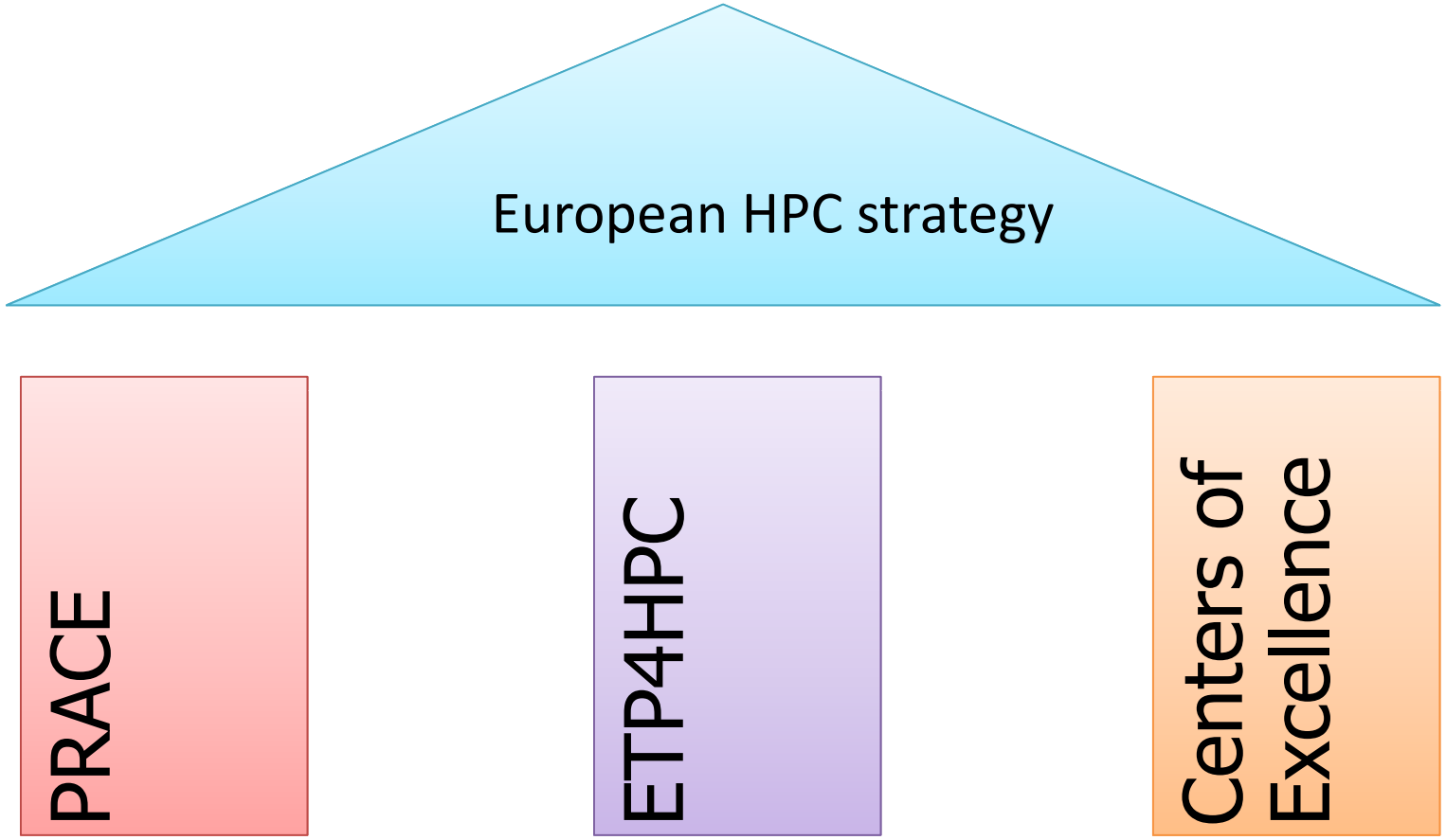
Trends in HPC

Carlo Cavazzoni, CINECA


European HPC strategy
PRACE
ETP4HPC
Centers of Excellence

