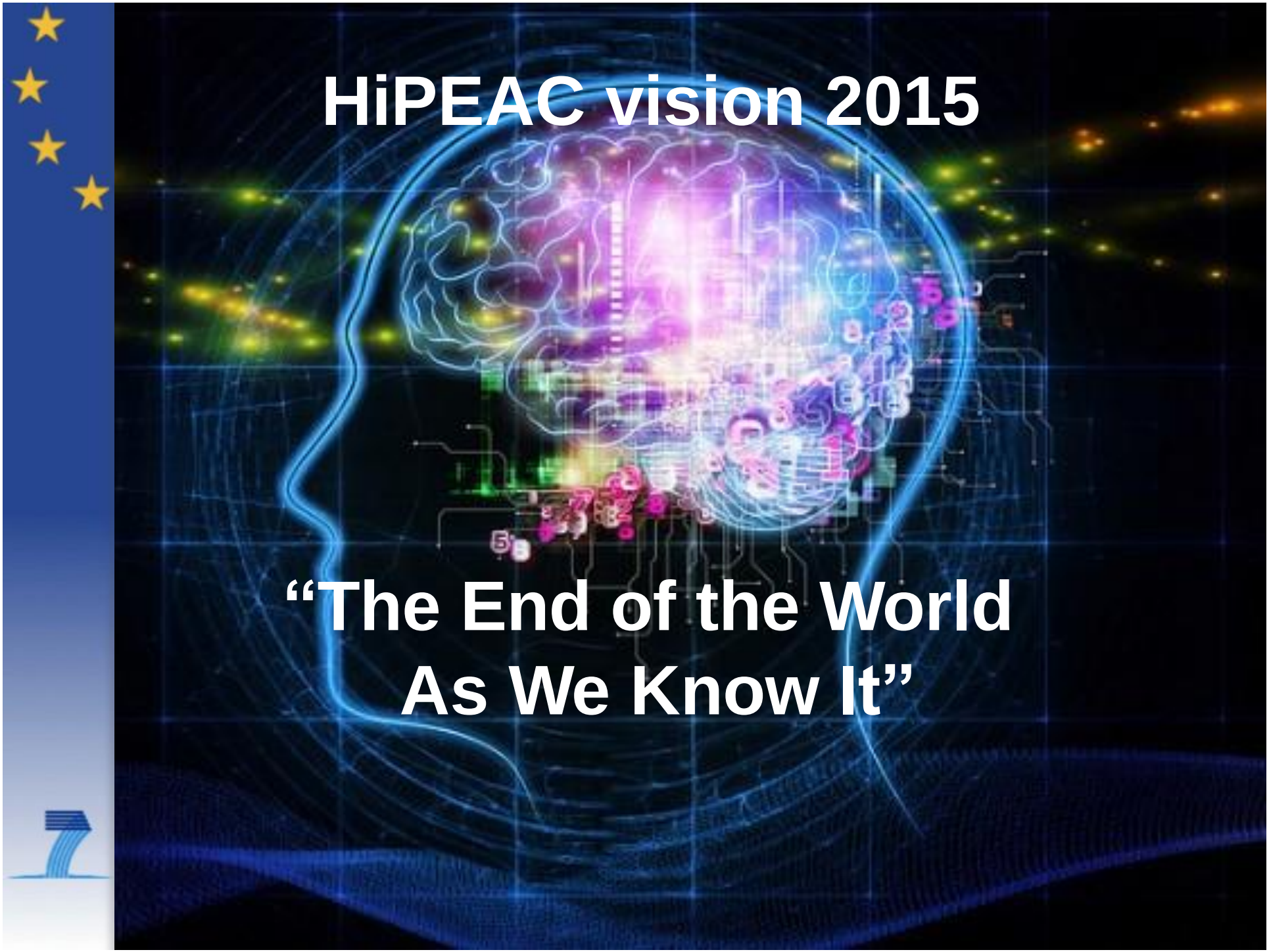

HiPEAC vision 2015
“The End of the World
As We Know It”

What is **HiPEAC**?

- HiPEAC is a European Network of Excellence on **Hi**gh **P**erformance and **E**mbedded **A**rchitecture and **C**ompilation
- Created in 2004, **HiPEAC** gathers over 370 leading European academic and industrial computing system researchers from nearly 140 universities and 70 companies in one virtual centre of excellence of 1500 researchers.

Coordinator: Prof. Koen De Bosschere (UGent)

HiPEAC mission:

HiPEAC encourages computing innovation in Europe by providing:

- Collaboration grants, internships, sabbaticals, the semi-annual computing systems week,
- The ACACES summer school, the yearly HiPEAC conference.

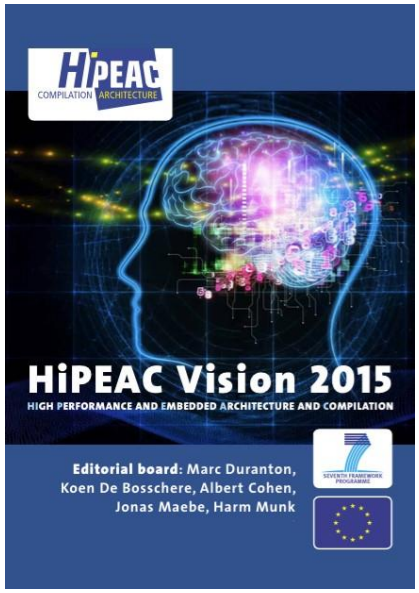
The **HiPEAC** Vision

2008

2009

2011

201ti



2015

http://www.hipeac.org/vision/

- Electronic and paper version available now
- Paper version:
  - Send to the members with the newsletter
  - Was available at HiPEAC 2015 conference

Editors: Marc Duranton (FR-CEA), Koen de Bosschere (BE-U Gent), Albert Cohen (FR-INRIA), Jonas Maebe (BE-U Gent), Harm Munk (NL-ASTRON)

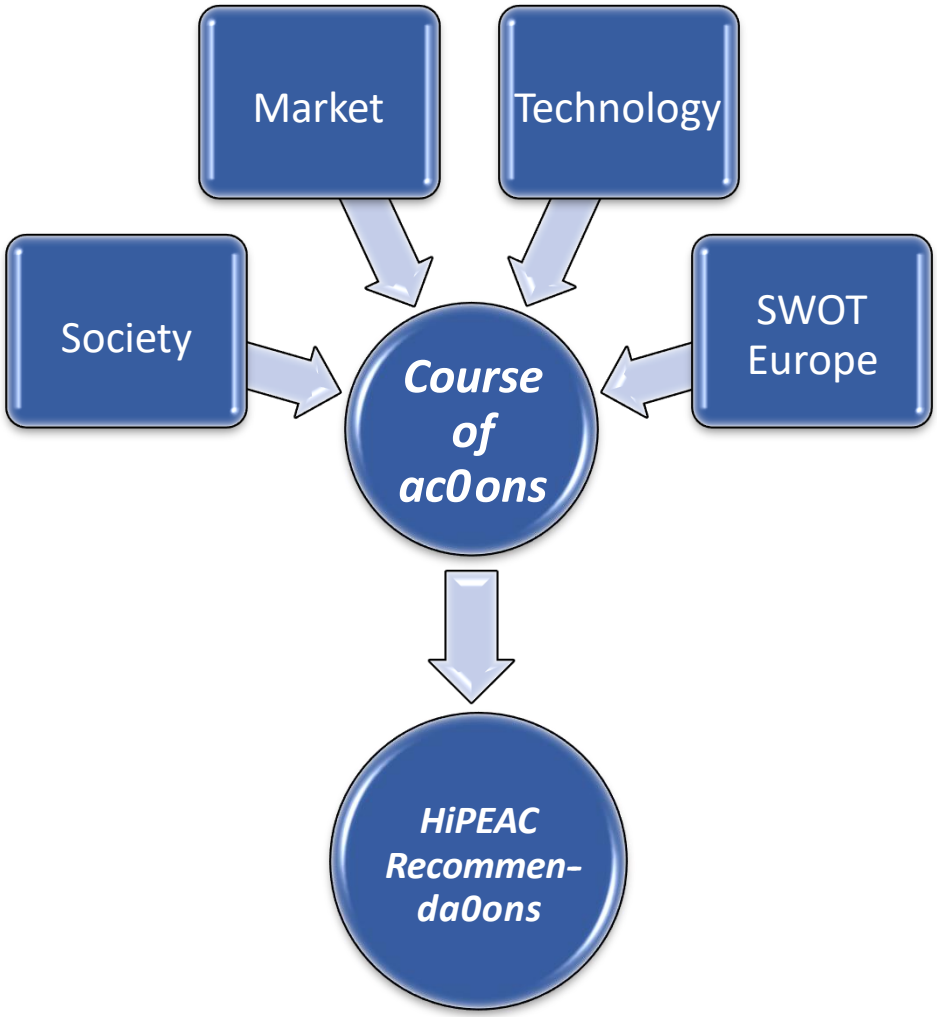
Glimpse into the HiPEAC Vision 2015

For the first time, we have noticed that the community really ***starts looking for disruptive solutions,***

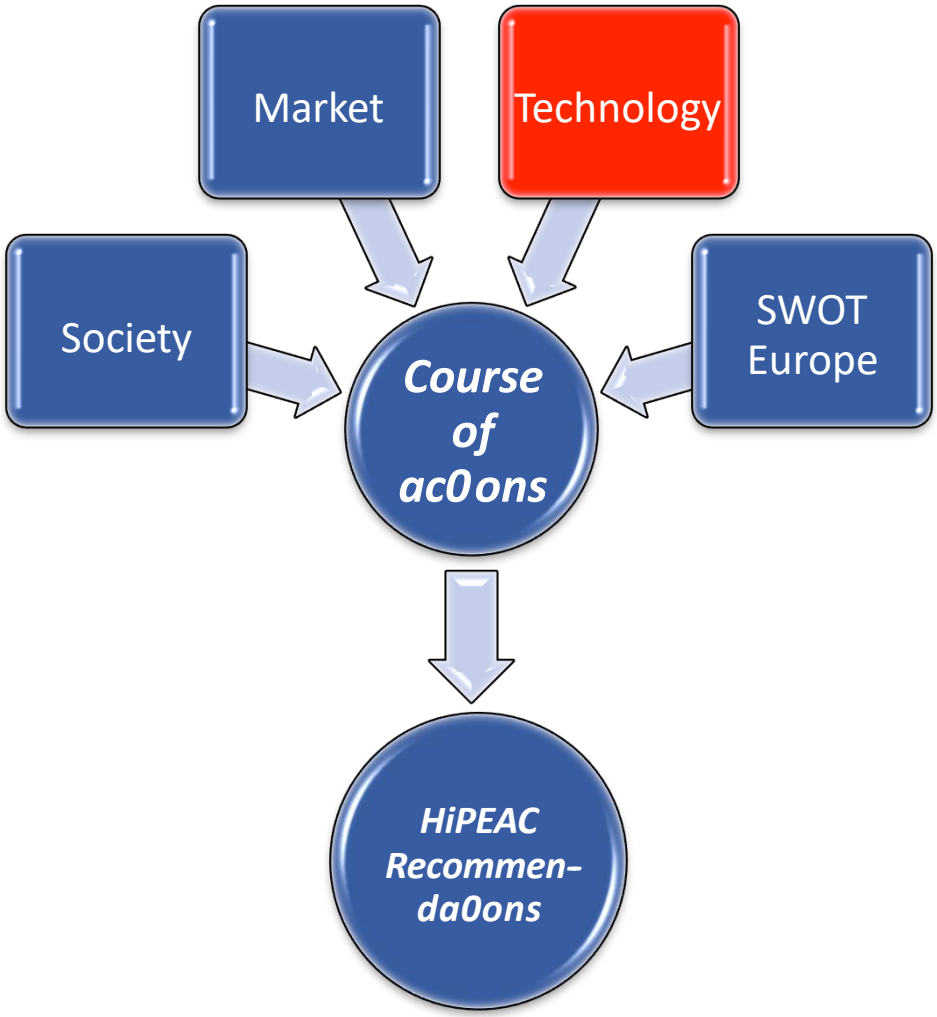
and that incrementally improving current technologies is considered inadequate to address the challenges that the computing community faces:

"The End of the World As We Know It"

