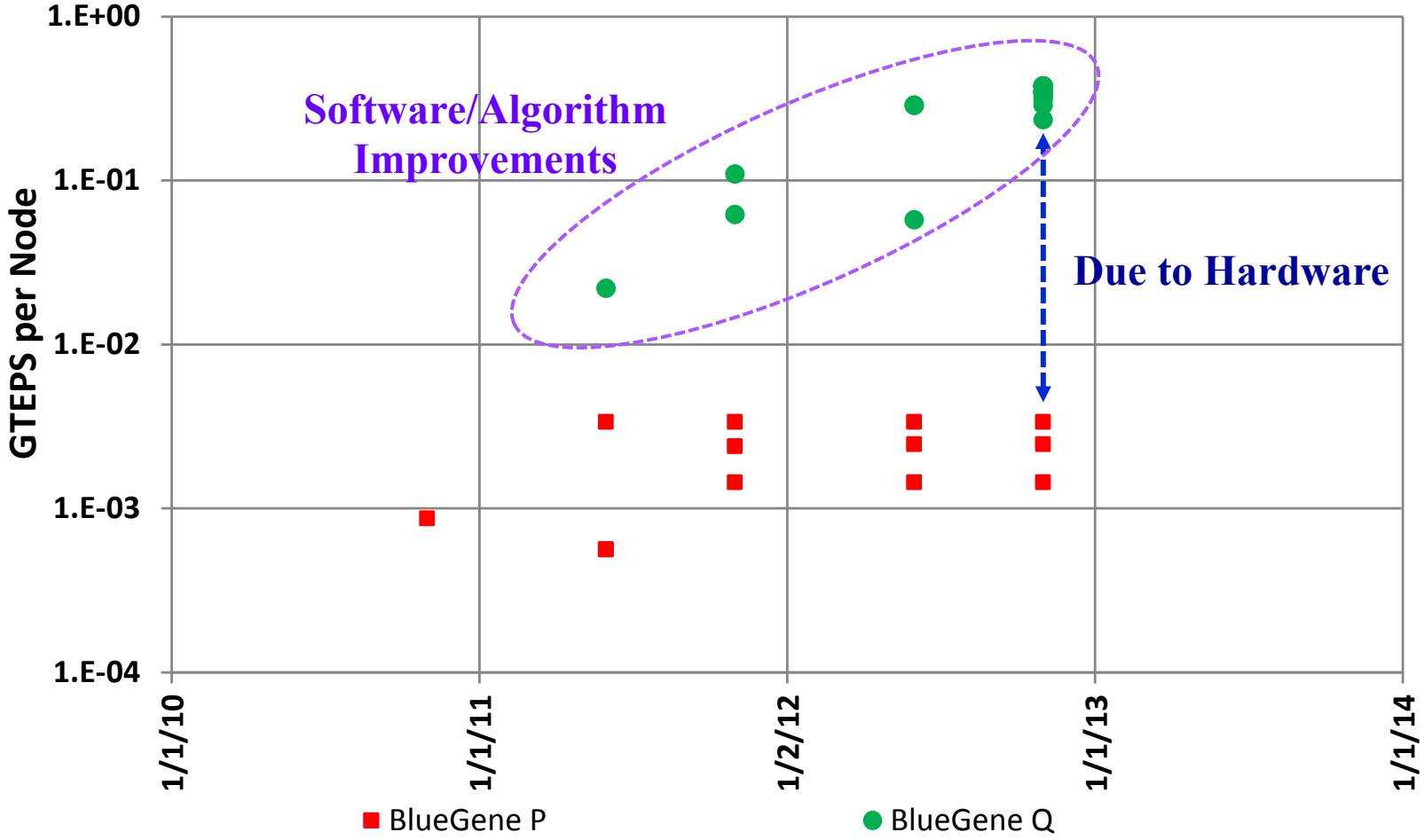
TEPS per Node vs Time



TEPS per Node vs # Nodes

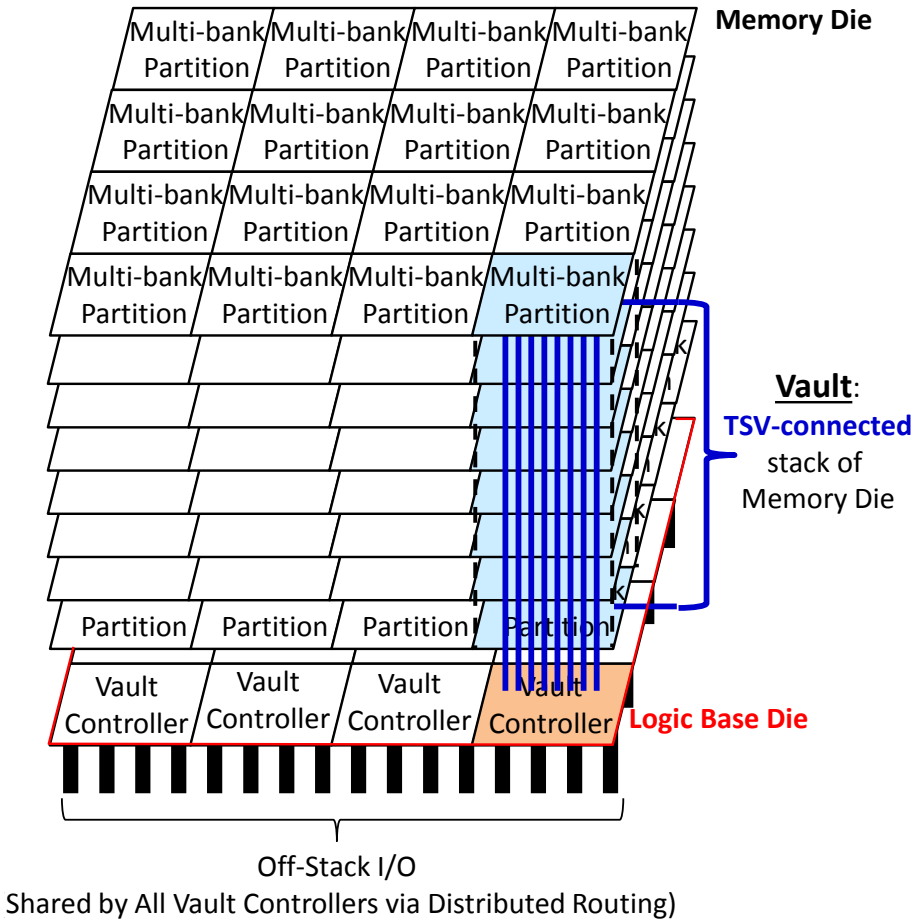