CINECA road map
2016-2021

2016-2019 : Marconi

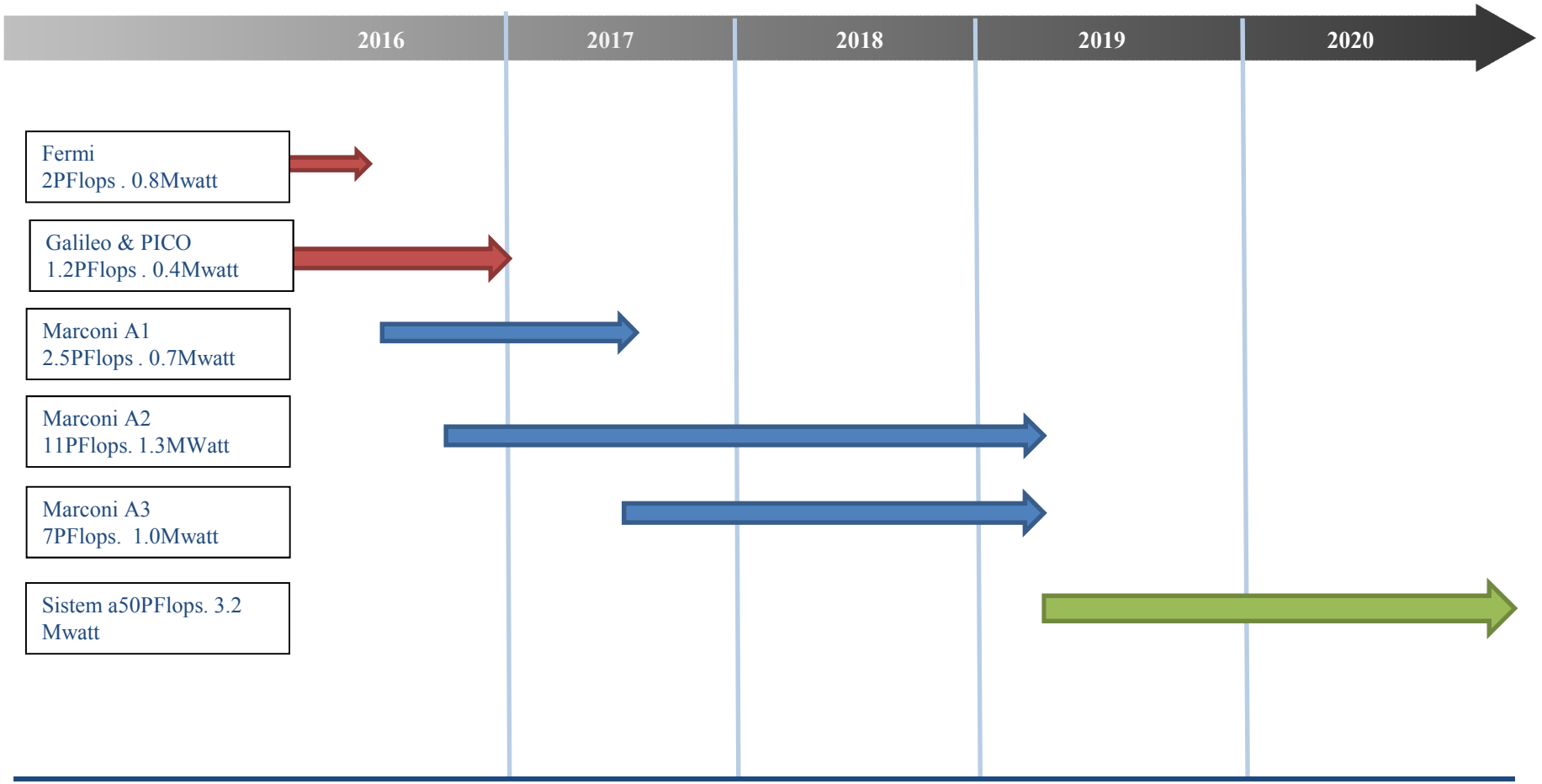
- > 18PFlop
- 100 racks
- 2.3MWatt
- 3Phases/partitions