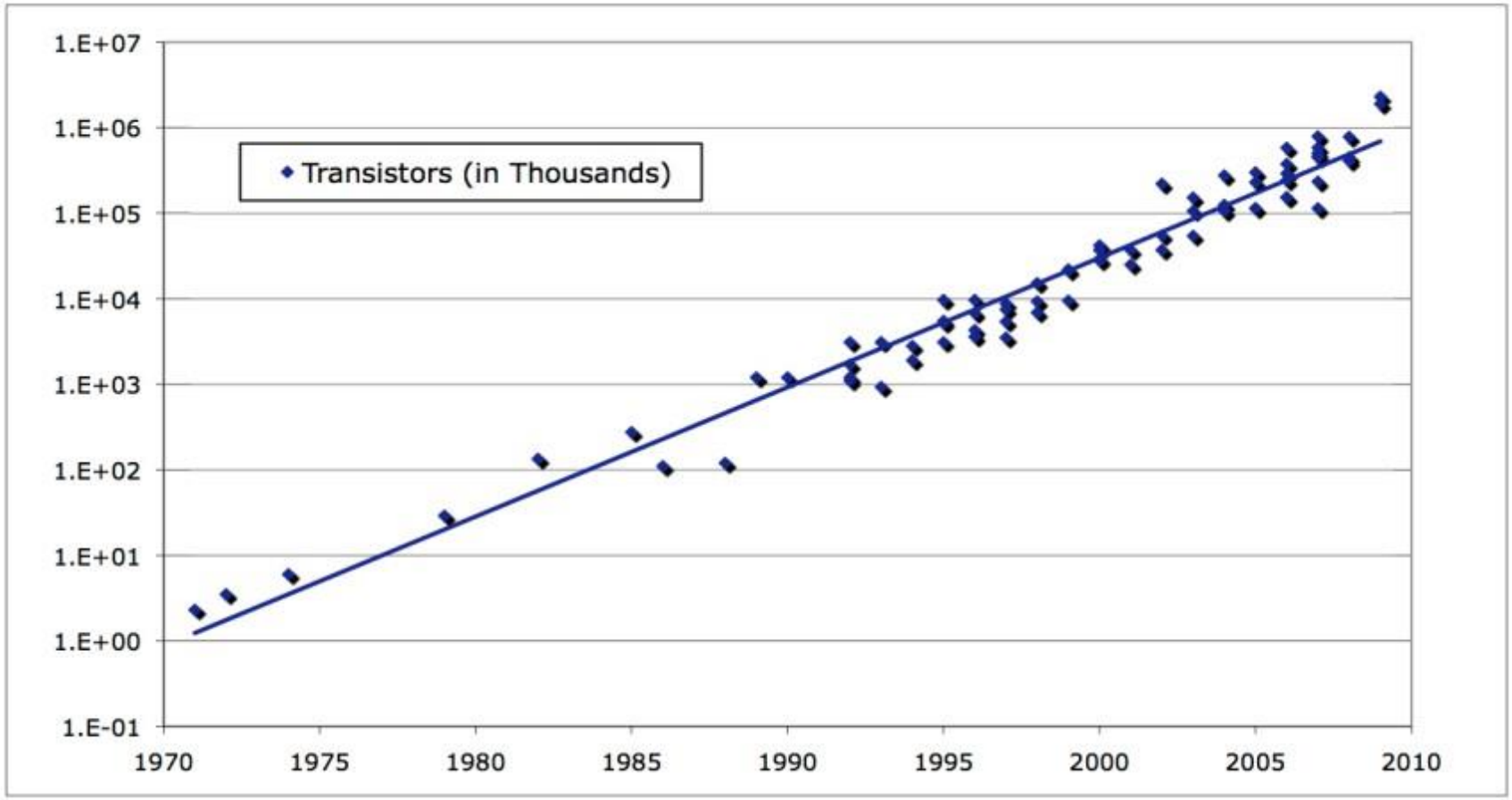
Structure of the HiPEAC vision 2015



Structure of the HiPEAC vision 2015



Moore’s law: increase in transistor density



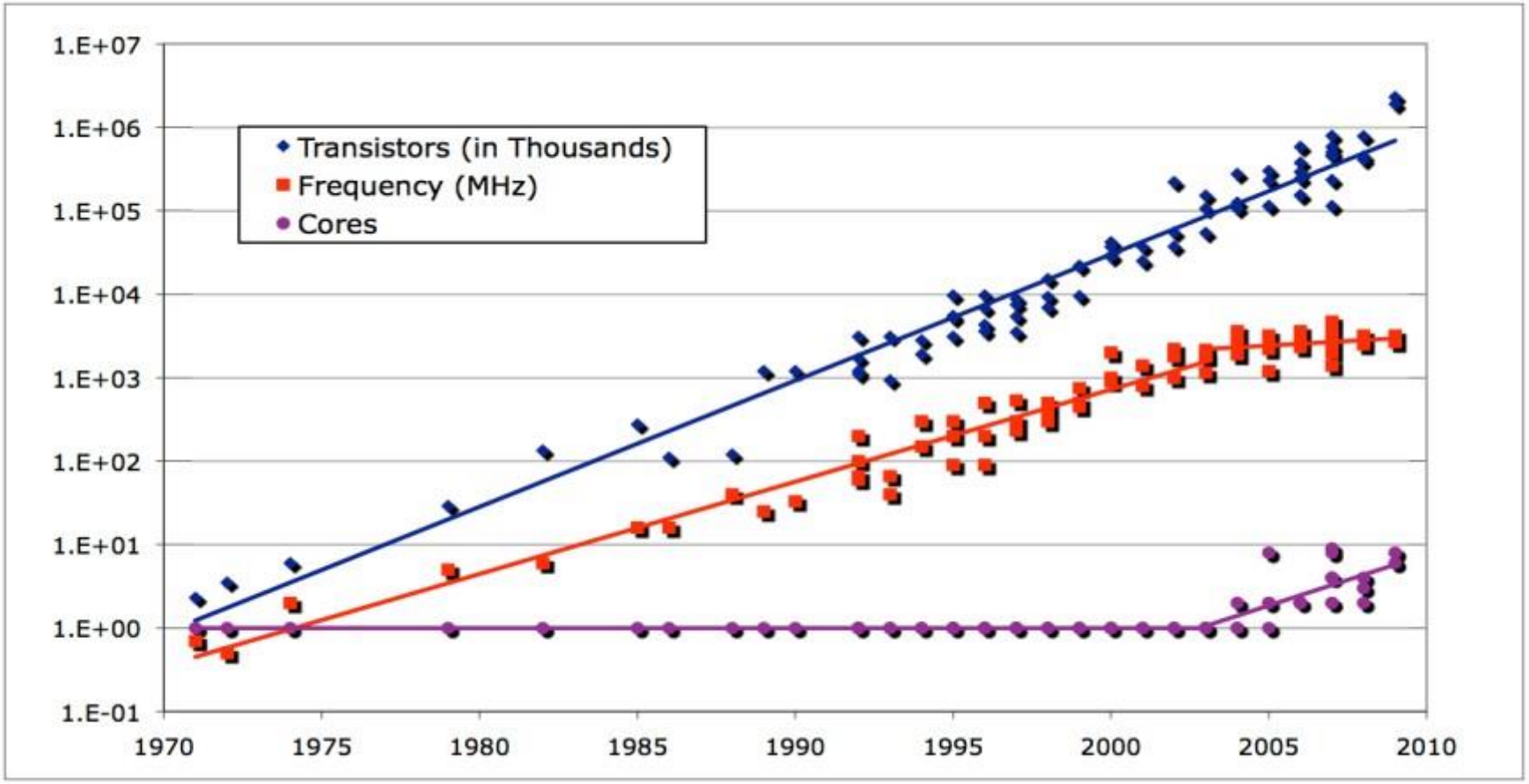
Source from Kunle Olukotun, Lance Hammond, Herb Sutter, Burton Smith, Chris Batten, and Krste Asanoviç

The end of Dennard Scaling

Parameter (scale factor = a)	Classic Scaling	Current Scaling
Dimensions	1/a	1/a
Voltage	1/a	1
Current	1/a	1/a
Capacitance	1/a	>1/a
Power/Circuit	$1/a^2$	1/a
Power Density	1	a
Delay/Circuit	1/a	~1

Source: Krisztián Flautner "From niche to mainstream: can critical systems make the transition?"

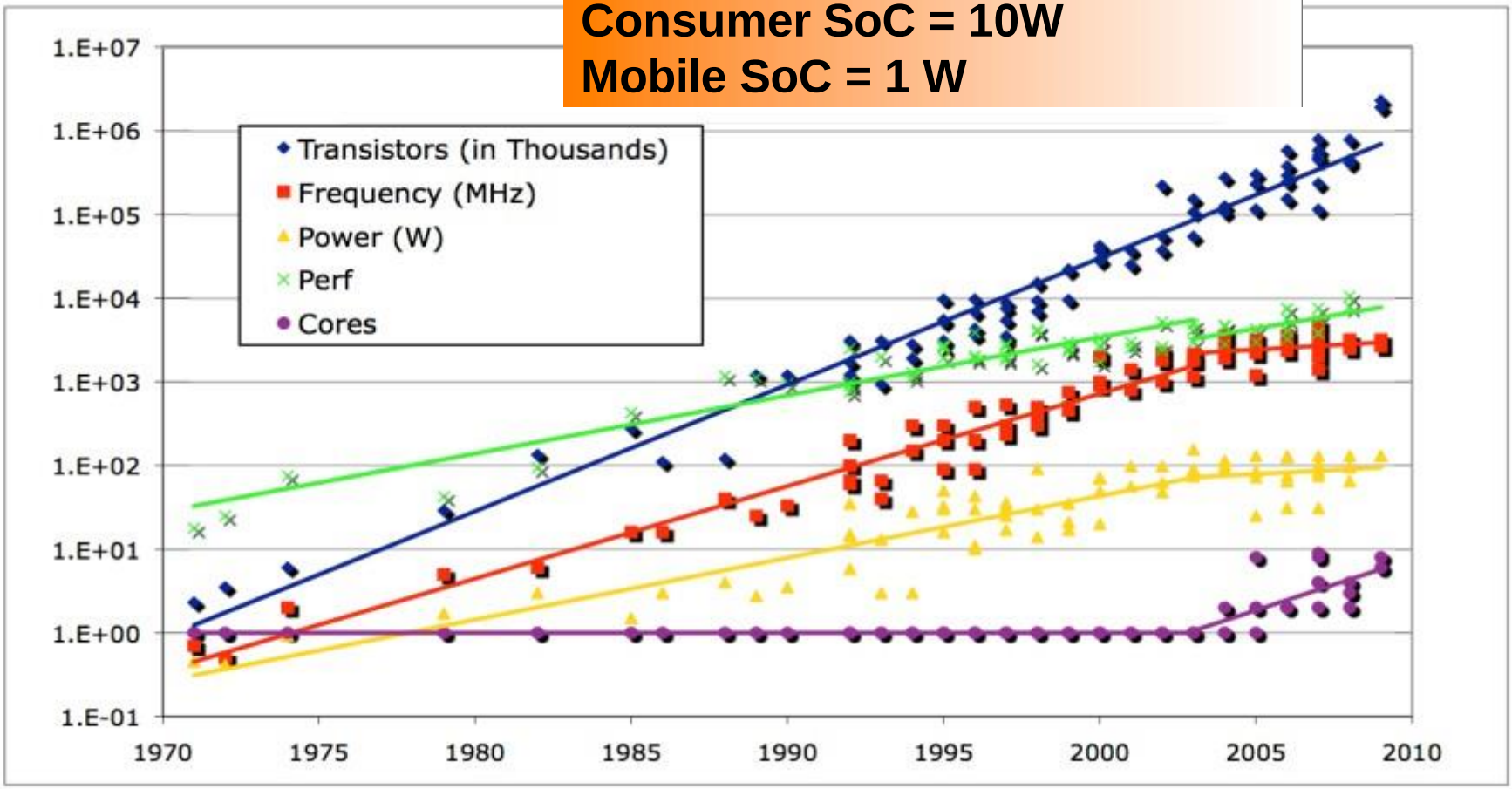
Limited frequency increase ⇒ more cores



Source from Kunle Olukotun, Lance Hammond, Herb Sutter, Burton Smith, Chris Batten, and Krste Asanoviç

Limitation by power density and dissipation

GP CPU = 200 W (45 nm)
Consumer SoC = 10W
Mobile SoC = 1 W

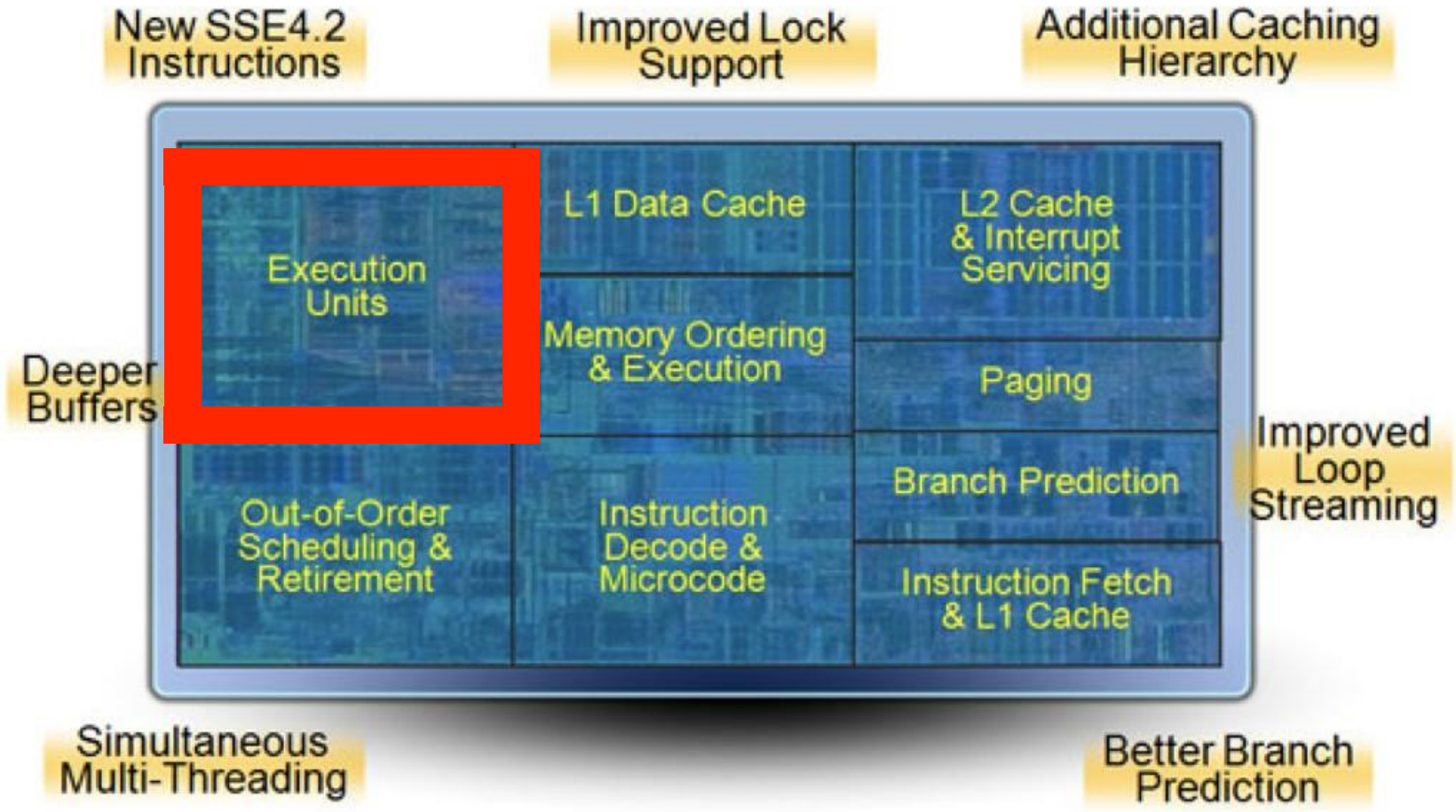


Source from Kunle Olukotun, Lance Hammond, Herb Sutter, Burton Smith, Chris Batten, and Krste Asanoviç

Why using several compute cores?