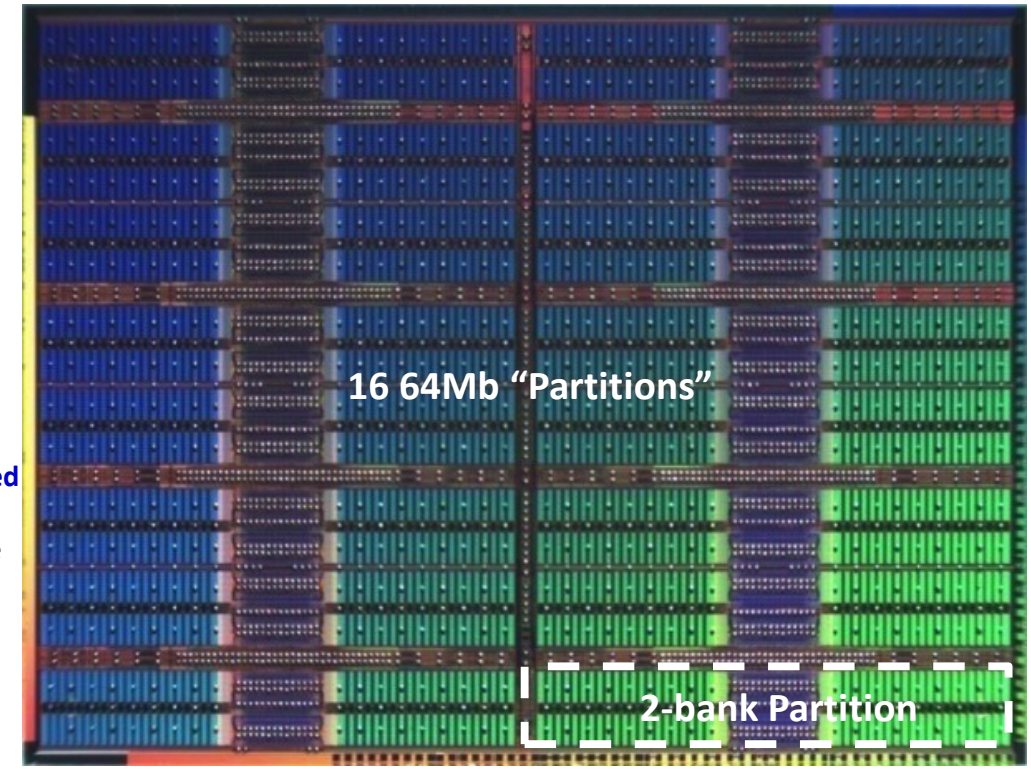
Where Does Improvement Come From? BlueGene Analysis



