Toward Exascale Resilience

Part 1: Overview of challenges and trends

Mattan Erez
The University of Texas at Austin

July 2015

With support from

- DOE
- NSF

Bottom line:

System Complexity increasing

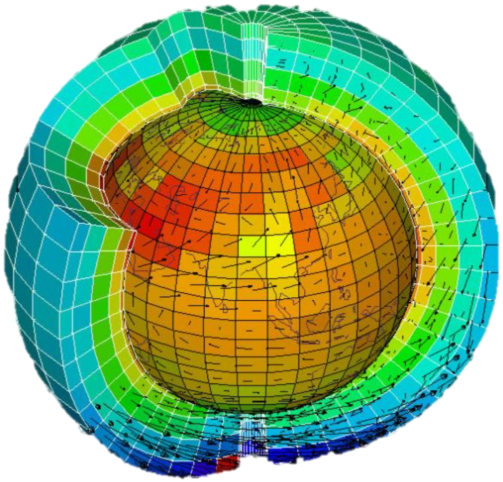
faster than

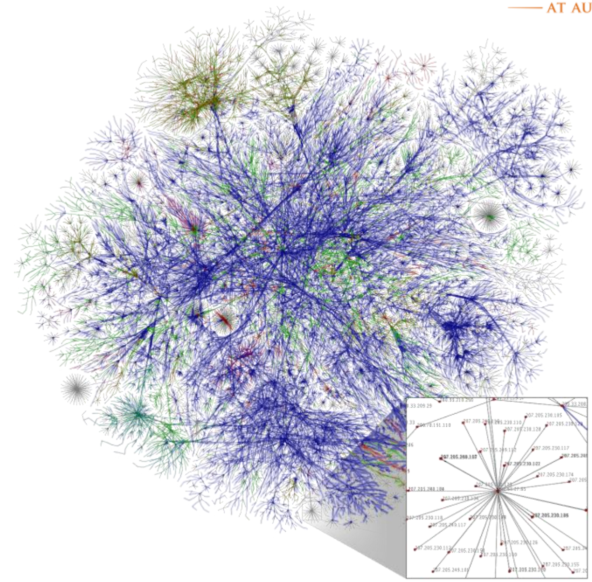
component reliability improves

No “right” answer

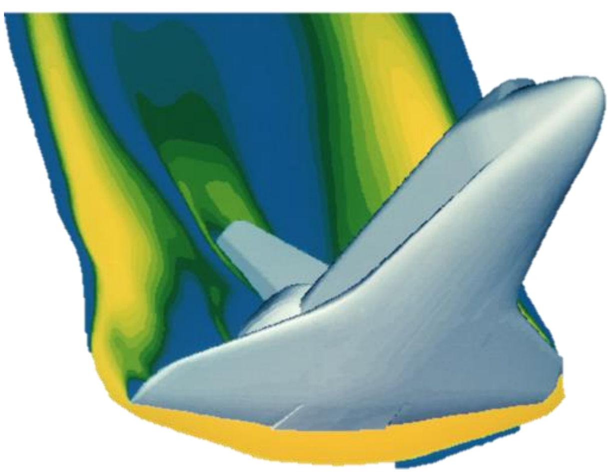
An **exascale** system is a high-performance system for solving **big problems**

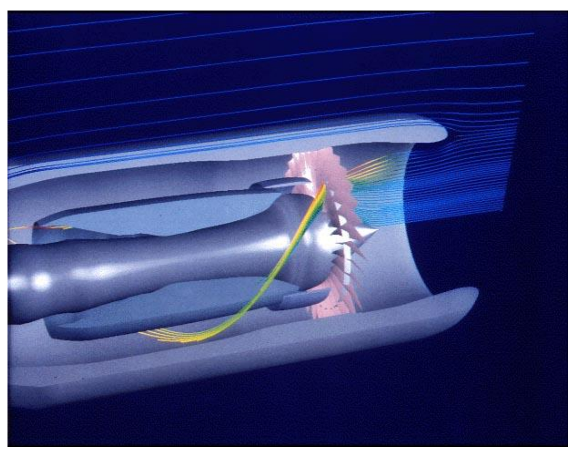
An **exascale** system is a high-performance system for solving **big cohesive problems**