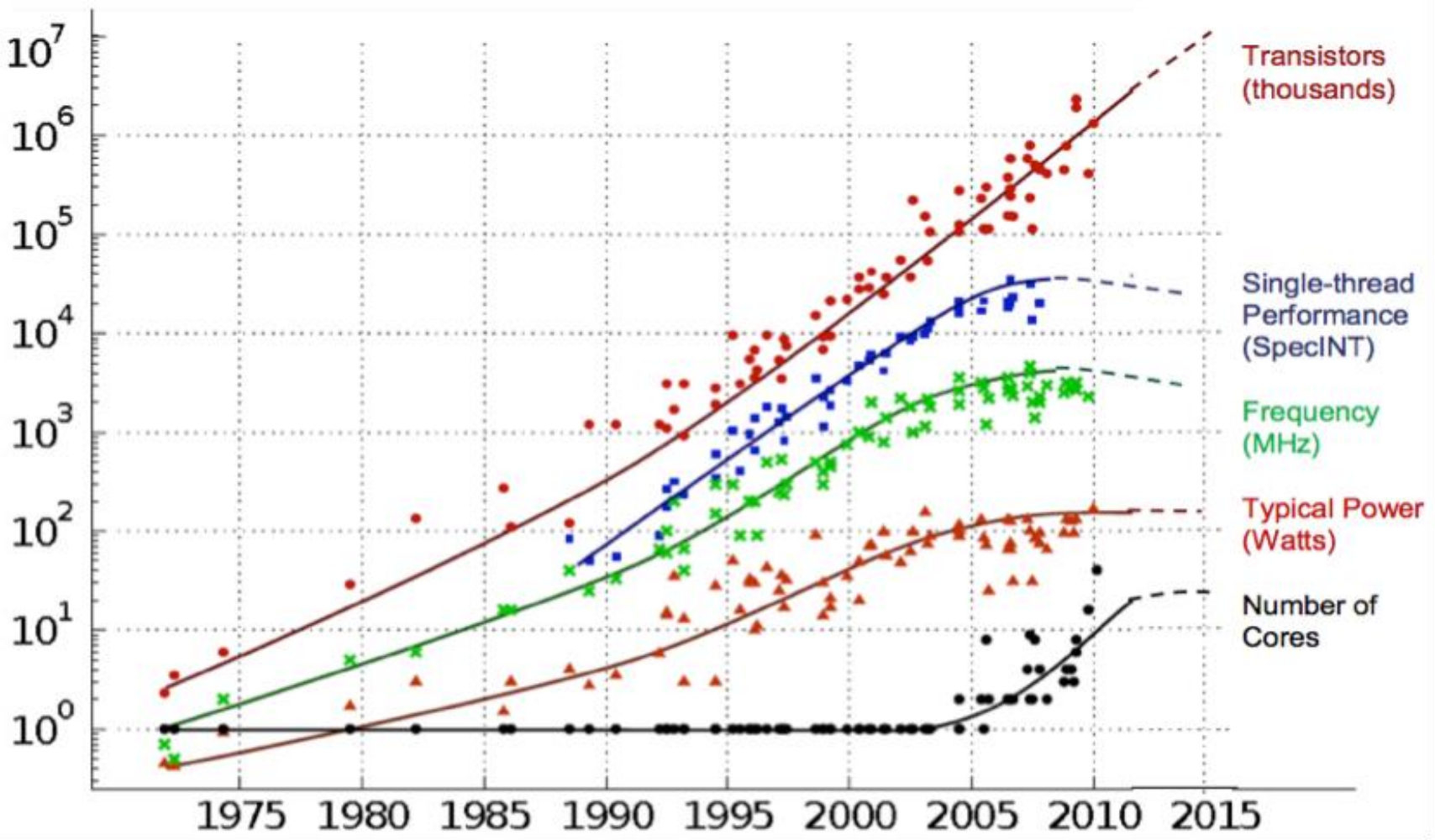
1. Using several cores is also an answer to the Law of Diminishing Returns [Pollack's Rule] :
   - Effectiveness per transistor decreases when the size of a single core is increased, due to the locality of computation
   - Controlling a larger core and data transport over a single larger core is super-linear
   - **Smaller cores are more efficient** in ops/mm^2/W
2. Large area of today's microprocessors are for best effort processing and used to cope with unpredictability (branch prediction, reordering buffers, instructions, caches).

Less than 20% of the area for execution units



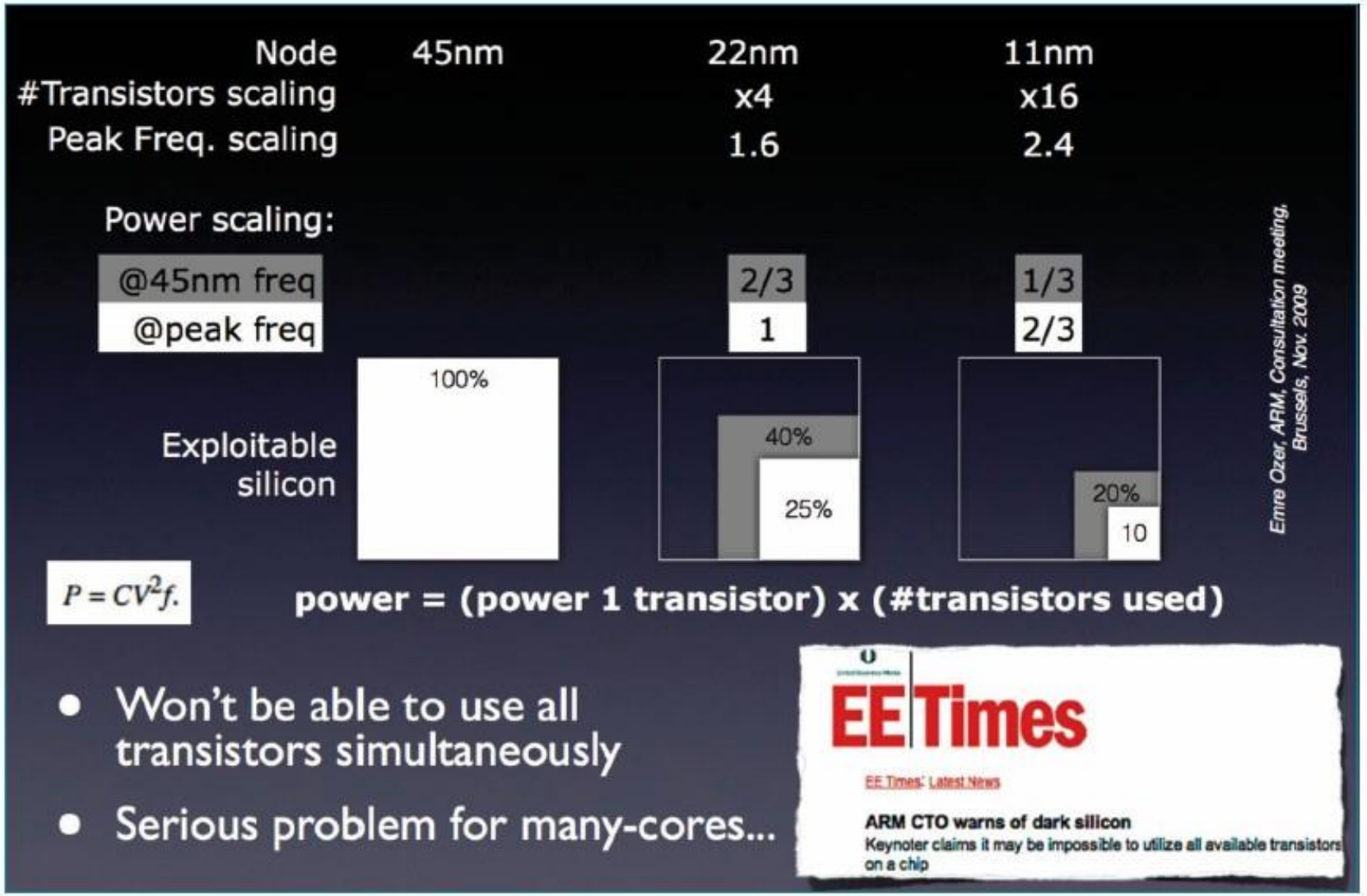
Source: Dan Connors, “OpenCL and CUDA Programming for Multicore and GPU Architectures» ACACES 2011

Stagnation of performance since few years

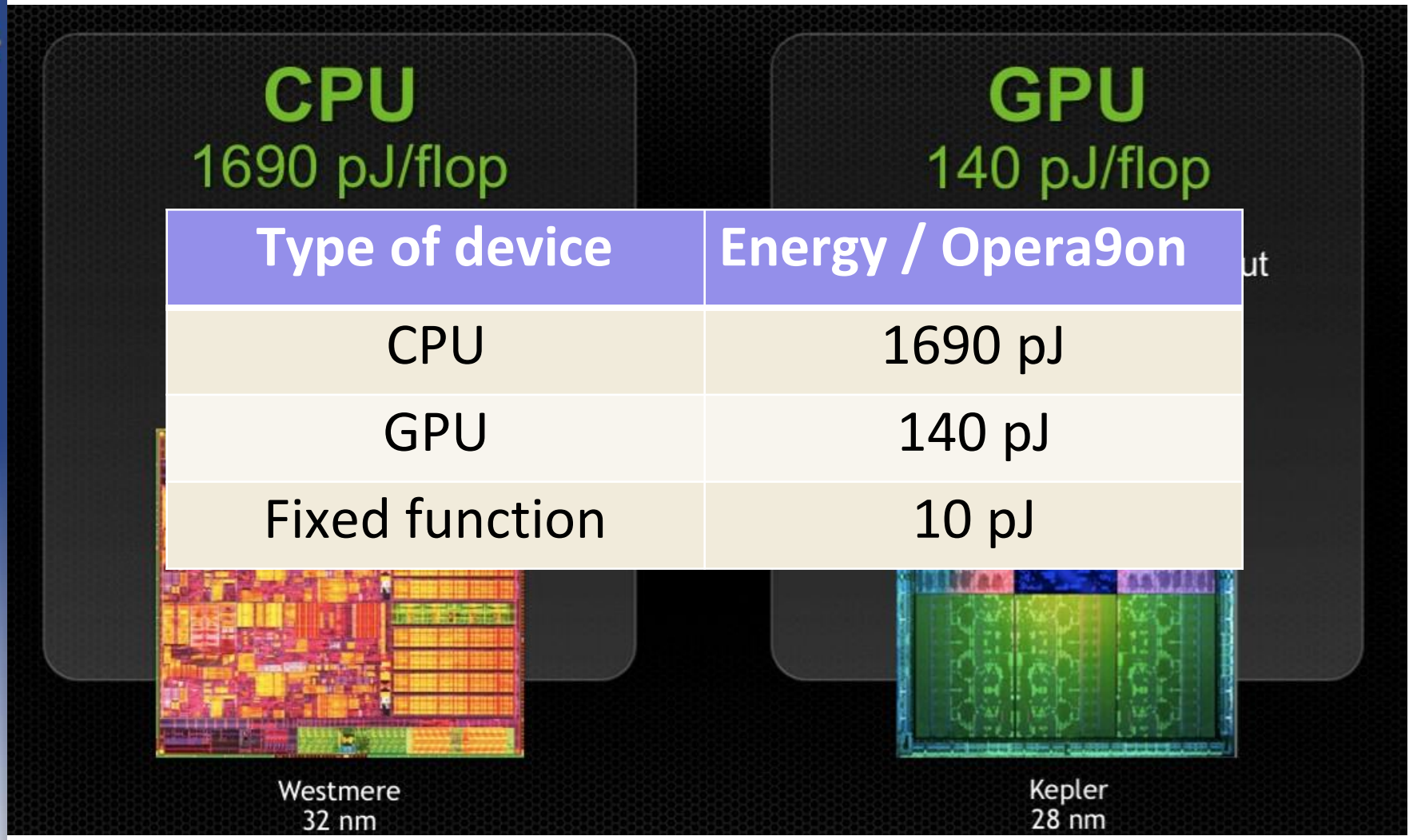


Source from C Moore, « Data Processing in ExaScale-Class Computer Systems », Salishan, April 2011

Power limits the active silicon area => more efficient specialized units



Specialization leads to more efficiency



Type of device	Energy / Opera9on
CPU	1690 pJ
GPU	140 pJ
Fixed function	10 pJ

Source from Bill Dally (nVidia) « Challenges for Future Computing Systems »
HiPEAC conference 2015

