Challenging the Common "Myths" About Exascale Computing

Steve Pawlowski
Intel Senior Fellow
GM, Central Architecture and Planning
CTO, Intel Architecture Group

July 14th 2010

Legal Disclaimer

Today's presentations contain forward-looking statements. All statements made that are not historical facts are subject to a number of risks and uncertainties, and actual results may differ materially.

NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL® PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. INTEL PRODUCTS ARE NOT INTENDED FOR USE IN MEDICAL, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS.

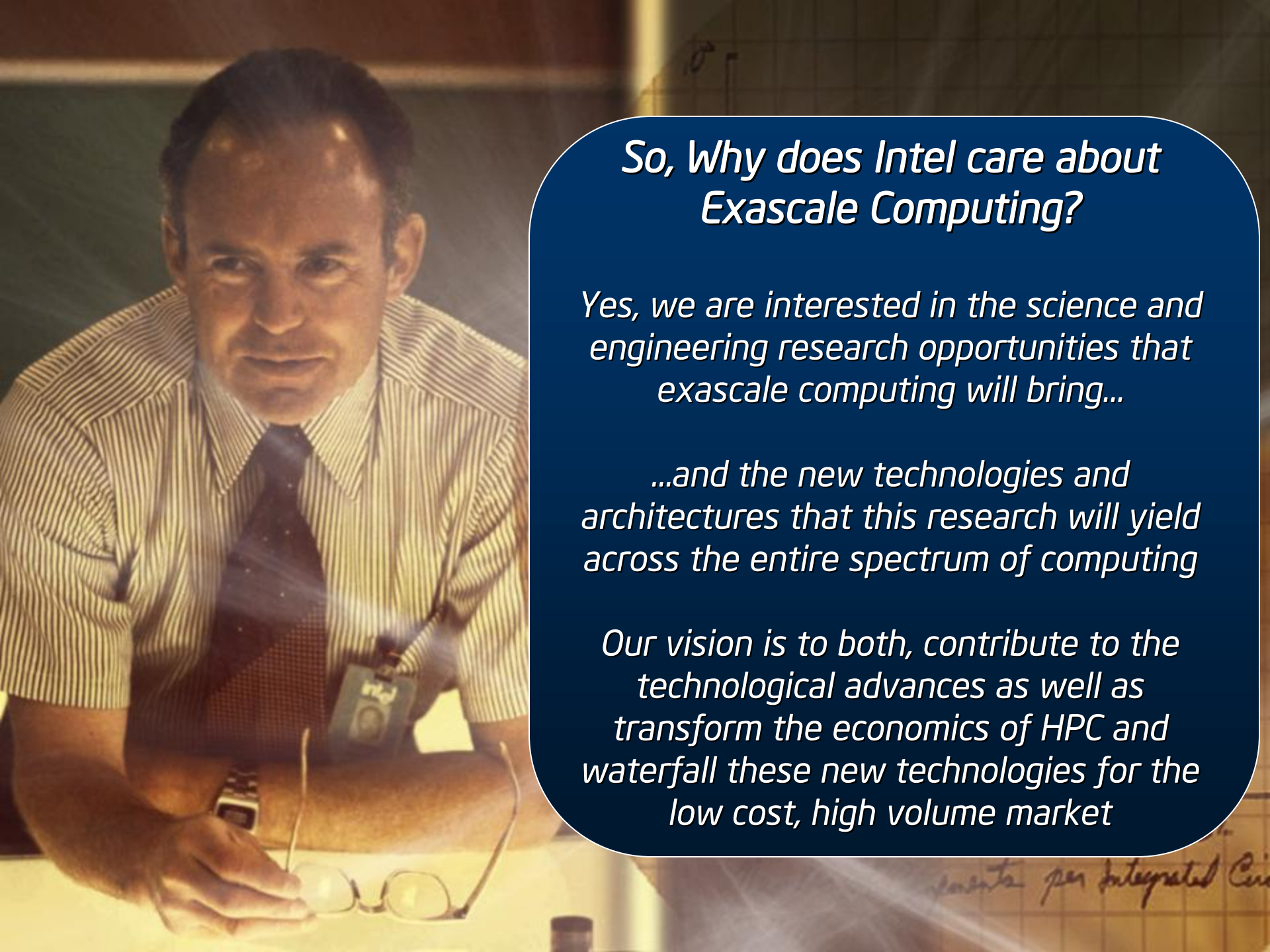
Intel does not control or audit the design or implementation of third party benchmarks or Web sites referenced in this document. Intel encourages all of its customers to visit the referenced Web sites or others where similar performance benchmarks are reported and confirm whether the referenced benchmarks are accurate and reflect performance of systems available for purchase.