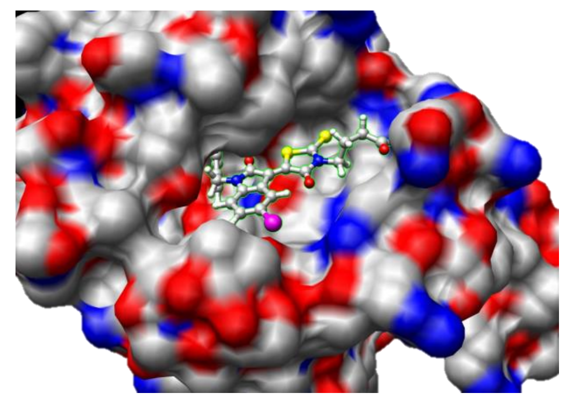




An **exascale** system is a high-performance system for solving **big cohesive problems**

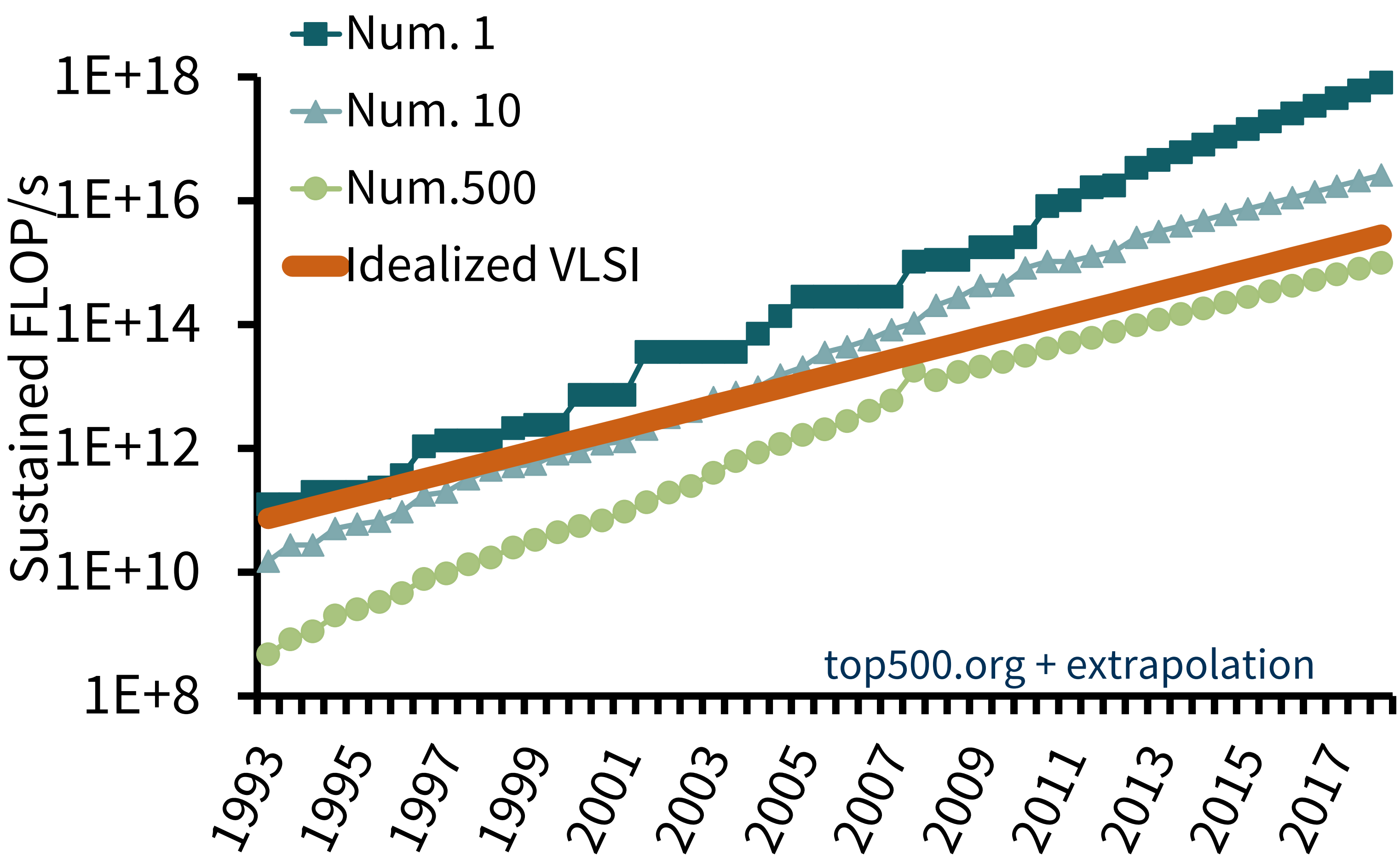






An **exascale** system is a high-performance system for solving **big** cohesive problems