Potential other optimizations

$$P_{per\ unit} = C\,V^2\,f + T_{sc}\,V\,I_{peak} + V\,I_{leak}$$

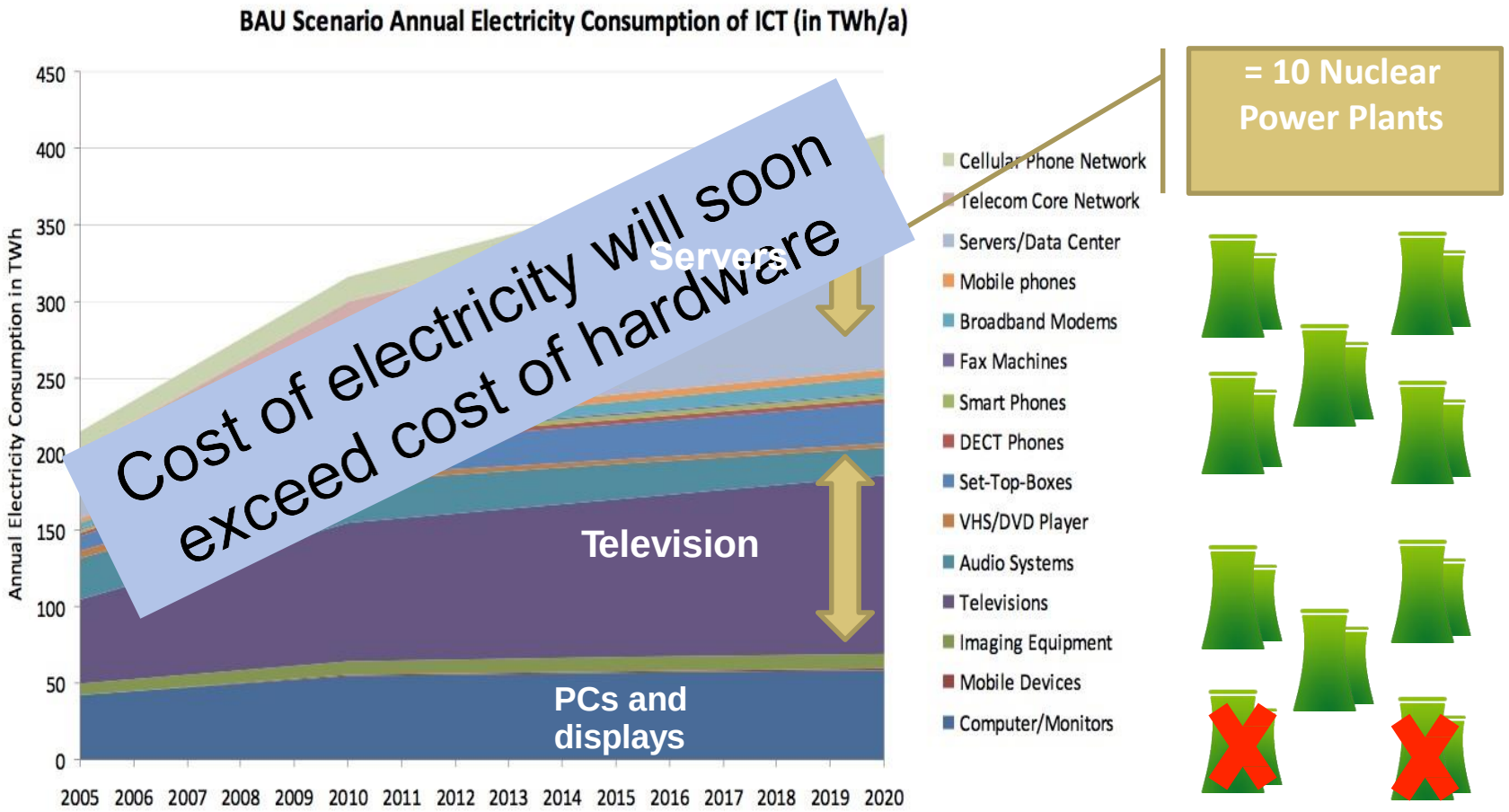
Average power, peak power, power density, energy-delay, ...

CIRCUITS	ARCHITECTURE	COMPILER, OS, APPLICATION
• Voltage scaling/islands • Clock gating/routing Clock-tree distribution, half-swing clocks • Redesigned latches/flip-flops pin-ordering, gate restructuring, topology restructuring, balanced delay paths, optimized bit transactions • Redesigned memory cells Low-power SRAM cells, reduced bit-line swing, multi-Vt, bit line/word line isolation/segmentation • Other optimizations Transistor resizing, GALS, low-power logic	• Voltage/freq scaling • Gating Pipeline, clock, functional units, branch prediction, data path • Split instrucn windows • SMT thread throttling	• Switching control Register relabeling, operand swapping, instruction scheduling • Memory access reduce Locality optimizations, register allocation • Power-mode-control
	• Bank partitioning • Cache redesign Sequential, MRU, hash-rehash, column-associative, filter cache, sub-banking, divided word line, block buffers, multi-divided module, scratch • Low-power states • DRAM refresh-control	• CPU/resource schedule • Memory/disk control Disk spinning, page allocation, memory mapping, memory bank control • Networking Power-aware routing, proximity-based routing, balancing hop count, … • Distributed computing Mobile agents placement, network-driven computation
	• Switching control Gray, bus-invert, address-increment • Code compression • Data packing/buffering	• Fidelity control • Dynamic data types • Power API

Source: P. Ranganathan, “System architectures for servers and datacenters »

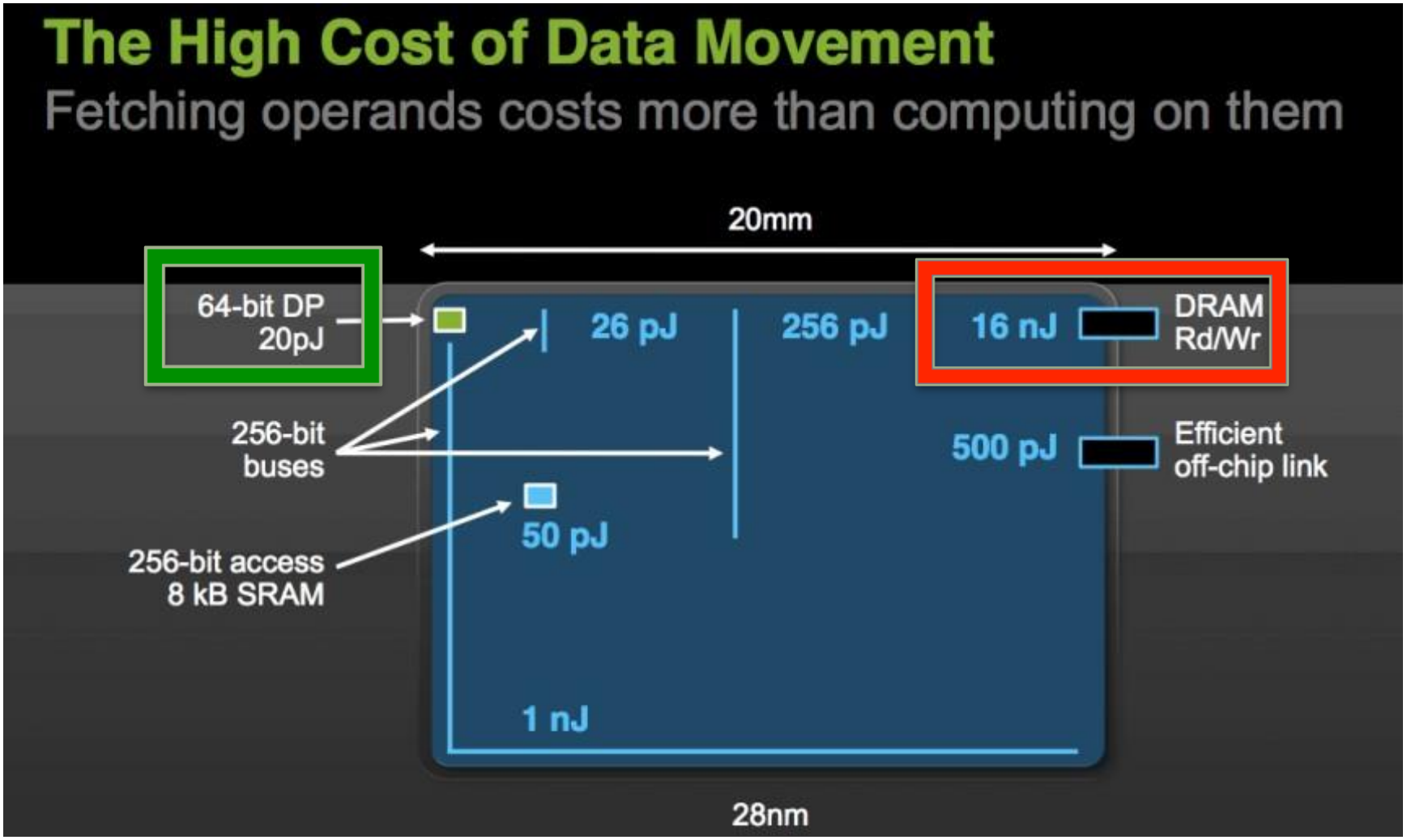
Energy consumption of ICT

- Estimated consumption 410 TWh in 2020, 25% for servers



Source: European Commission DG INFSO, Impact of Informa9on and Communica9on Technologies on Energy Efficiency, final report, 2008

Cost of moving data

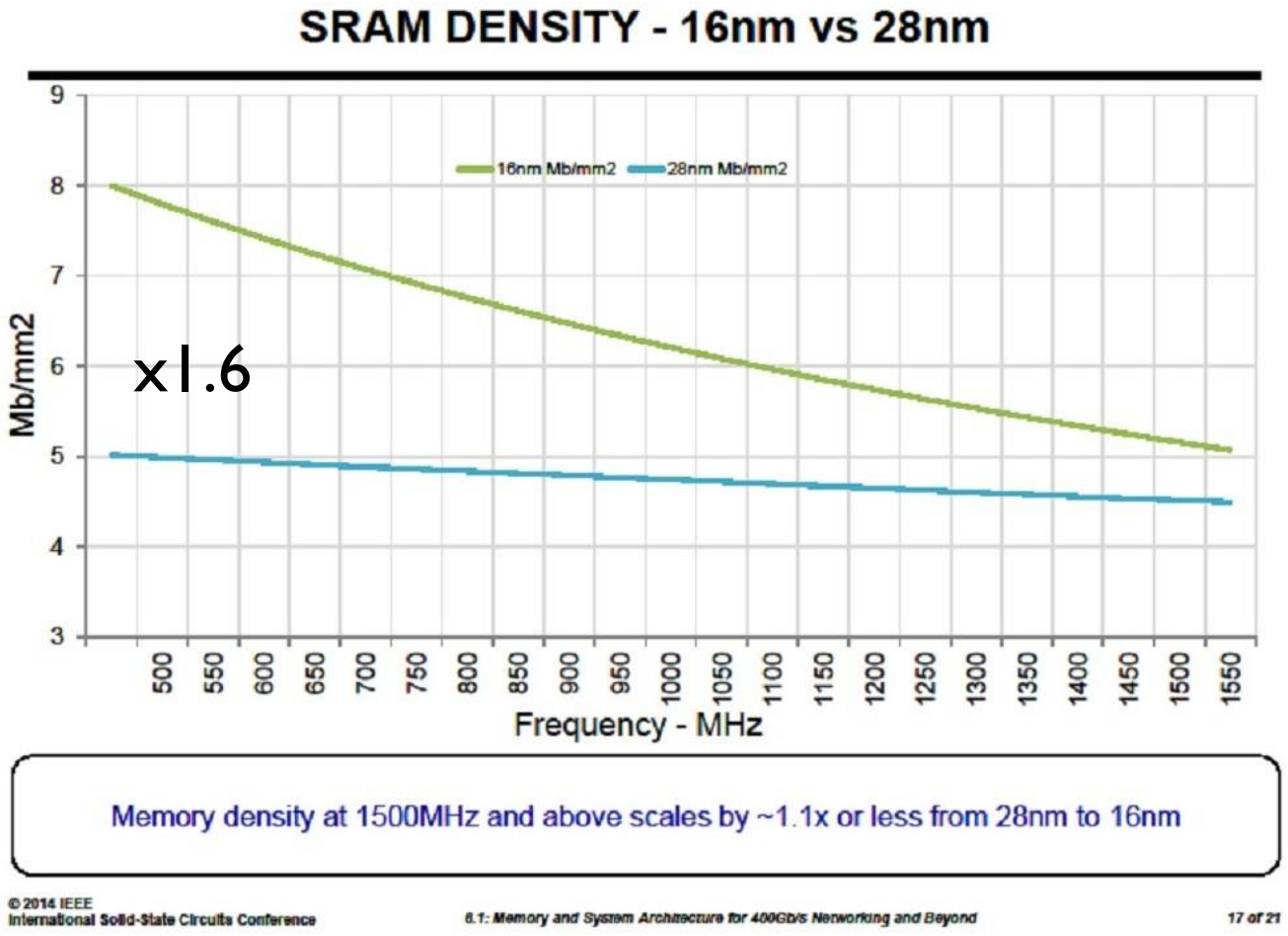


Source: Bill Dally, « To ExaScale and Beyond »

www.nvidia.com/content/PDF/sc_2010/theater/Dally_SC10.pdf

Performances of SRAM hardly increase

Node	45nm	16nm	14nm	10 nm
Density	150 F^2	2ti7 F^2	ti00 F^2	450 F^2



Source: Joel Hruska, « Stop obsessing over transistor counts: It's a terrible way of comparing chips », http://www.extremetech.com