Partitions	Availability	# di Racks	Power
A1 – Broadwell (2.1PFlops)	April2016	25	700KW
A2 - Knight Landing (11 Pflops)	July 2016	50	1300KW
A3 – Skylake (7PFlops)	June 2017	30	1000KW

2019-2020 : 50PFlops

- > 50PFlop
- 130 racks
- 3.2 MWatt

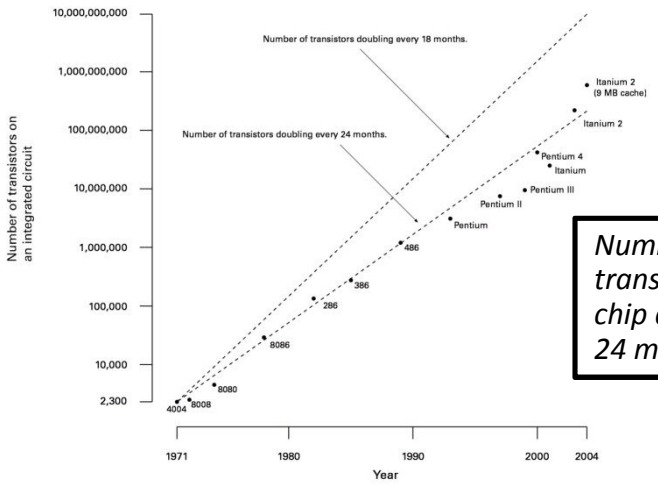
Partitions	Availability	# di Racks	Power
Non conventional proc. (40PFlops)	2019	80	2400KW
Standard procs- (10PFlops)	2019	28	800KW



1.2Mwatt	2.4Mwatt	2.3Mwatt	2.3Mwatt	3.2Mwatt
50 rack	120 rack	120 rack	120 rack	150 rack
100mq	240mq	240mq	240mq	300mq

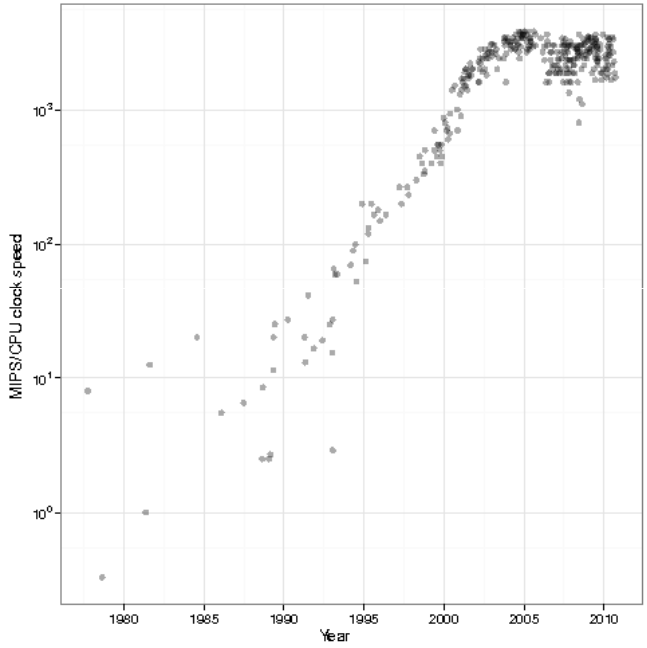
HPC Trends

Moore's Law



Number of transistors per chip double every 24 month

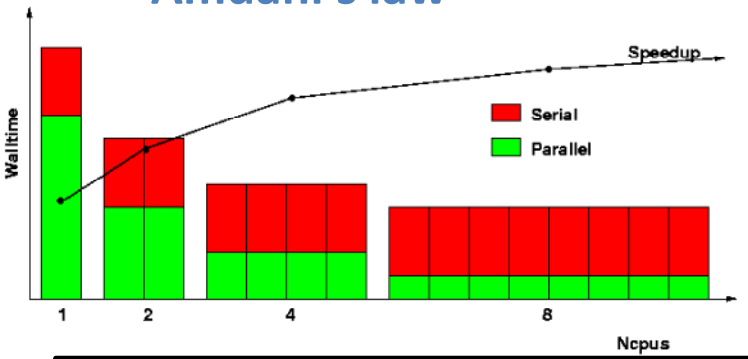
Dennard scaling law (downscaling)



The core frequency and performance do not grow following the Moore's law any longer

Increase the number of cores to maintain the architectures evolution on the Moore's law

Amdahl's law



maximum speedup tends to $1 / (1 - P)$
P= parallel fraction

The upper limit for the scalability of parallel applications is determined by the fraction of the overall execution time spent in non-parallel operations.