Complexity and **components**



Performance outpaces even idealized VLSI

– HW component count increases exponentially

Logic

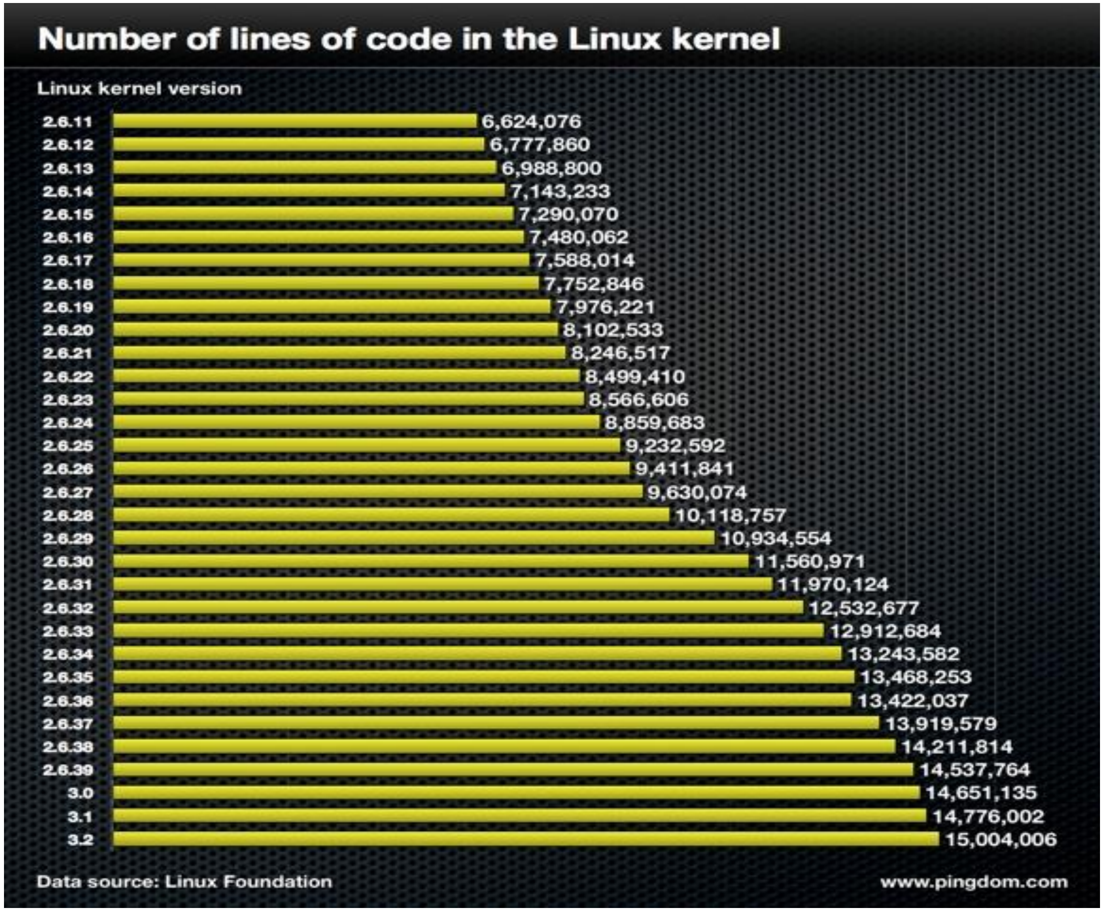
Memory

Links

Power

Cooling

From Linux Foundation