SRAM takes more and more SoC area

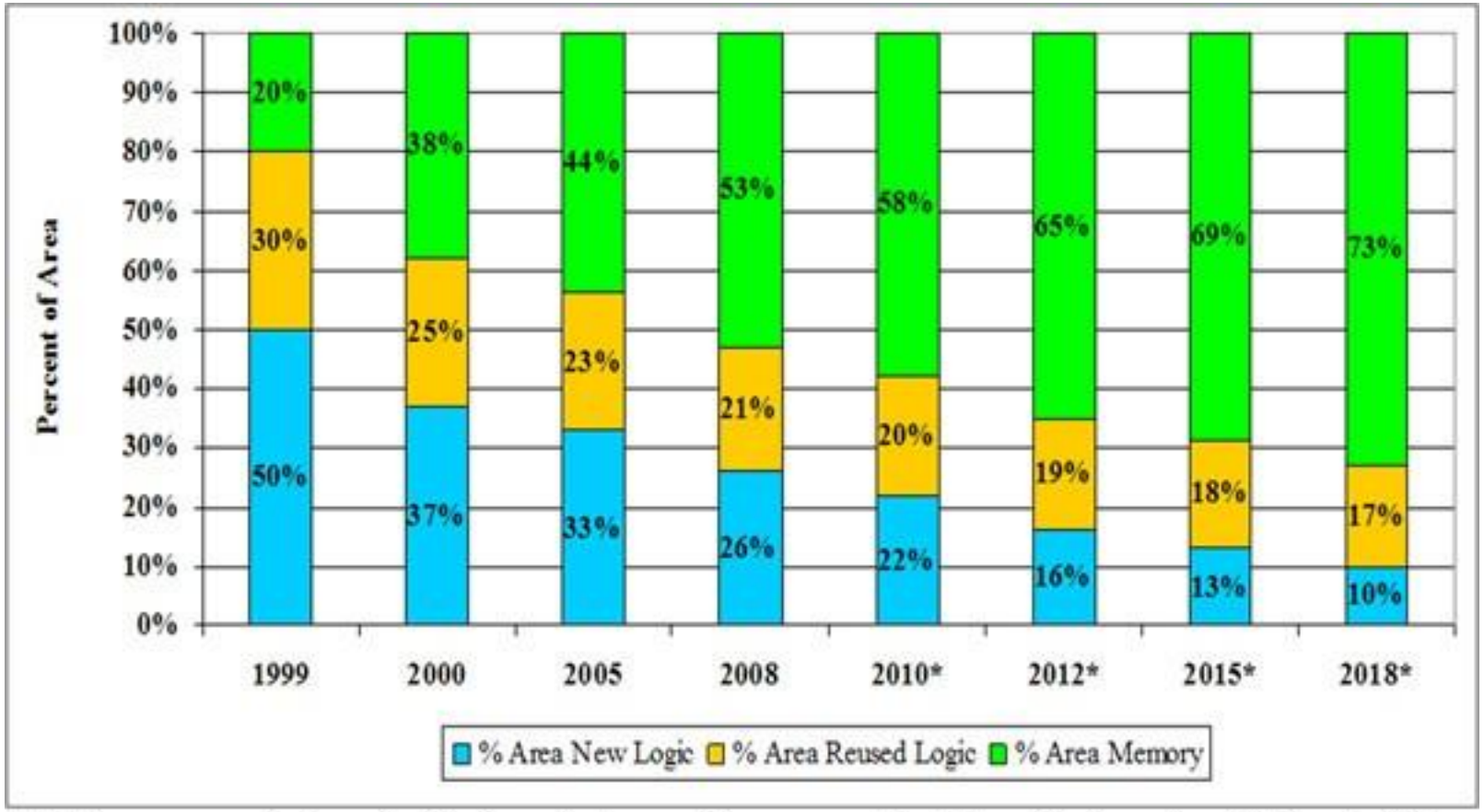


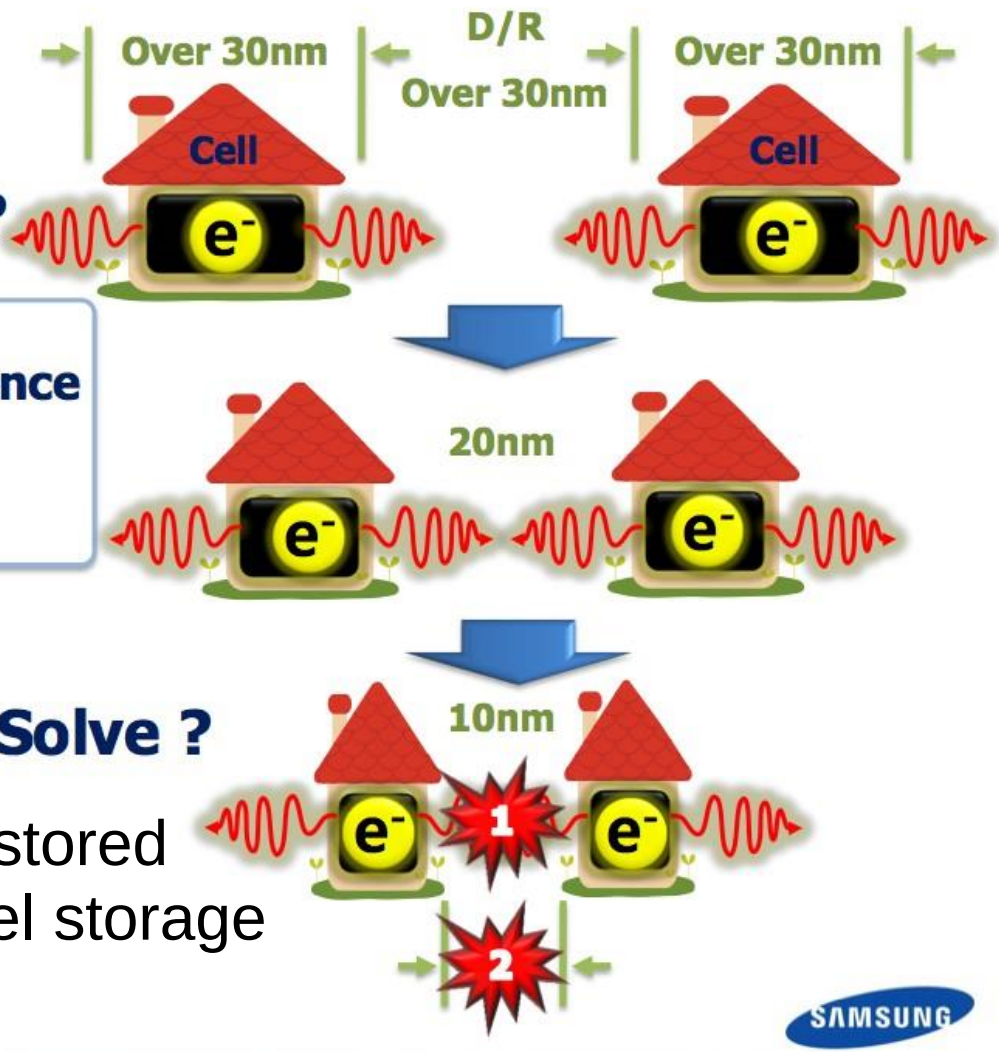
Fig 3. To compress design schedule time, designers often reuse earlier design blocks and use third party IP. It is very rare that a new chip featuring billions of transistors is designed completely from scratch. Generally, most of a new design's transistors are used to form memories or functions derived from similar functions implemented in earlier designs. (Source: Semico Research Corporation, Study Number SC103-10, October 2010)

Flash scaling also hits limits

2 Questions

Q1. Why so difficult ?

1. Cell to Cell Interference
2. Patterning



Q2. 3D V-NAND Can Solve ?

- At 20nm, about 70e- stored
- Vt not ok for multilevel storage

Flash Memory SUMMIT

SAMSUNG

Source: http://www.anandtech.com/show/72ti7/samsungs-vnand-hitting-the-reset-button-on-nand-scaling

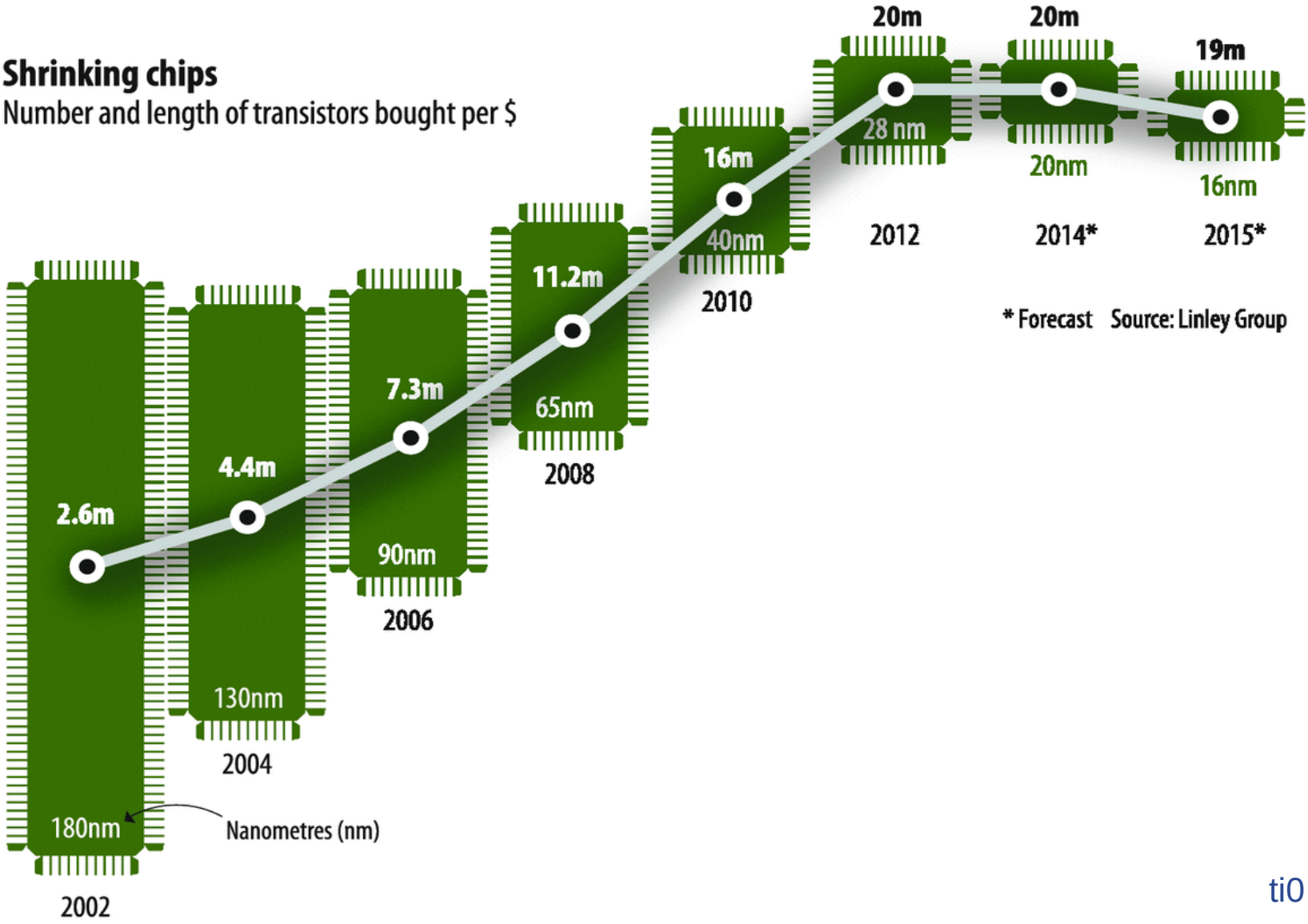
The future will be non volatile memories

But still in development and which technology will be the winner?