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. See www.intel.com/products/processor_number for details.

Intel, processors, chipsets, and desktop boards may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel, Intel Xeon, Intel Core microarchitecture, and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
*Other names and brands may be claimed as the property of others

Copyright © 2010, Intel Corporation. All rights reserved.



So, Why does Intel care about Exascale Computing?

Yes, we are interested in the science and engineering research opportunities that exascale computing will bring…

…and the new technologies and architectures that this research will yield across the entire spectrum of computing

Our vision is to both, contribute to the technological advances as well as transform the economics of HPC and waterfall these new technologies for the low cost, high volume market

Myth 1:

We don't need an exascale? We could do just as well with fifty 20PFlop machines

Myth 1: No Need for Exascale Research – Just Use Fifty 20PF Systems

Let's test this myth with a simple example knowing that the max power for feasible deployment is **40MW:**

- Power Dissipated by #1 system on Top500 = 6.95MW
- Sustained performance of # 1 system = 1.759PF
- #1 systems required to get to Exaflops = 568
- Power dissipated by the "Exascale" system today = 4GW !
- Assuming traditional 18% Perf/Power CAGR over 10 years for this system, Projected Power for this "Exascale" system = 530MW

Reality: False

- Power, Failure Rates, and Cost to build this system disprove this myth

Source: http://www.top500.org/list/2009/11/100

Myth 2:

There is no trickle down of HPC technology to industry

Myth 2: No *trickle down of HPC technology to industry*

Let's test this myth by looking at some historical trends and making an educated extrapolation of the expected exascale design:

- Performance of #1 system on Top 500 in 1993: 59.7GF
- 17 years later…
  …Today's desktop CPU performance : 52GF
  …Today's mobile CPU performance: 28GF

If designed correctly, a 40MW Data Center Exascale system would
…provide a 40KW Petascale Rack Sized System and
…provide a 40W Terascale Embedded Sized System

Reality: False

- Economies of scale will force the trickle down (*and bubble up from Ultra Mobile*)
- Developing *low cost, powerful* rack/embedded systems will push software to trickle down to support mainstream applications