HPC Trends

Peak Performance	Exaflops 10^18	Moore law
FPU Performance	Gigaflops 10^9	Dennard law
Number of FPUs	10^9	Moore + Dennard
App. Parallelism	serial fraction 1/10^9	Amdahl's law
System TCO	Power >10^6 Watt	Moore + Dennard

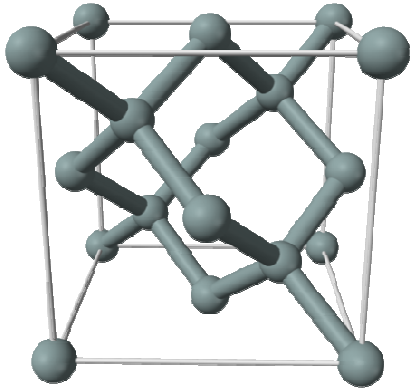
opportunity

challenge

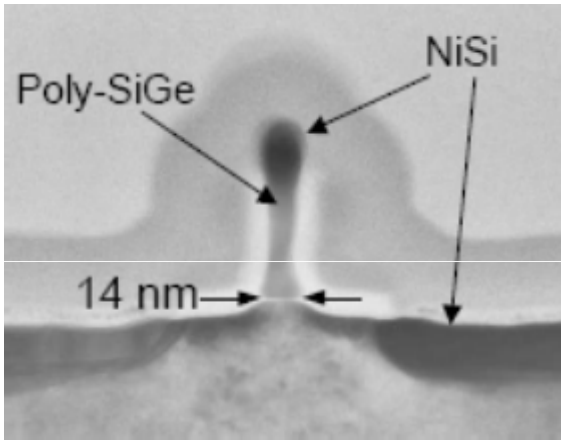
PRACE

CINECA

The silicon lattice



Si lattice



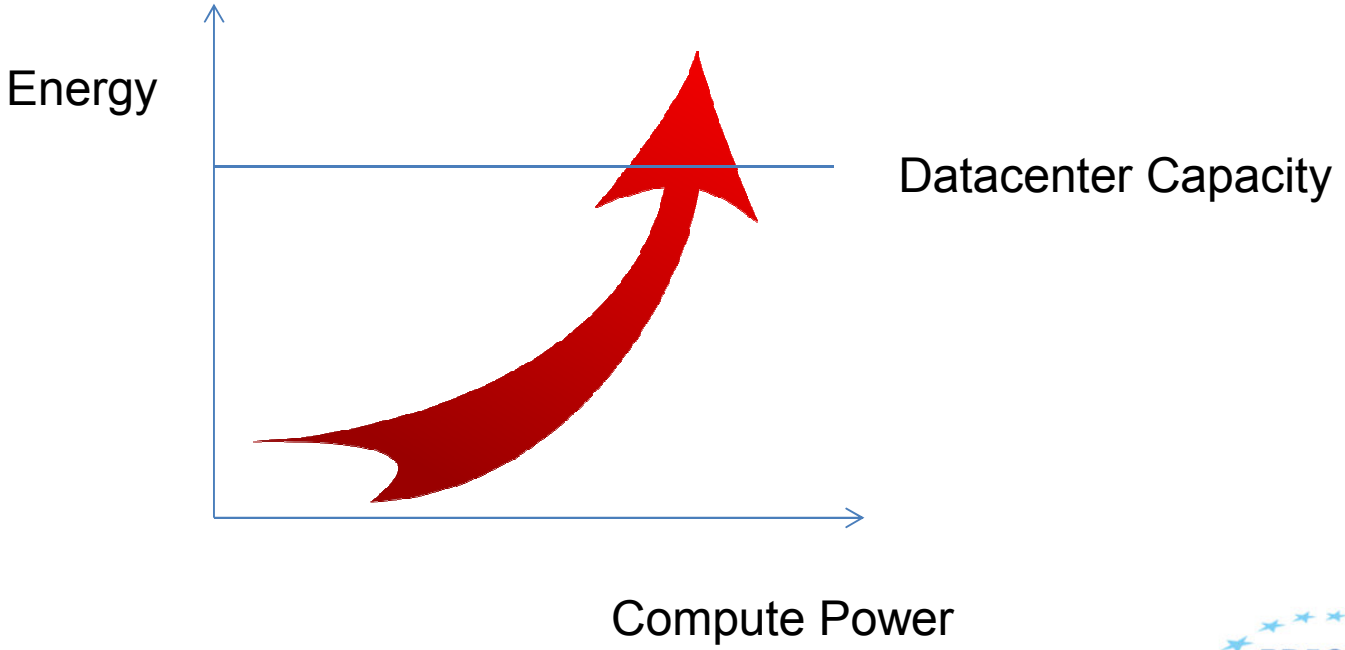
50 atoms!