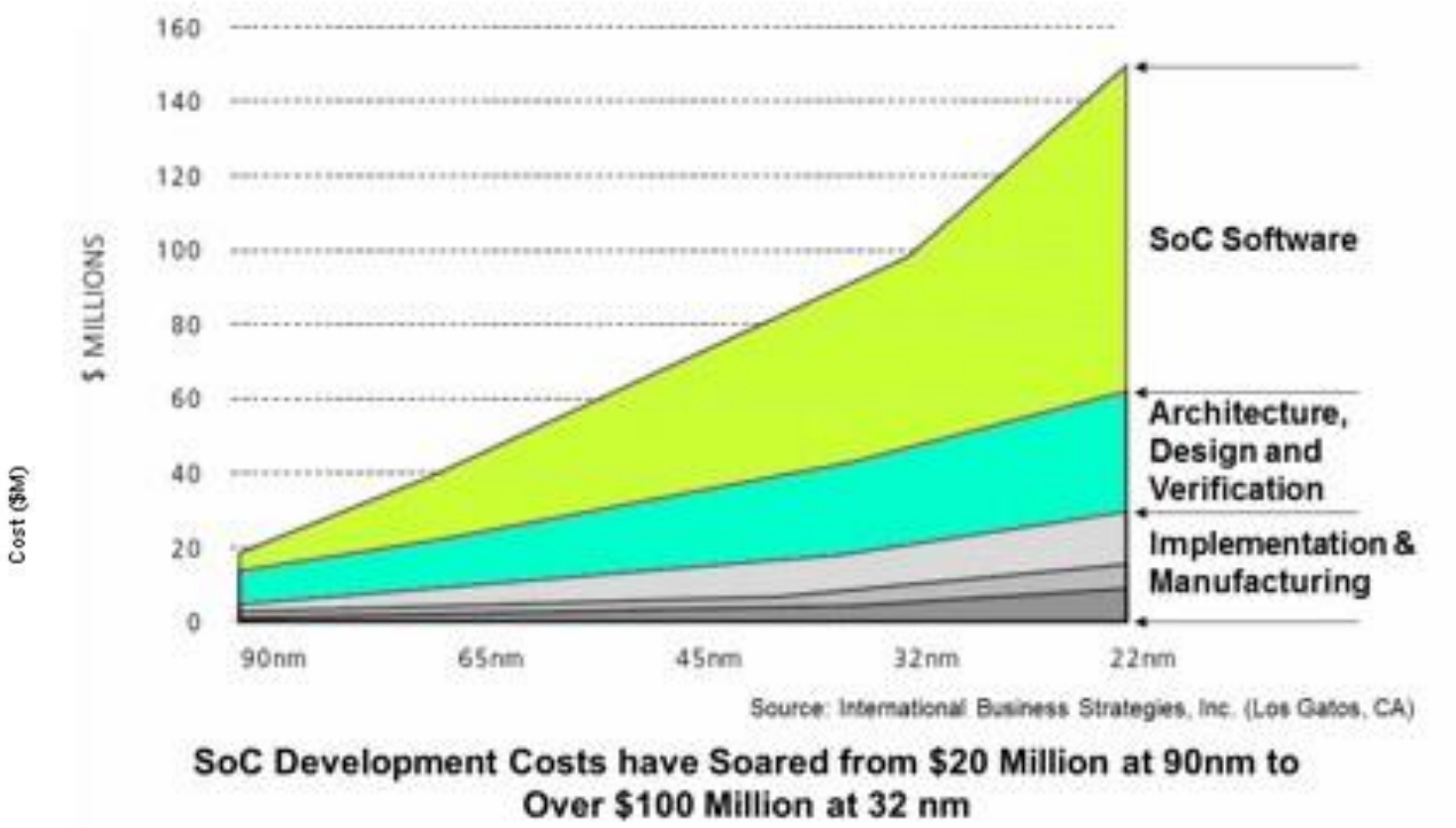
	FeRAM	RRAM	Magnetic field write MRAM	PRAM	STT MRAM
Non-volatile	Y	Y	Y	Y	Y
Memory cell factor (F^2)	16-32	4-6	16-32	5-8	5-7
Read time (ns)	20-50	10-20	3-20	5-20	3-15
Write/erase time (ns)	50	20	10-20	>30	3-15
Number of rewrites	10^{12}	10^{5}	10^{15} min	10^{12}	10^{15} min
Power consumption at write	Low	Low	Somewhat high	Low	Low
Required input voltage (V)	2-3	1.2	3	1.5-3	1.5

From Coughlin Associates

The cost per transistors is not decreasing anymore



And the development cost is increasing

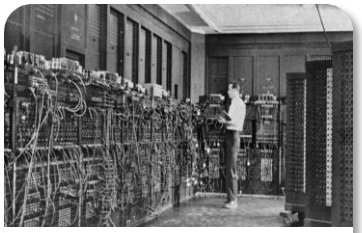


Rock's law: cost of IC plant doubles every 4 years

Reaching 10th or 100th of $ Billions…

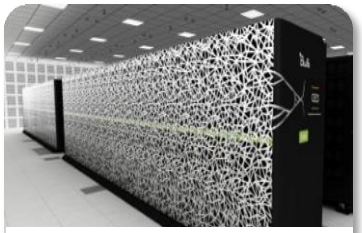
“Main drives in compuAng”

High Performance Computing

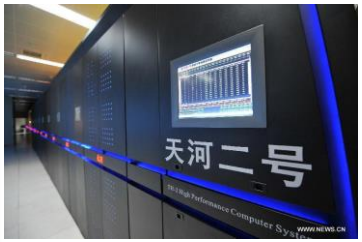


1946
ENIAC, vacuum tube computer, 5KOPS, 150KW

1965
General Electric GE6ti5, 4 processors, 2 MIPS



2012
Bull B510, 10K cores, 4TeraBytes RAM, 200 TFlops

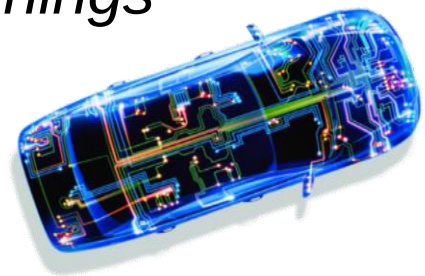


2015
Tianhe-2 (MilkyWay-2): Intel Xeon E5-2692 12C 2.200GHz, Intel Xeon Phi ti1S1P, ti.12M cores, 1,024 TB RAM, 50 PFlops, 17,8 MW

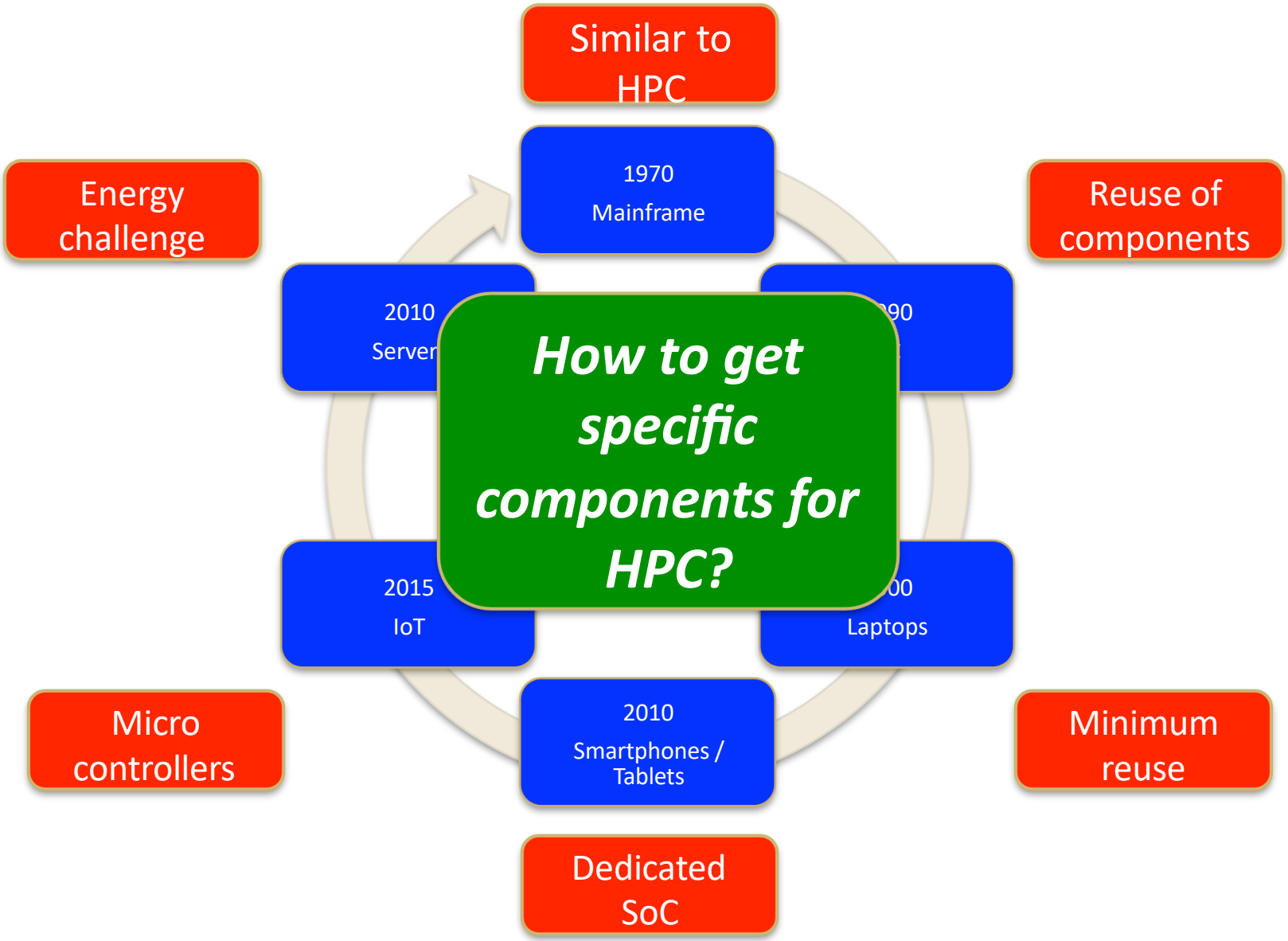
Yesterday

Today

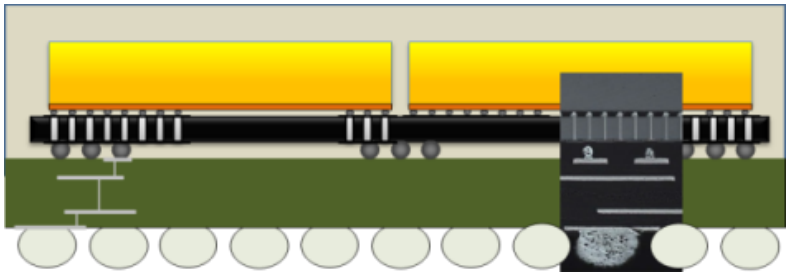
Things

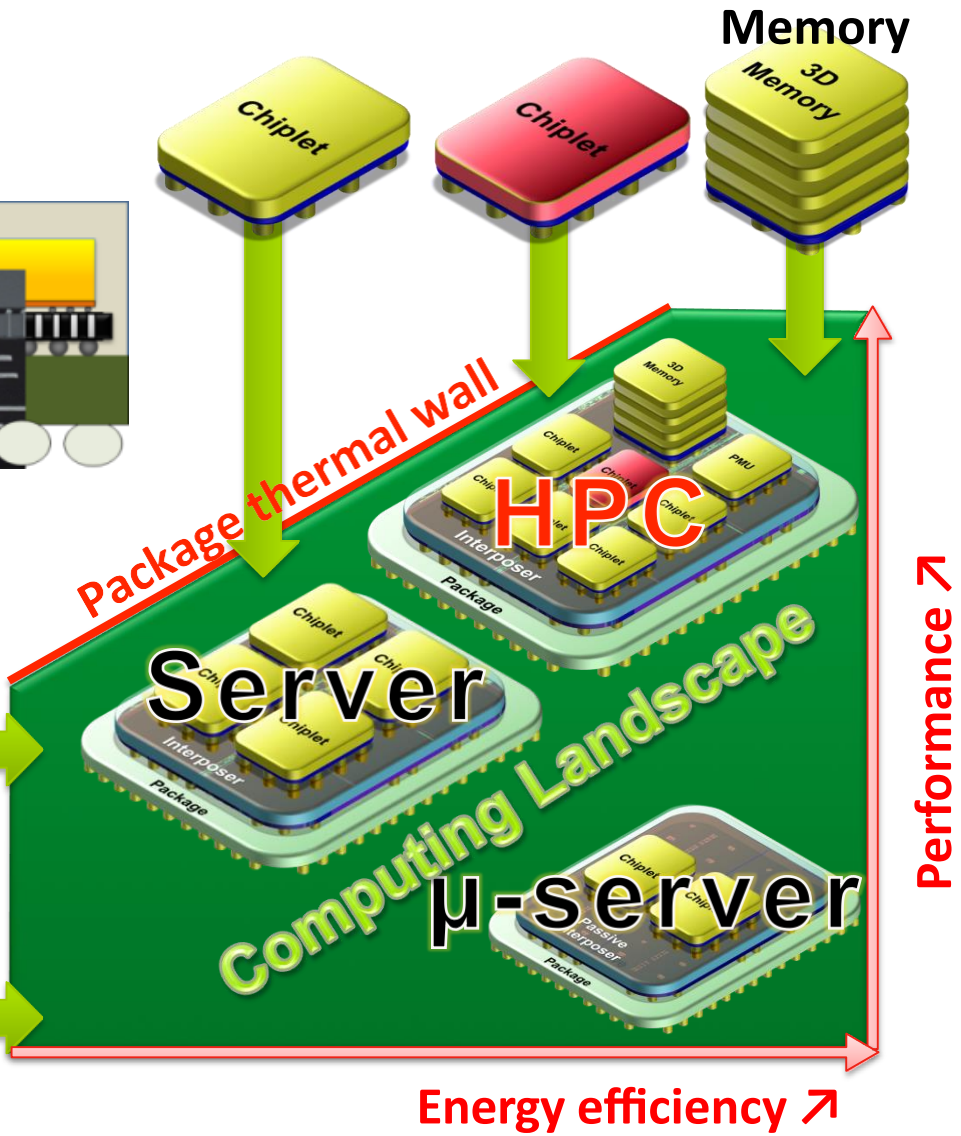


$$ = Fuel of innovation?
HPC: not anymore a drive for component developments



Specialization with interposer





⇨Lower cost:
- Improved yield
- Reduced NRE

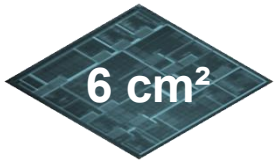
⇨Lower energy:
- Heterogeneity
- Shorter wires
- Photonic ready