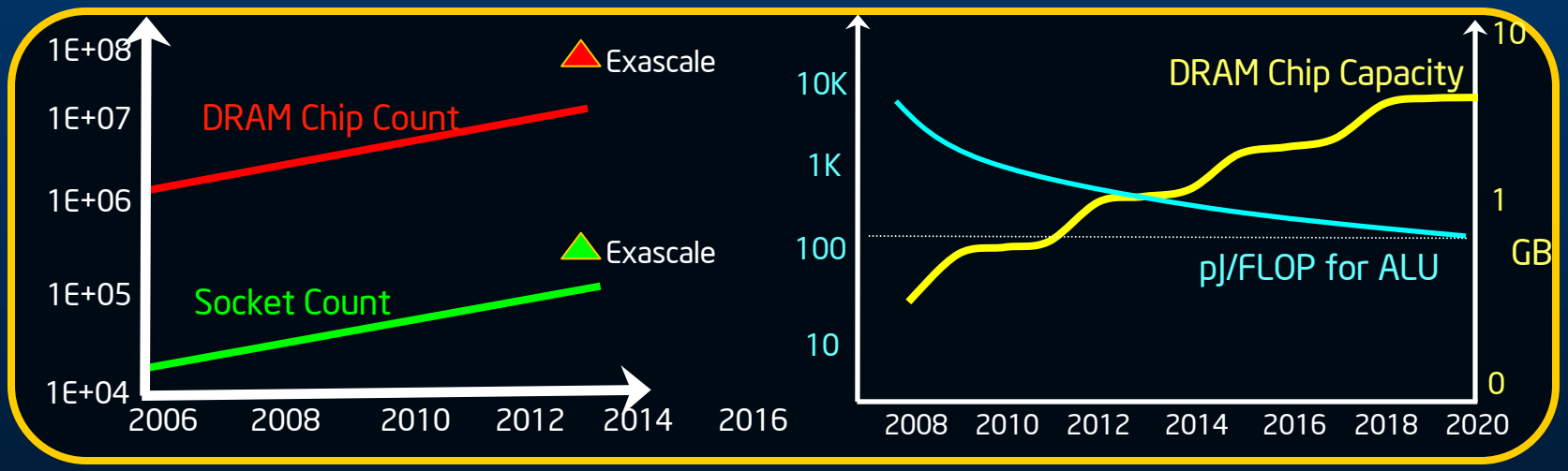
Source: Intel, http://www.top500.org

Myth 3:

There is no need to accelerate the development of exascale as it will evolve over time at its "normal" course and speed

Myth 3: "Business As Usual" will get us to Exascale

Let's test this myth by looking at these trends:



- Core count increase trend falls short of the exascale requirements
- Energy/Op for ALU operations alone is too high
- DRAM power is too high and Capacity/BW is too low

Reality: False

Current Natural Progression of Technology is Insufficient
- A Paradigm Shift in All Areas of Computing is Needed

Source: Intel, Modeling and Simulation at the Exascale for Energy and Environment (2007)

Business as usual will not work because

Challenges Need to Be Addressed to Reach Exascale

Energy Per Operation

Associated with Computation, Data Transport, Memory, and other overheads

Extreme Concurrency and Locality

Associated with programming billions of threads

Resiliency

Associated with growth in component count, lower voltages, security, etc

Memory/Storage Capacity, Bandwidth and Power

Associated with inability of current technology trend to meet requirements

Source: Exascale Computing Study: Technology Challenges in achieving Exascale Systems (2008)

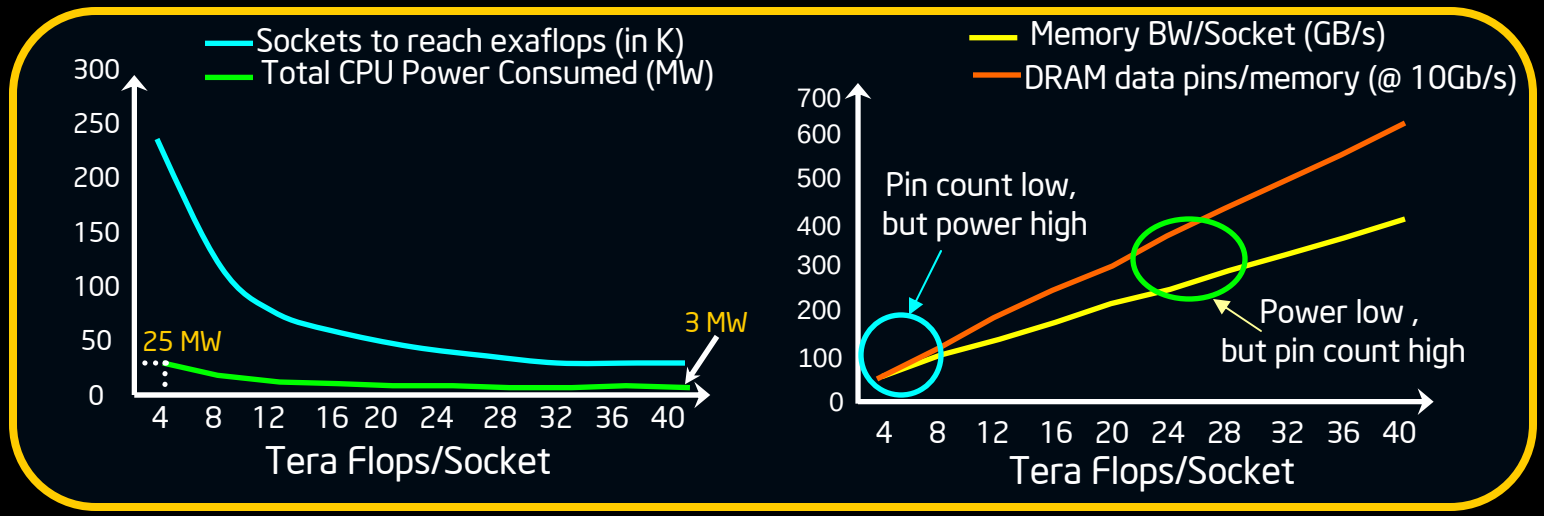
Myth 4:

It can be done with commodity technology

Myth 4: Commodity Technology is Enough

Let's test this myth with a simple example:

Assumptions: (i) Half of the Exaflops FP operations require memory access;
(ii) 20% cache miss rate
$\Rightarrow$ *Total Memory BW = 10PB/s*
(iii) Assume Total Memory Capacity = 10PB



Reality: False

High Pin Counts, Power and Memory Capacity/BW
rule out using today's commodity technologies

But, we can make Myth 4 a reality by …

Re-thinking System Level Memory Architecture

Evaluate using emerging memory technologies such as resistive memories in the mainstream

Consider new levels of memory hierarchy to bridge the memory performance gap

Pro-actively minimize data movement between levels of memory hierarchy

Develop innovative packaging and IO solutions that reduce communication

Myth 5:

The challenge of power consumption is too daunting

Power an Issue? Yes, but…

We have an understanding of where the issues are…
…And how to address them

Interconnect power reading 8bytes from memory @ 0.5EB/s = 100MW !
Transporting data, both on and off-chip, increasing interconnect power

Reading memory delivering 0.5EB/s = 300MW !
Memory Policies and DRAM Architecture increasing memory power

Power = Cdyn * Frequency * V^2 + Leakage
The increasing number of transistors increasing logic power

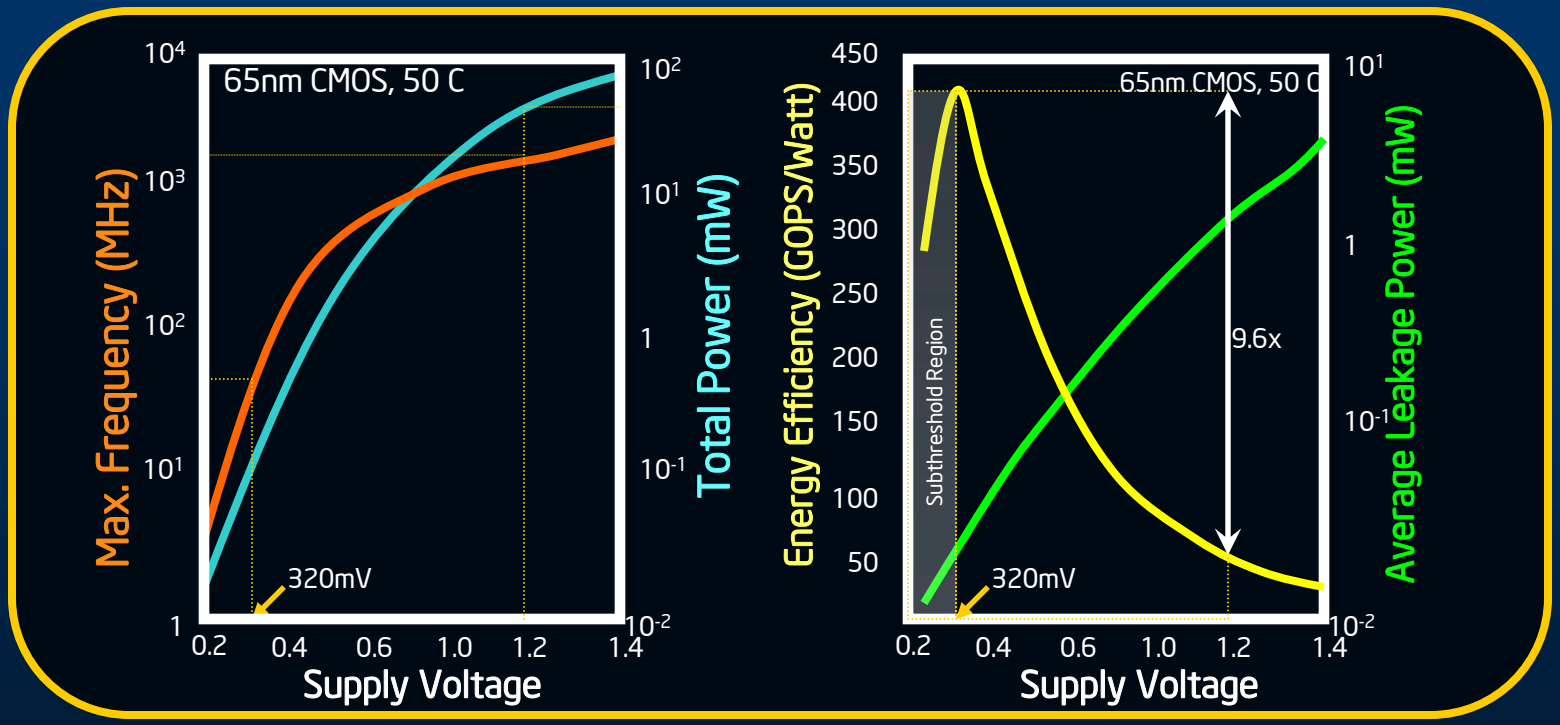
Wall to logic power conversion causing distribution power loss

Reality: Daunting Yes, But Not Overwhelming
Bad News: Many Areas Where Power is Wasted Today
Good News: Many Areas Where Power *Can* Be Saved!

Source: Exascale Computing Study: Technology Challenges in achieving Exascale Systems (2008)

One focus area to reduce power is