Oh-oh! Huston!

Energy trends

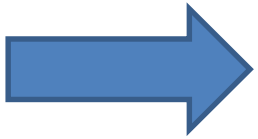
"traditional" RISK and CISC chips are designed for maximum performance for all possible workloads

A lot of silicon to maximize single thread performace

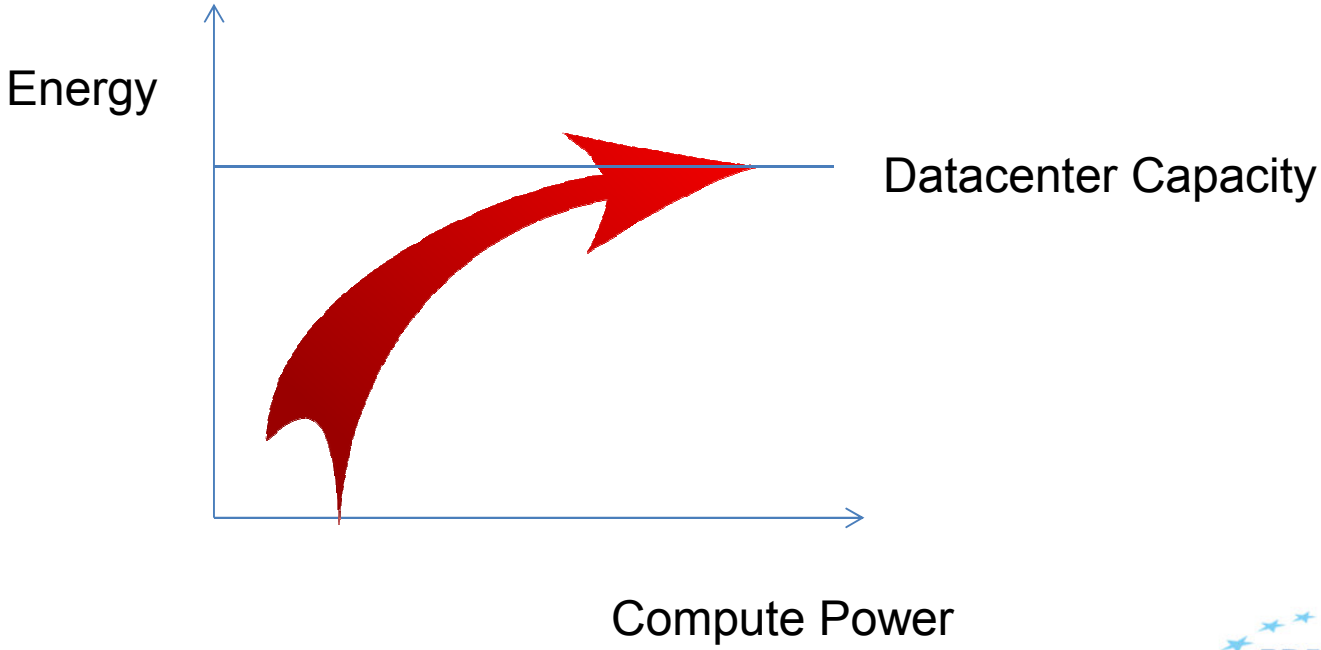


Change of paradigm

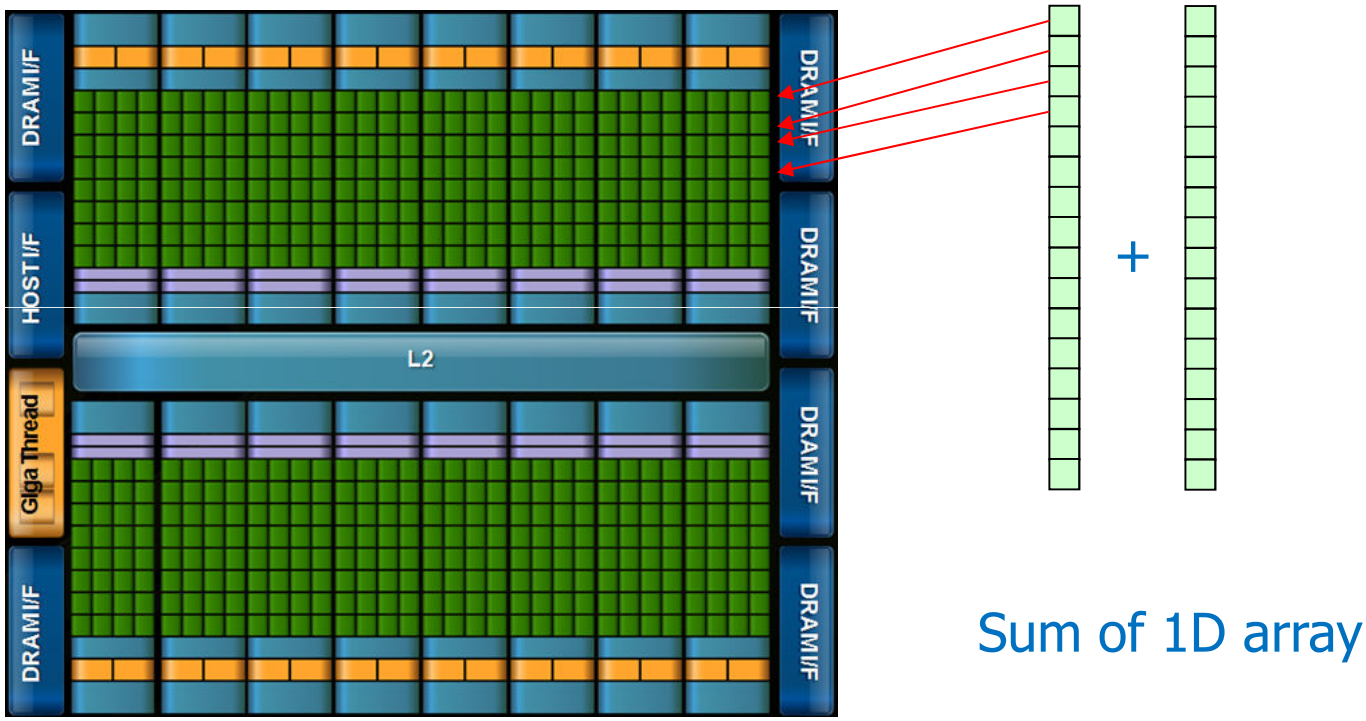
New chips designed for maximum performance in a small set of workloads