Many cores: technology to reduce energy consumption and cost

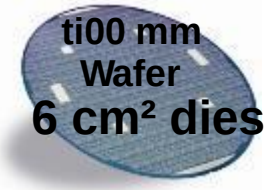
Traditional Chip

Large IC with maximum integration

1 die



5000 $ wafer cost
89 dies of 6 cm²
20% yield
→→281 $ die cost

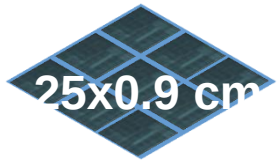


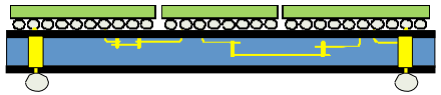
→→281 $ die cost

IC cost* 95% final test yield
→→296 $ IC cost

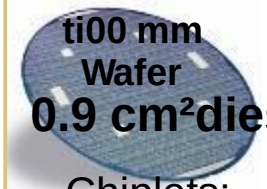
Multiples small dies

Small dies stacked on a large interposer





25 chiplets+ 1 interposer



Chiplets:
5000 $ wafer cost
714 dies of 0.9 cm²
80% yield
→→8.75 $ die cost

Interposer:
500 $ wafer cost
14 dies of 25 cm²
98% yield
→→ti6.44 $ die cost

→→255 $ total die cost

90% final test yield
→→284 $ tiD-IC cost

Same cost 6 cm² versus 25 cm²

2 GFLOPS/W — **More energy efficiency** — 10 GFLOPS/W

*: test and package costs are not included but considered equal for both technologies in this exercise

Together…

Electrons for compute

Electrons like to interact; easily moved; interaction needed for compute

+ Ions for storage

Ions like to interact; stay put; good for storage

+ Photons to communicate

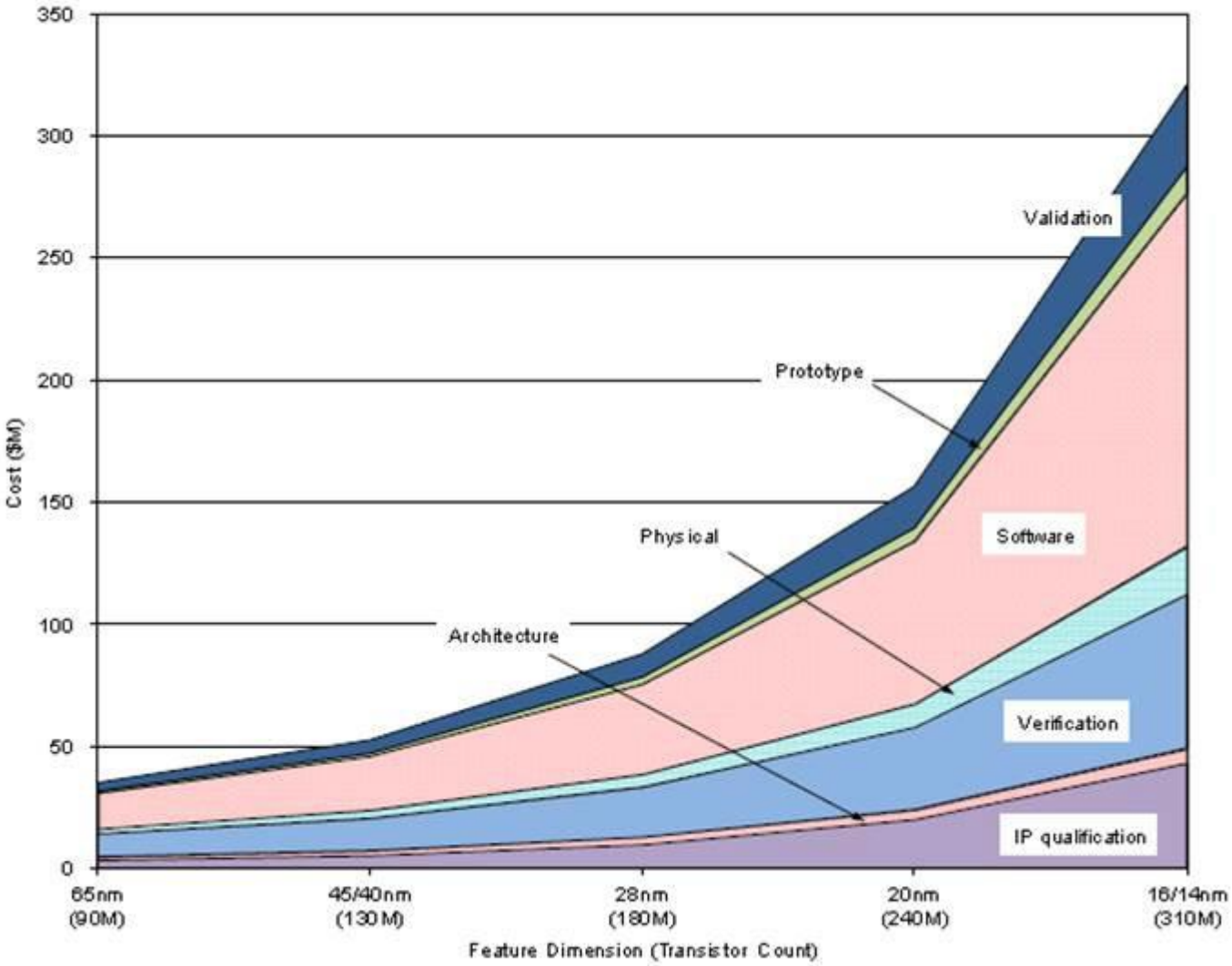
Photons don't like to interact or stay put; good for long-distances

See the presentation on "The Machine" from HP

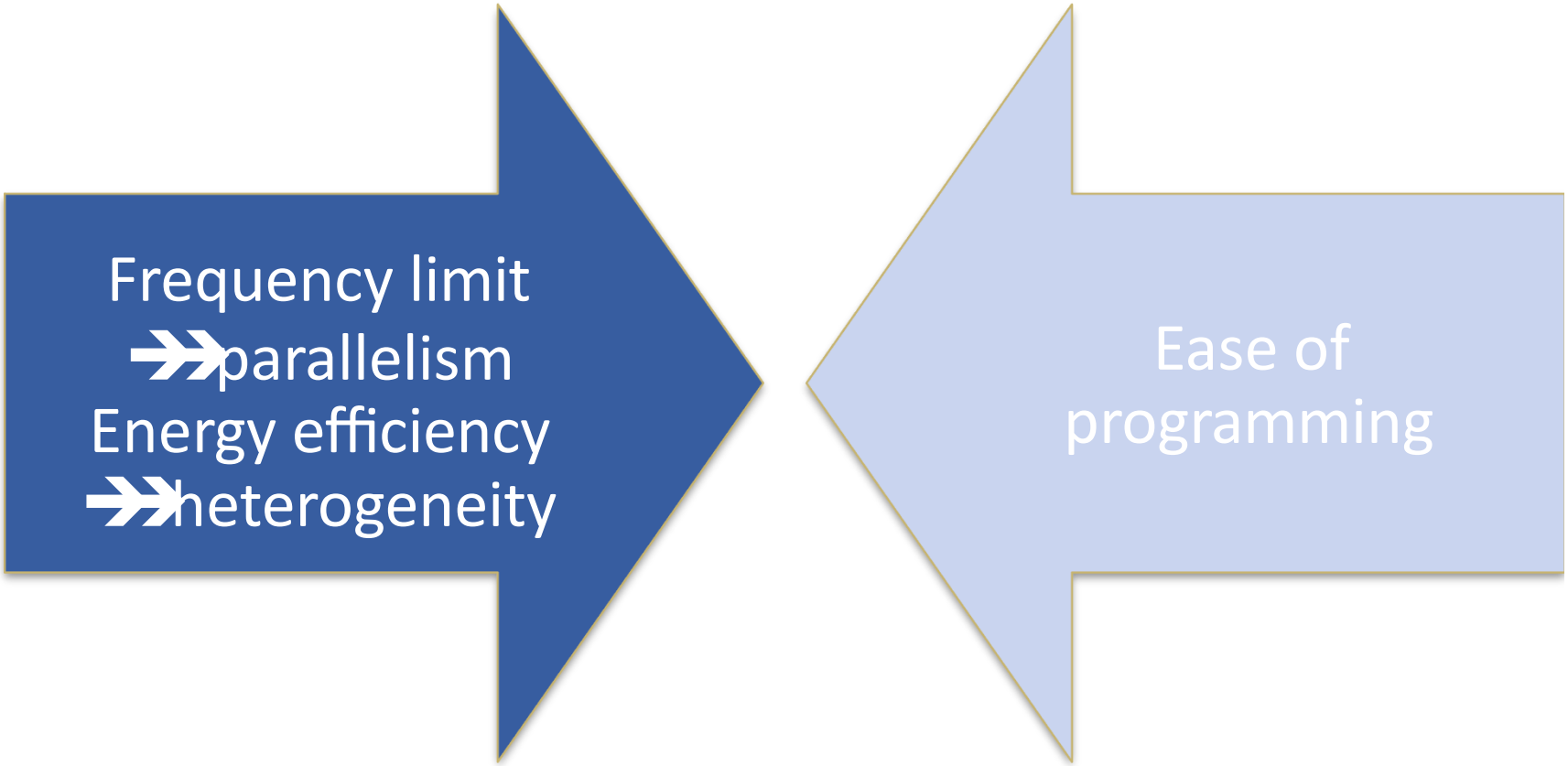
Courtesy: Jouppi2011

Source: P. Ranganathan, "Saving the world together, one server at a time…" ACACES 2011

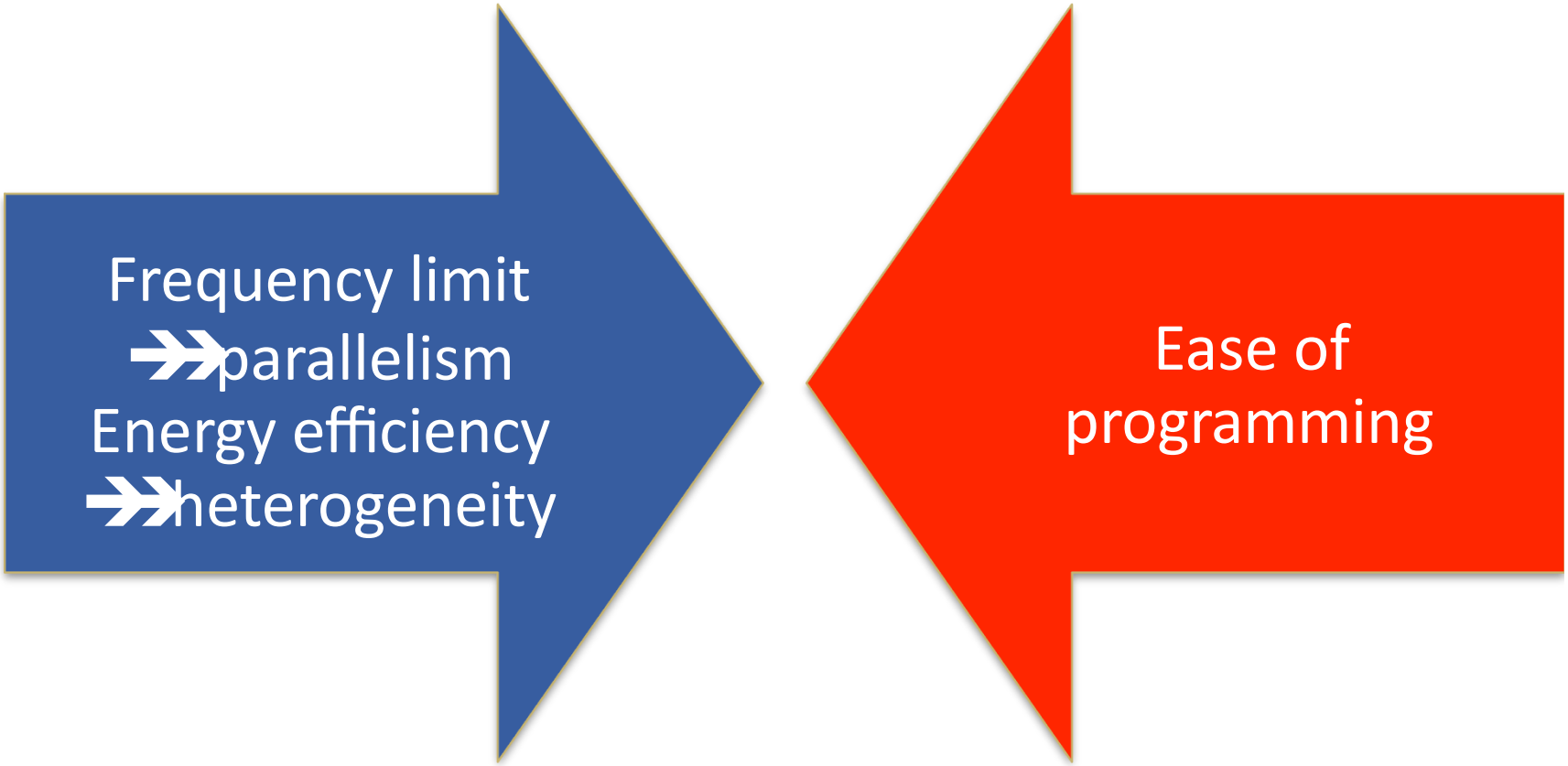
Software cost is rapidly increasing



Parallelism and specializaAon are not for free…

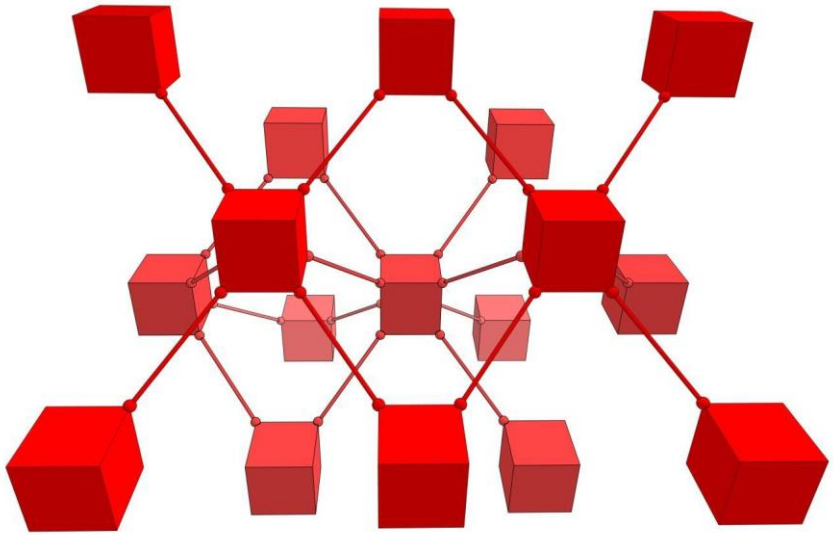


Parallelism and specializaAon are not for free…



Managing complexity….