Extreme Voltage Scaling



- Significant power savings by scaling down supply voltage
- Re-optimize circuit parameters *for energy, <u>not</u> minimum latency*
- Using shallow pipelines in addition to voltage scaling further reduces power

Moore's Law is Alive and Well

Source: Intel

Myth 6:

It will take "miracles" to make exascale happen given the reality of technological invention

Myth 6: Need Miracles to Get to Exascale

Reality: False

Miracles? No

But Innovations Surely Are Needed !

Challenges do exist in Software Design, the first being…

Developing Applications for Exascale

Application Scaling to Exascale Systems

(i) Understanding how applications may grow and fit in new bigger systems
(ii) Understanding how existing applications adapt when executing on bigger systems

Few Examples to Learn From Today's Petascale Systems

More than 93% of current Top 500 Systems Peak Performance below 100TF/s

Both, Strong and Weak Scaling Have Limitations As System Size Grows

- **With strong scaling**, work/core decreases and communication overhead increases
  - ✓ *New techniques required to provide adequate BW and low latency to keep cores busy*
- **With weak scaling**, a 1000x increase in work requires a 180x increase in memory

No Single Silver Bullet to Scale Applications to Exascale

A second challenge that is holding back the programmer lies in…

Expressing and Managing Parallelism / Locality

The Mindset Needs to Change

- Programs organized according to sequential thinking and linear problem decomposition principles will be hard to parallelize
- *Not Just Software's Problem* - Hardware must adapt to morph itself to suit application requirements