SW growing too

– HW controlled by complex and sophisticated SW

Management SW

Node OS

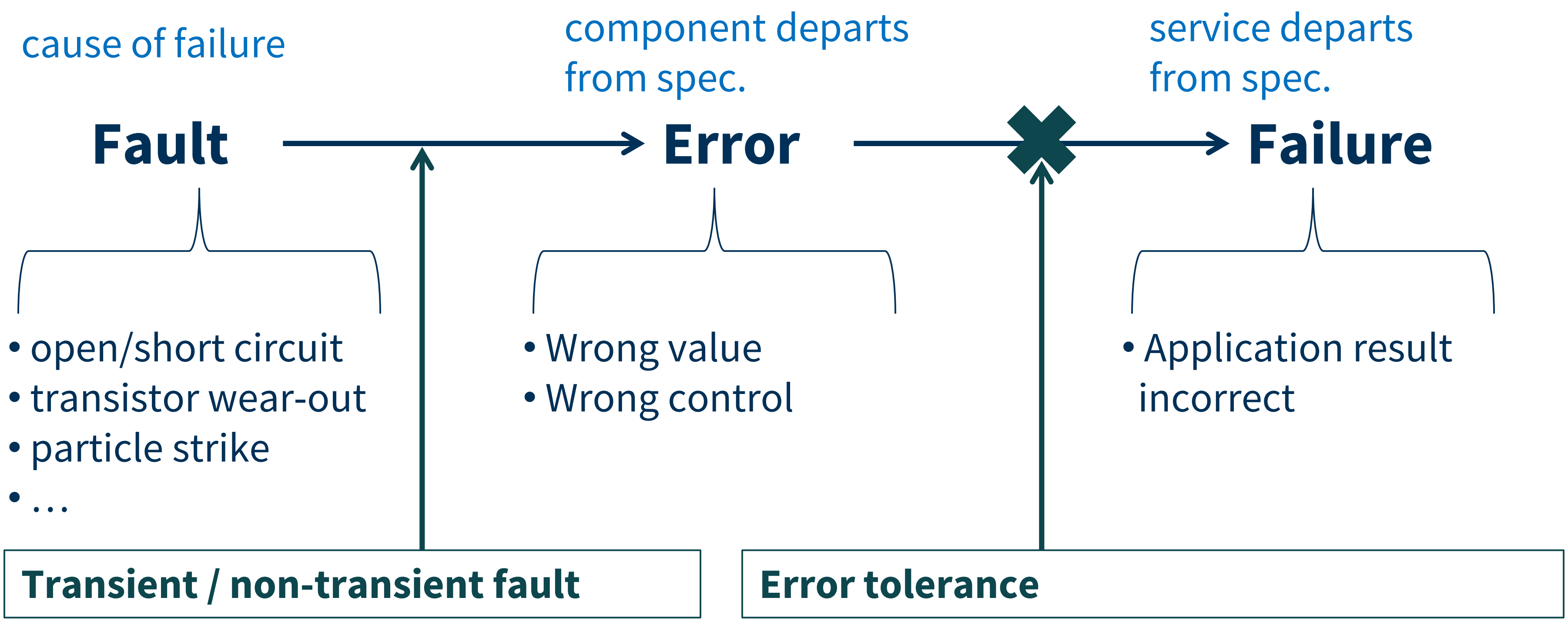
Application runtime

Application

What about component **reliability**?