Looking Forward

Is There an Alternative? Yes – 3D



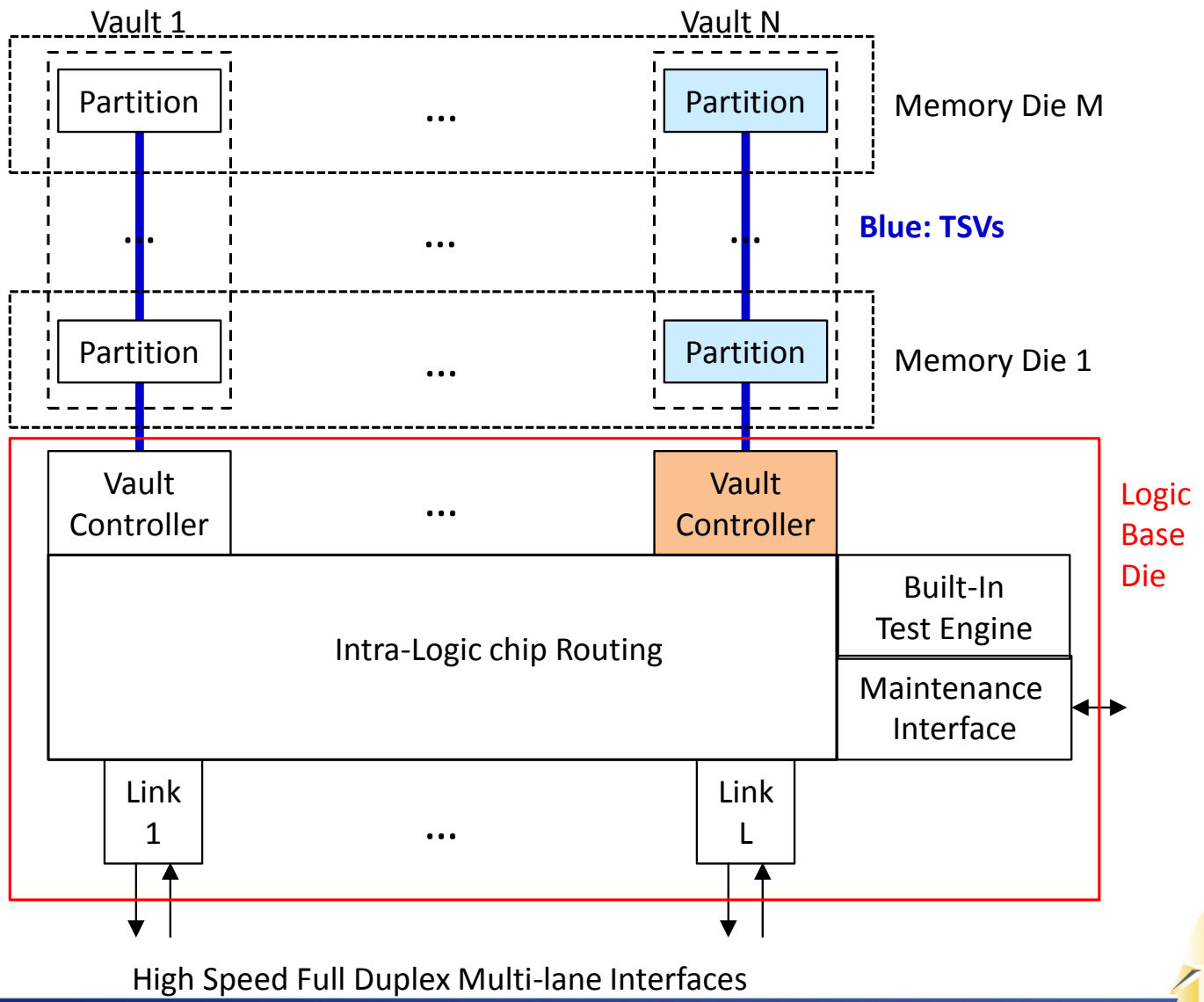


Micron HMC Prototype 2011

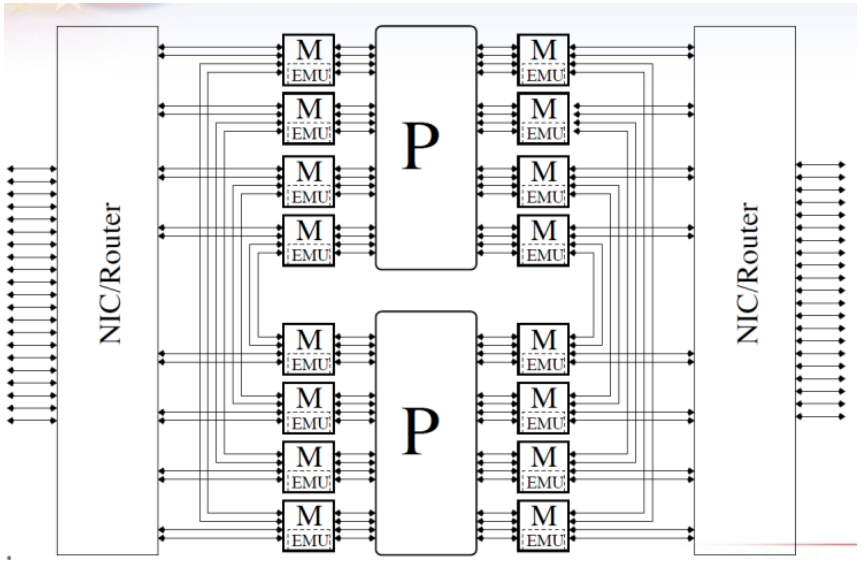
160 GB/s =>320 => …

see http://www.hybridmemorycube.org/

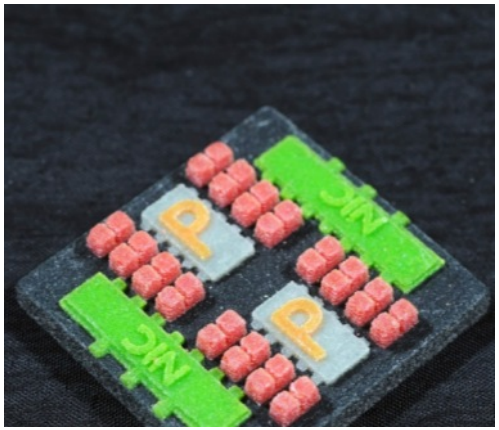
HMC Stack Architecture



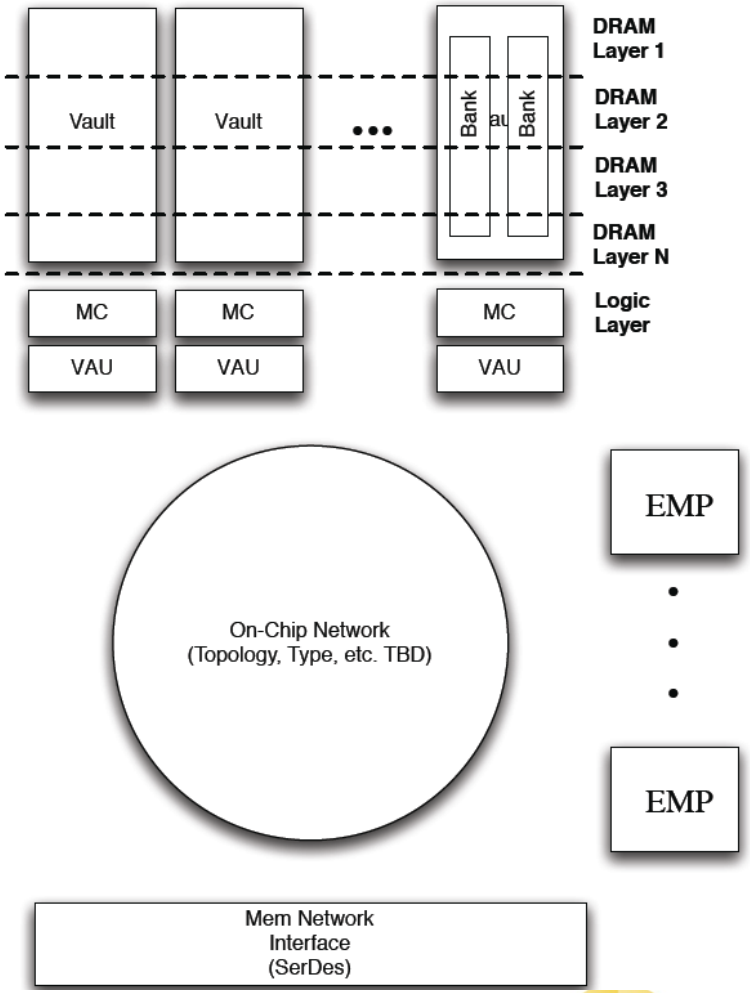
SNL Xcaliber Architecture



(a) X-caliber Node Architecture