Limitations With Programming Languages, Compilers, & Runtime

- Current PL* and Compilers are unable to handle algorithmic choice
- Two difficult choices available today with Runtime -
  - *Parallelism with no locality OR Locality with no parallelism*

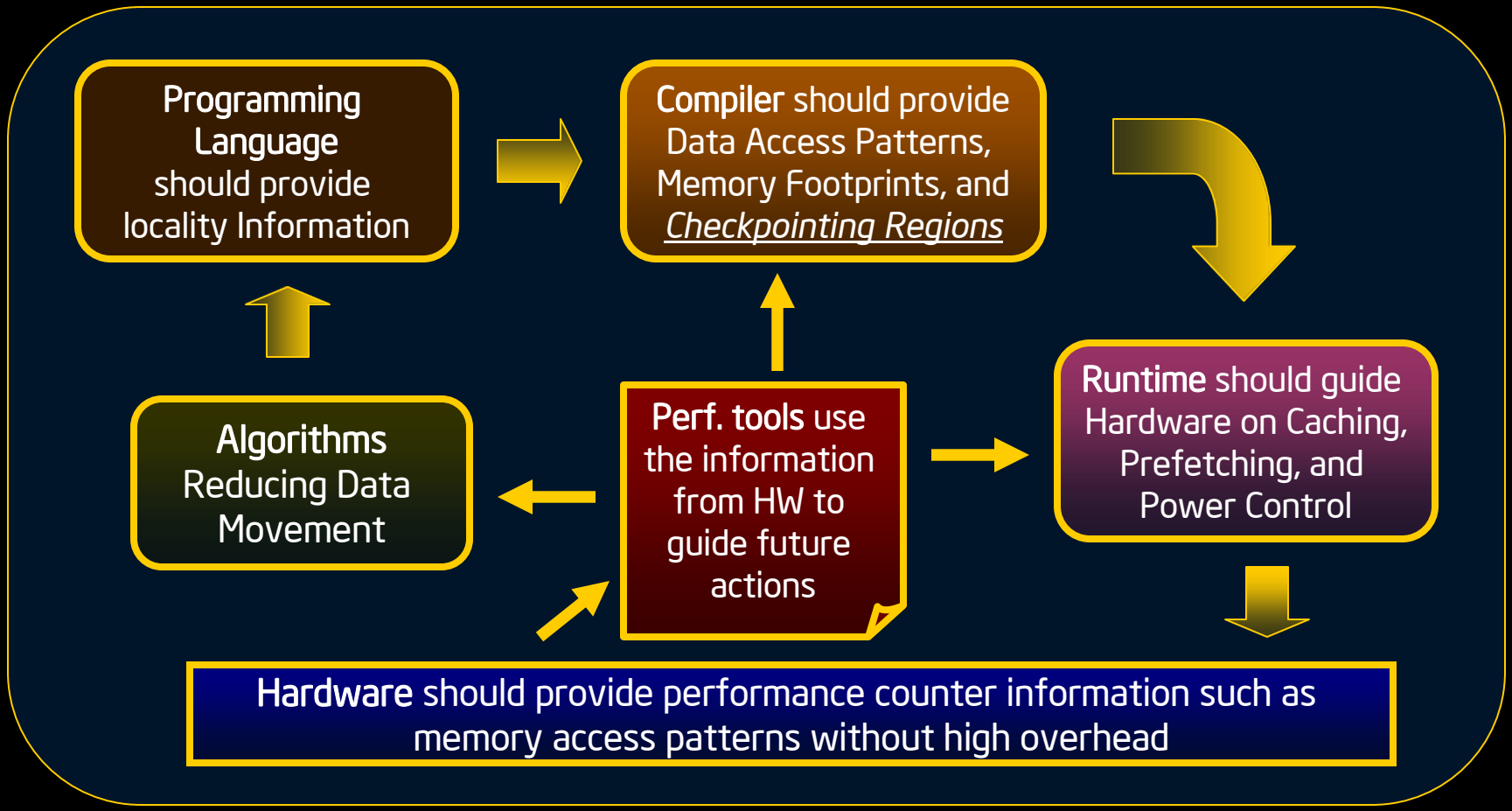
Better Exposure of Communication Requirements