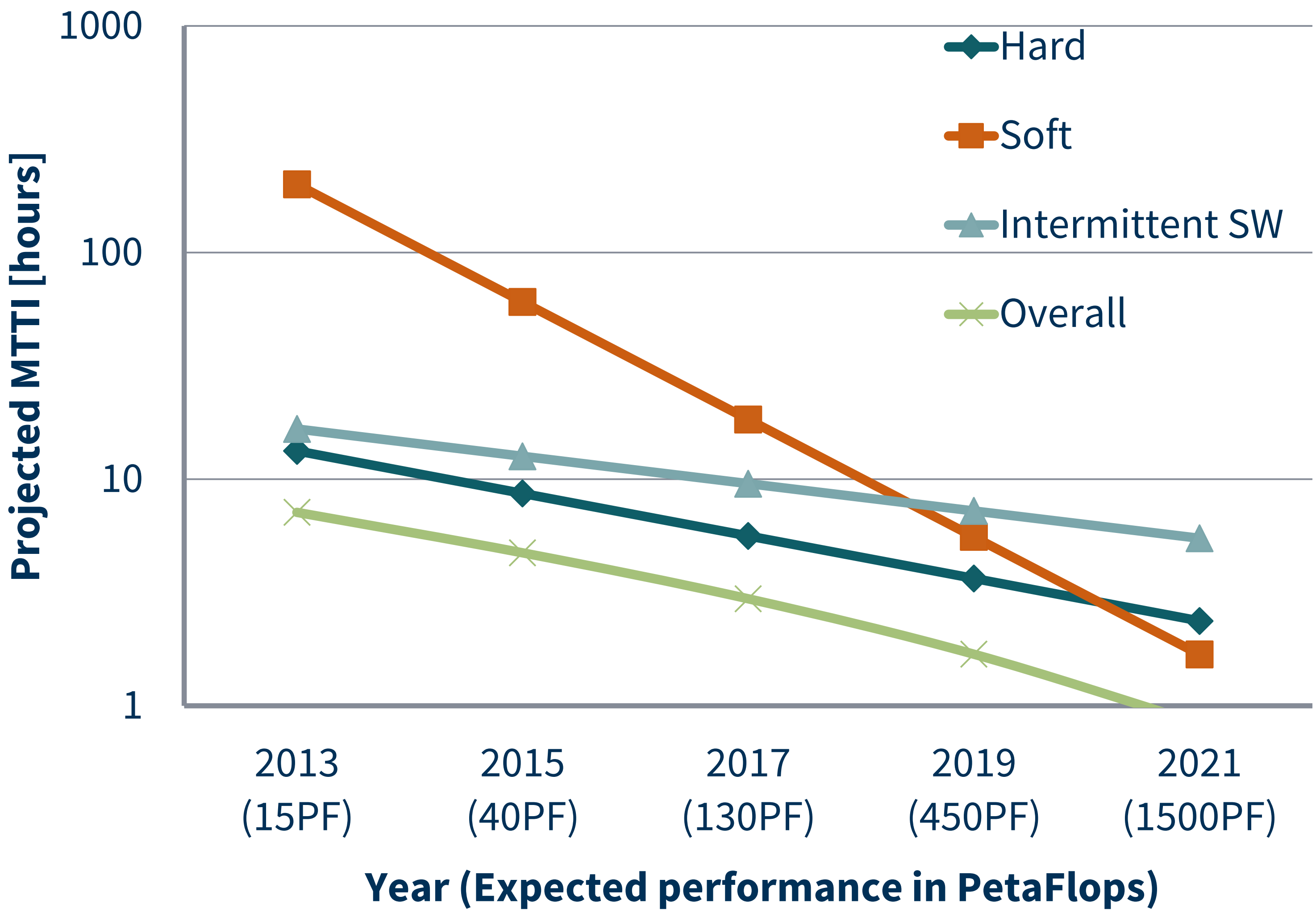
Reliable == runs without failing

- Failure means result is out of specification scope
- Reliability is hierarchical –
  reliable systems from unreliable components



What about component **reliability**?