"Nontrivial software written with threads, semaphore, and mutexes is ***incomprehensible*** *by humans"*



Edward A. Lee

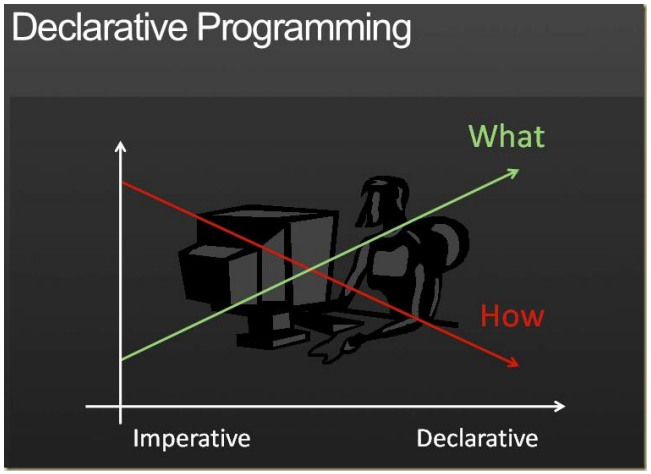
The future of embedded software

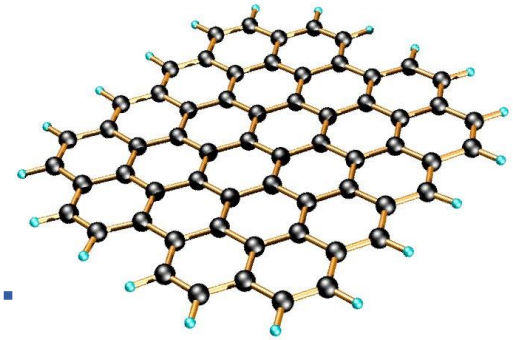
ARTEMIS 2006

Parallelism seems to be too complex for humans ?

Time to think differently?

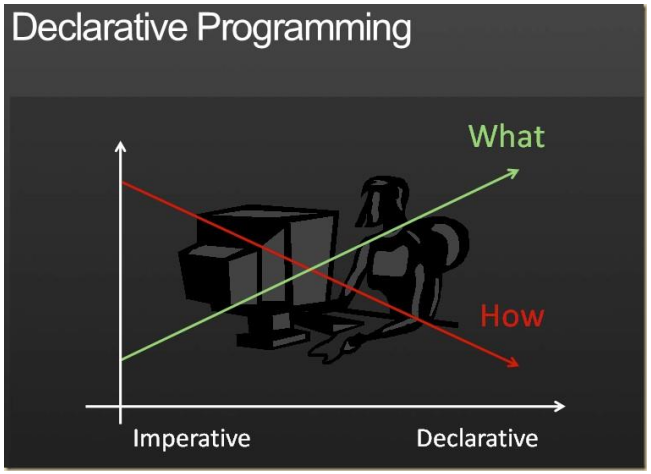
- Approximate computing
- Probabilistic CMOS
- Neuromorphic computing
- Declarative programming
- Graphene
- Spintronic
- Quantum…

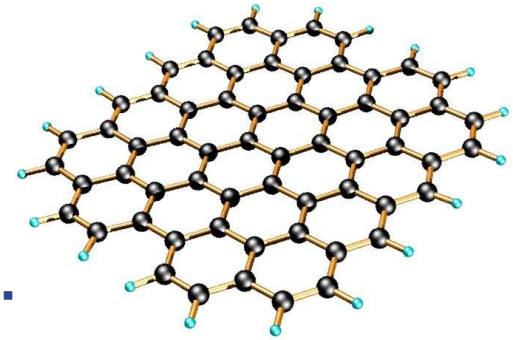




Time to think differently?

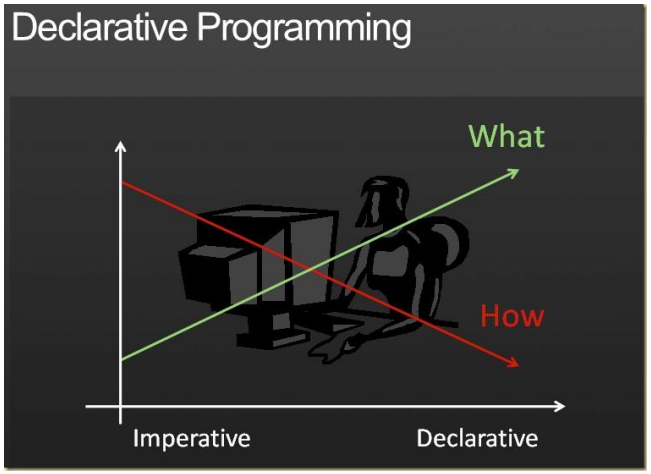
- ~~Approximate~~ computing
- Probabilistic CMOS
- Neuromorphic computing
- Declarative programming
- Graphene
- Spintronic
- Quantum…

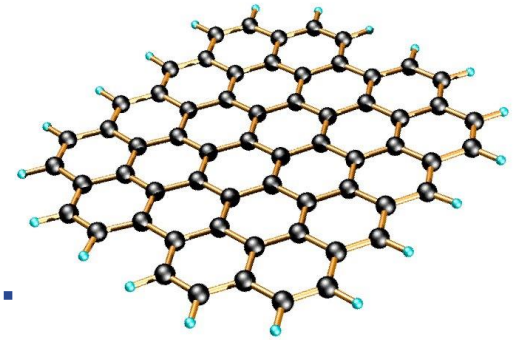




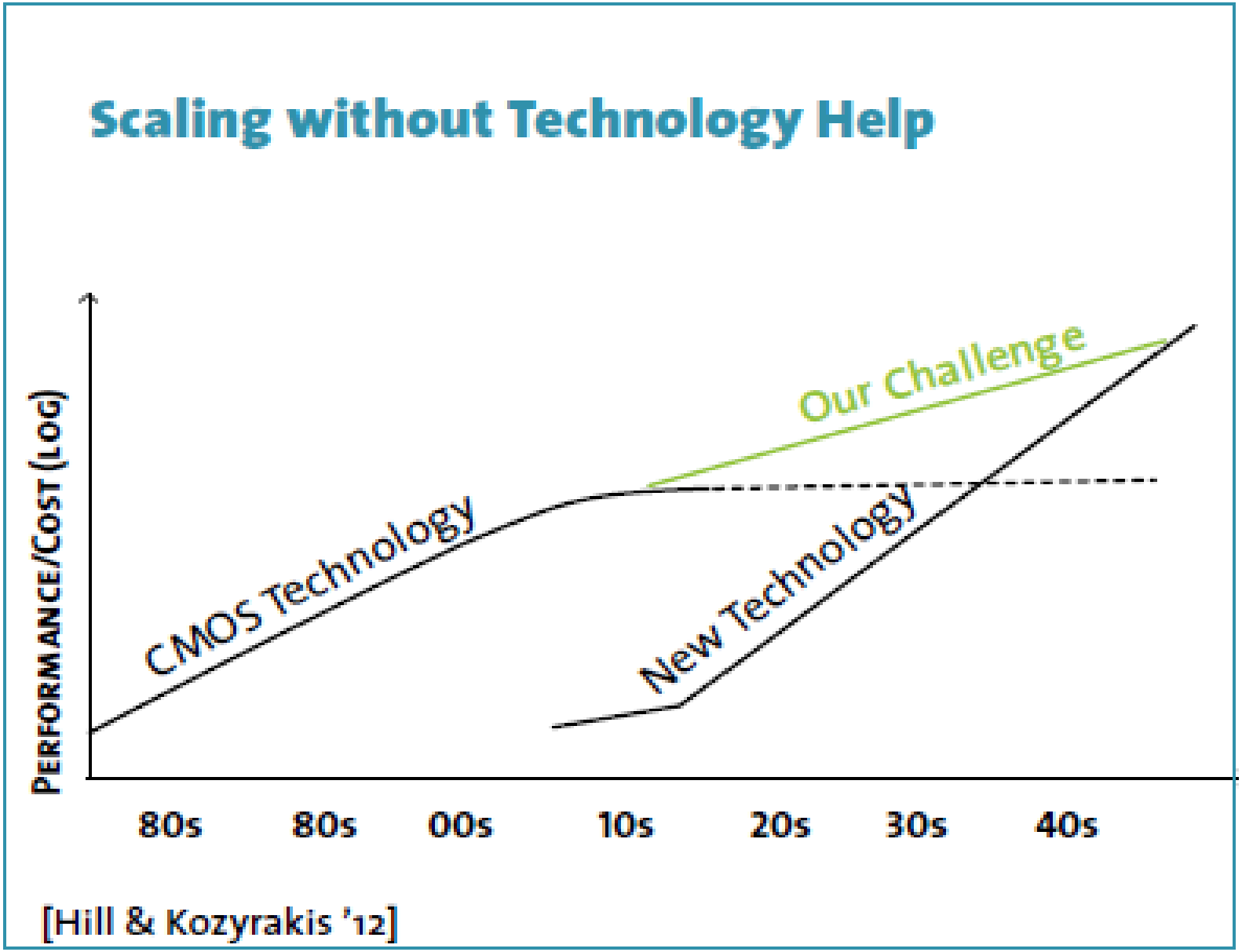
Time to think differently?

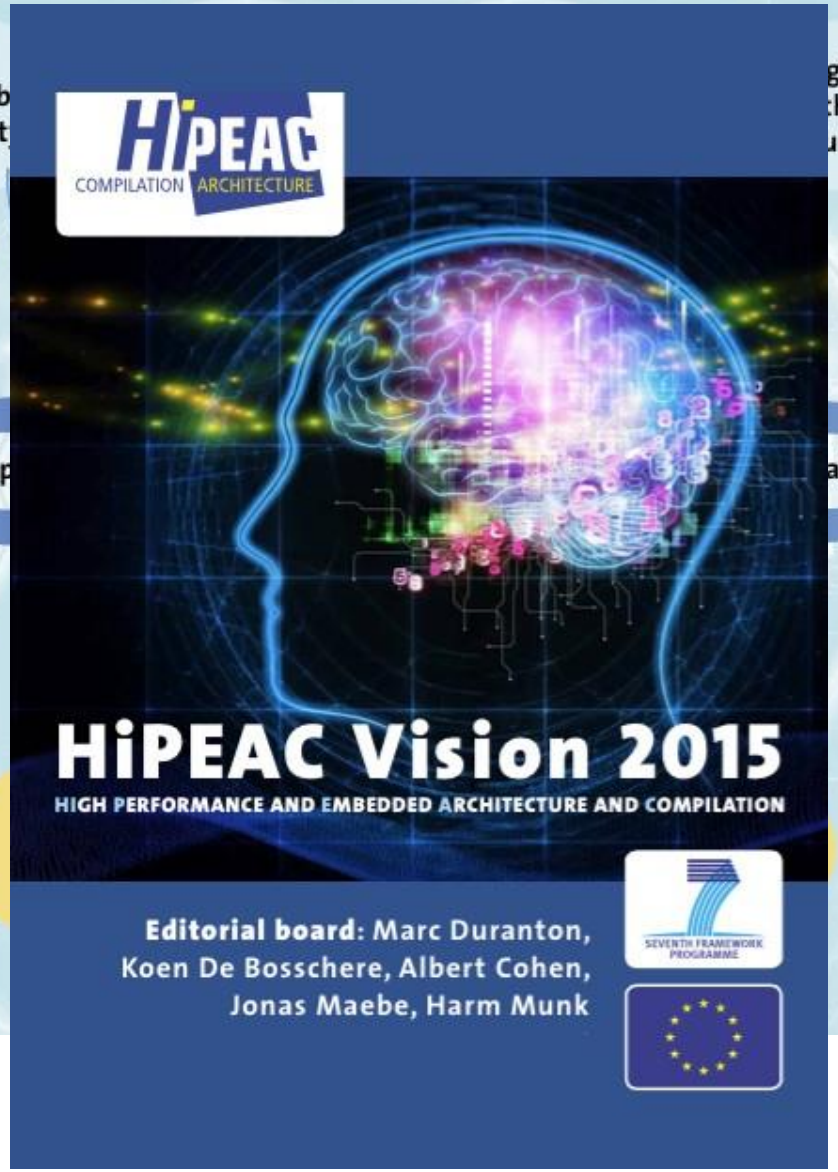
- Adequate computing
- Probabilistic CMOS
- Neuromorphic computing
- Declarative programming
- Graphene
- Spintronic
- Quantum…





We are entering a transition period…





http://www.hipeac.org/vision