- Fully Connected Networks will be unfeasibly expensive in Exascale
- PL must capture opportunities for re-use and locality
- Optimized mapping of processes onto HW elements is still a problem

*PL: Programming Languages

One way to overcome these challenges is to look into…

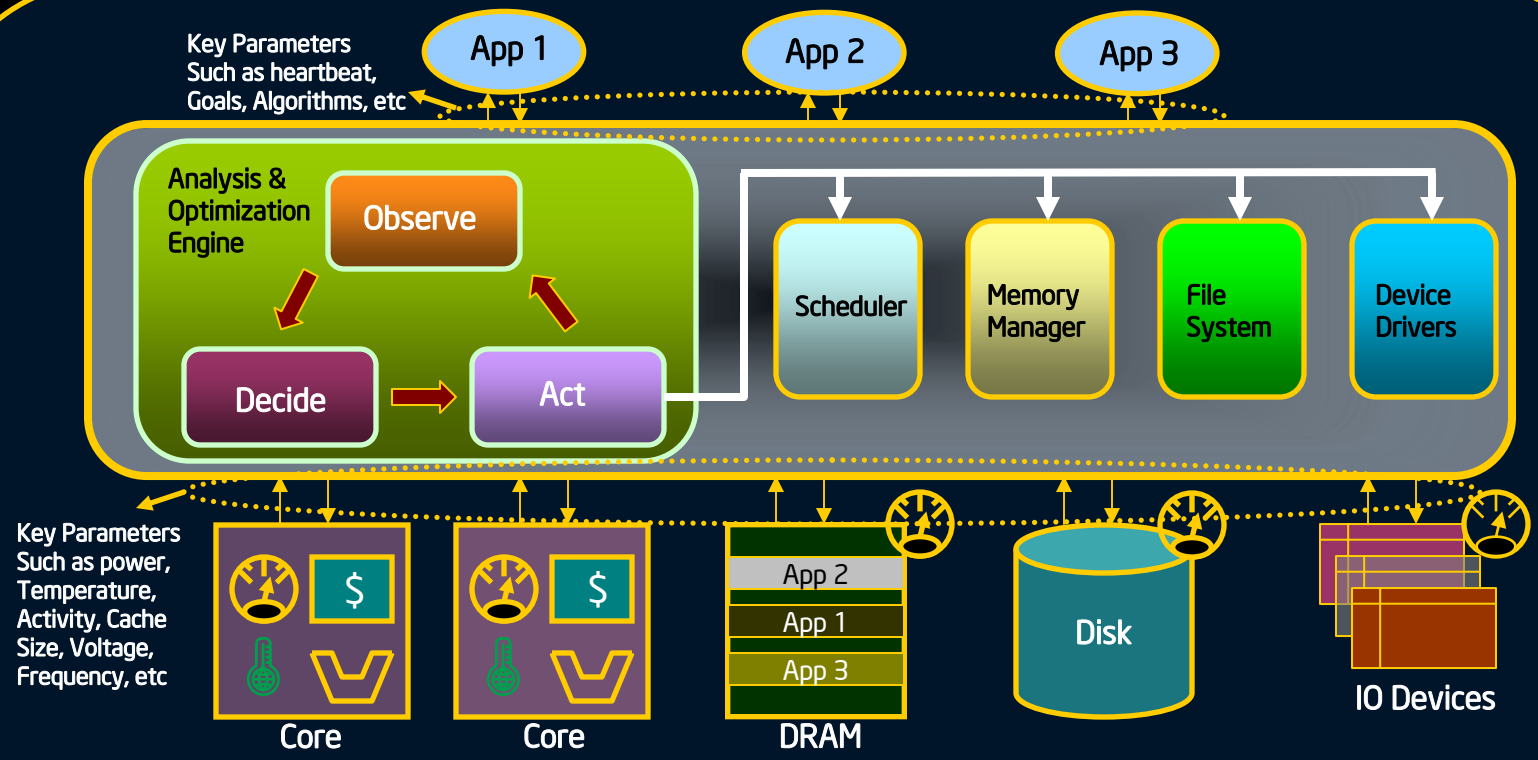
Hardware/Software Co-Design



Concurrency & Energy Efficiency is best achieved with a HW-SW Co-Design

And taking the previous idea a step further, we must strive to...

Create Self-Aware Systems



- **Introspective**: *Observes itself, reflects on its behavior and learns*
- **Goal Oriented**: *Client specifies the goal; System figures out how to get there*
- **Adaptive**: *Computes delta between goal and observed state; takes action to optimize*
- **Self Healing**: *Continues to function through faults and degrades gracefully*
- **Approximate**: *Does not expend more effort than necessary to meet goals*

Myth 7:

The technology will be too hard to use

Myth 7: *Exascale too hard to use? Not really…it will be*

Challenging to Develop, need to make it easier to Use