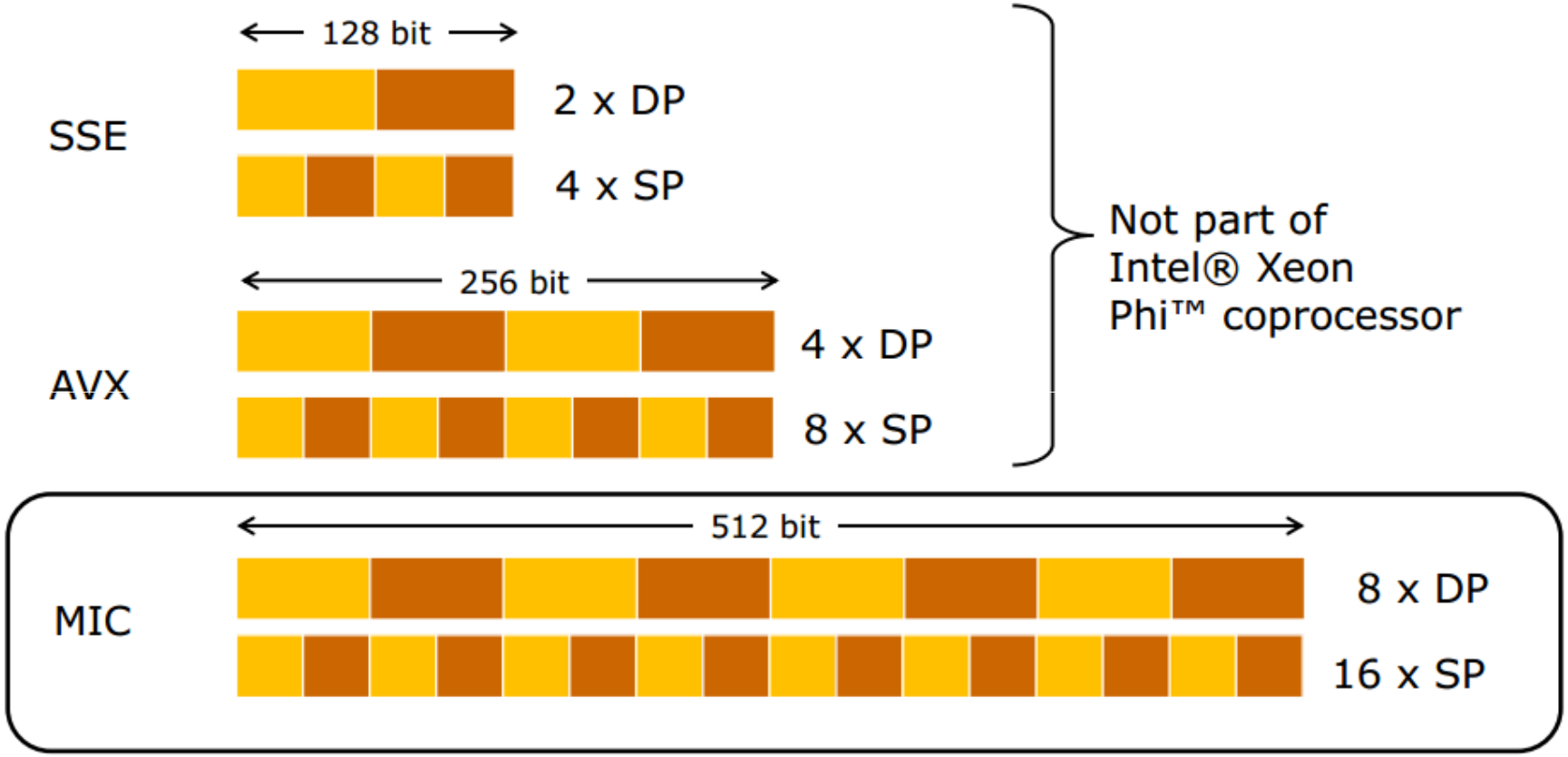
Simple functional units, poor single thread performance, but maximum throughput



Accelerator/GPGPU



Intel Vector Units



New AVX512 -> 2 vector FPU/core -> 32Flop/Clk

GPU vs MIC

- At low level (innermost loops) in both cases we need to expose an high level of parallelism
- At high level GPUs push for offloading programming model, with “micro”-tasking
- At high level MIC push for a task level programming model, with task nesting
- Both need much less synchronizations at MPI level

Applications Challenges

- Programming model
- Scalability
- I/O, Resiliency/Fault tolerance
- Numerical stability
- Algorithms
- Energy Awareness/Efficiency