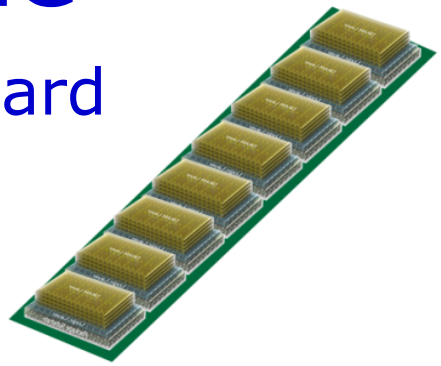
(b) X-caliber Node Mockup

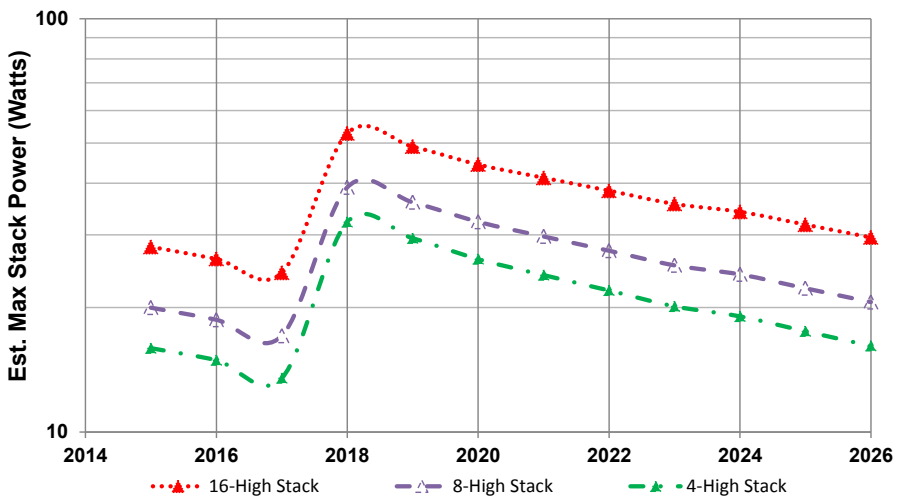


(c) X-caliber stack notional architecture

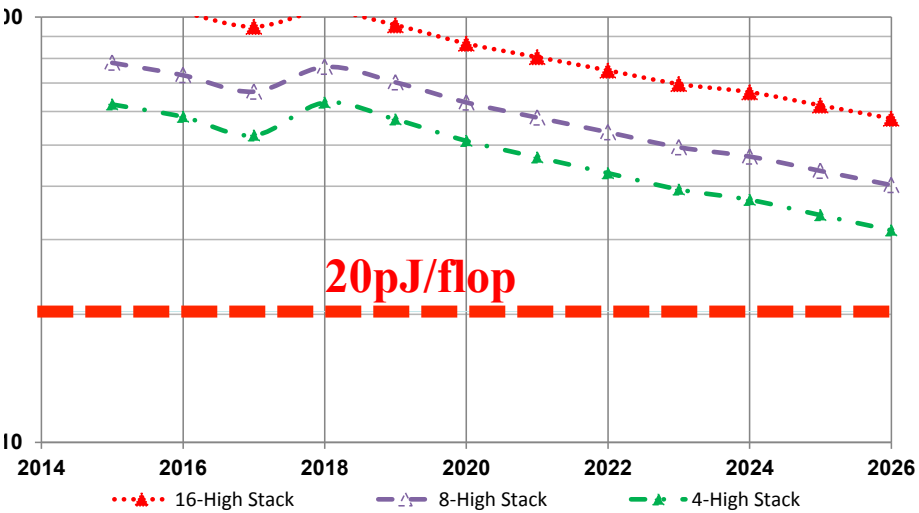
Extrapolation from HMC

- Use ITRS, HMC projections looking forward
- Add simple core/vault
- Ditch the Ps, NICs' and use only stacks
- Package multiple stacks per "DIMM"





Assumes all resources running 100%



If off-chip and memory bandwidth <100% we are almost there!

Summary

- 2004: Turning point in HPC architecture
- Heavyweight alone are going flat
- Hybrid more energy efficient for flops – but there are memory issues
- Lightweight more scalable for both flops and communication
- No escape for massive parallelism
- 3D stacks an interesting alternative