- 10 years ago, multi-core was perceived as hard to program
- Programming 10K threads or 1 billion threads…

  ….how can we minimize the impact to the programmer?
- As mentioned earlier, we must find a way to…

  …Extrapolate software to be scaled from today's limits

  …And develop a simple execution model for the programmer

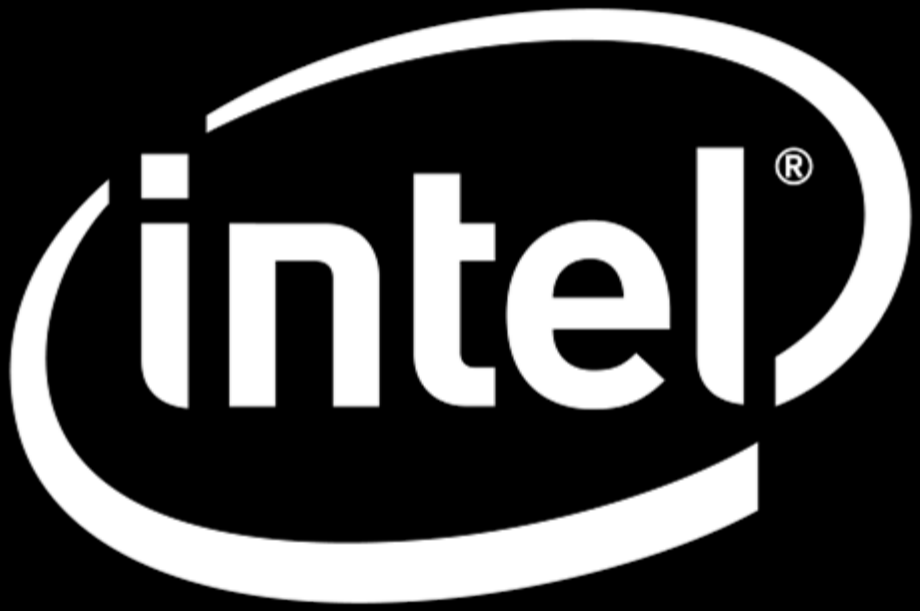
Reality: False

We can help with the parallel programming problem…

…Especially When Unified Under a Single ISA and Single Memory Model

In Conclusion

- Developing an Exascale System – at a reasonable cost - is challenging, but achievable with lots of engineering work
- Power is the key challenge; Data Movement is the key contributor
- Both, new technologies as well as re-architecting existing technologies – with a focus on power - is required
- Hardware and Software must work together to build an energy efficient and easy to use platform
- Focus on ease to/on the programmer – integrating an accelerator and CPU into a single silicon is only a *business* decision


intel