- Significant uncertainty about predictions
- Largely because technology doesn’t stand still



An alarmist extrapolation

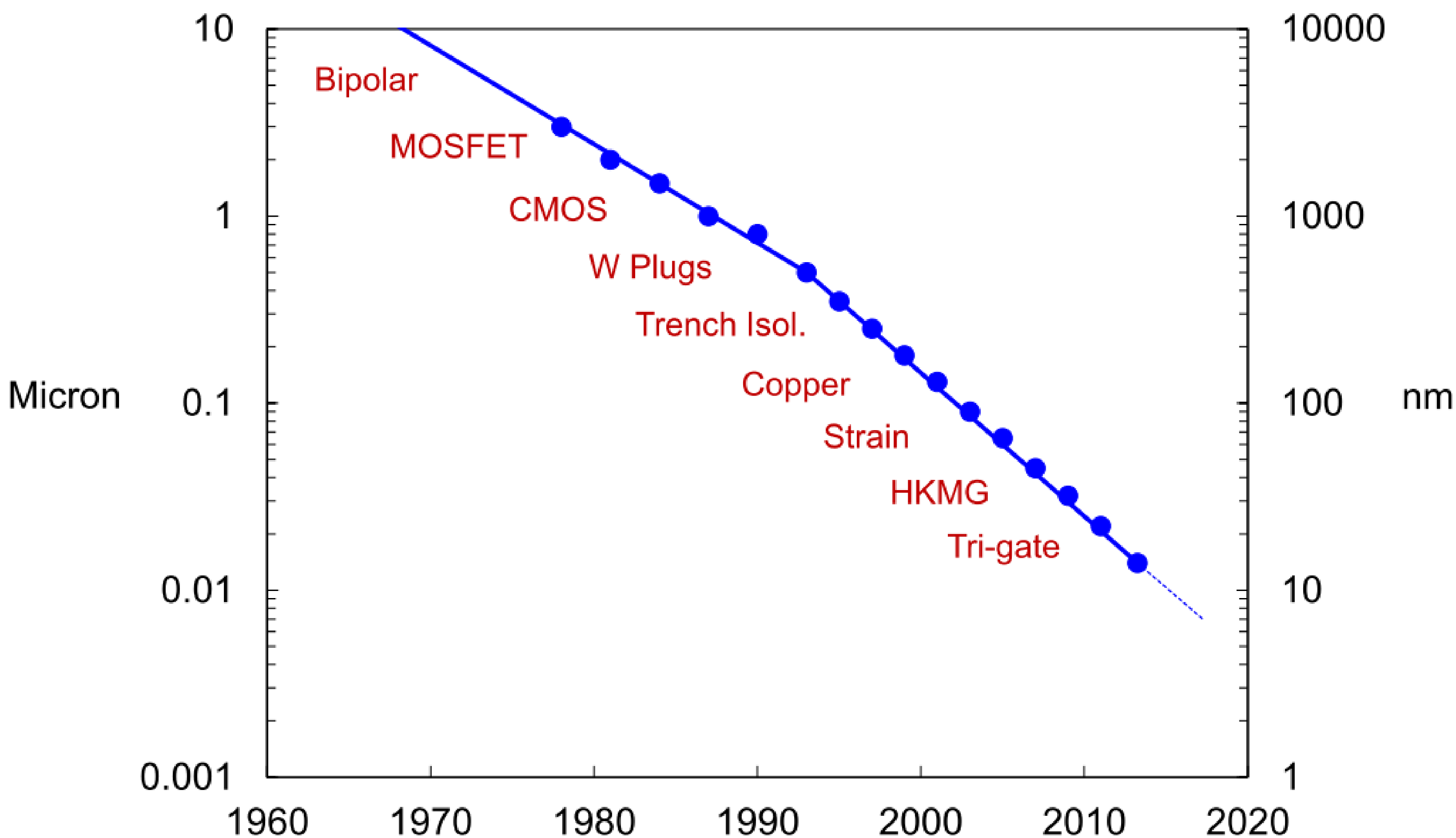
– Also with significant silent data corruption concern

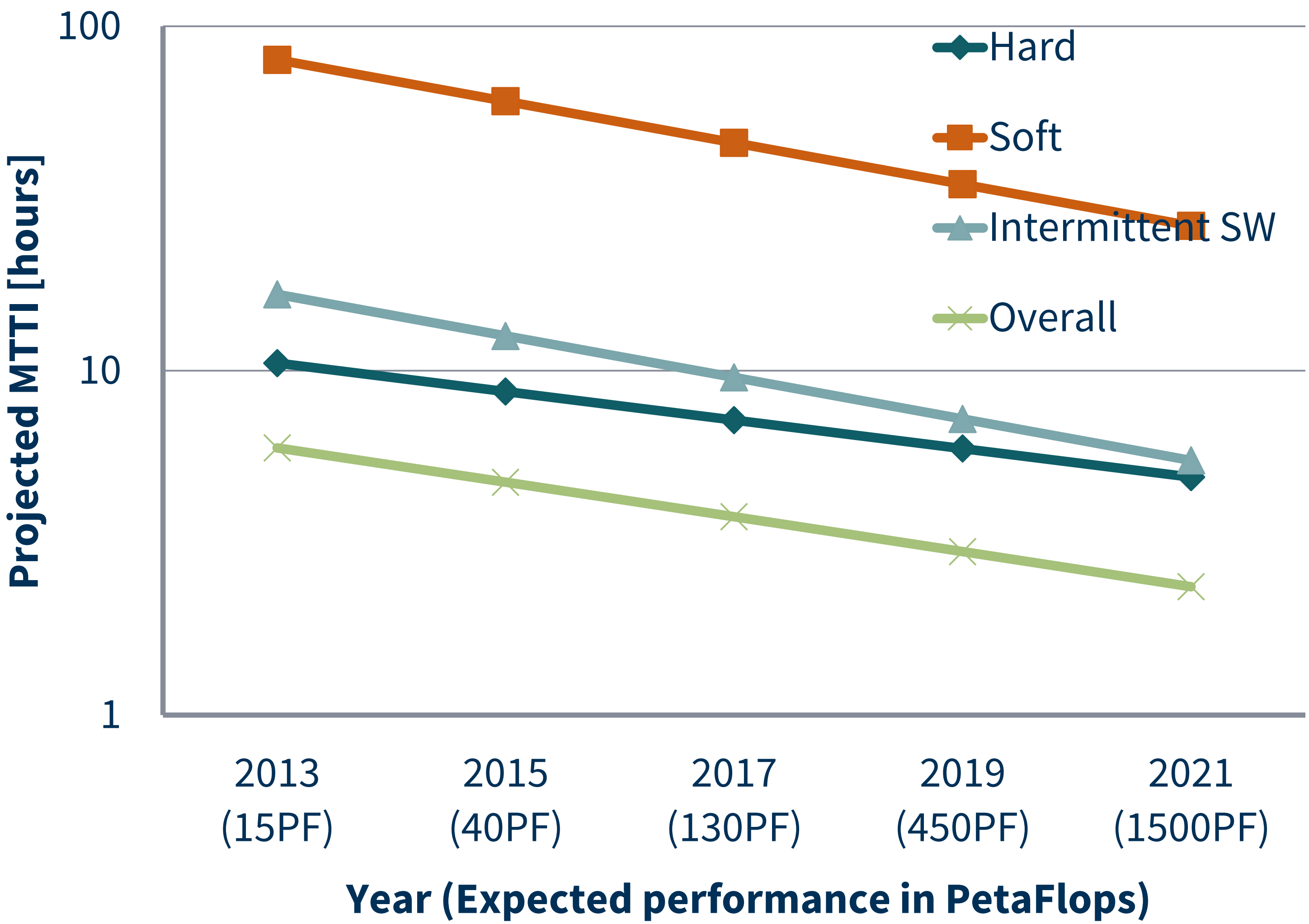
- Vendors work hard to keep reliability constant
  - Reliability doesn't come cheap though

VLSI exponentials result from discrete steps

50 Years of Moore's Law

From Mark Bohr, Intel ("Moore's Law: Yesterday, Today, and Tomorrow", May 26, 2015





An optimistic projection

– SDC concerns les clear

Can **all vendors** succeed?

- At what cost?

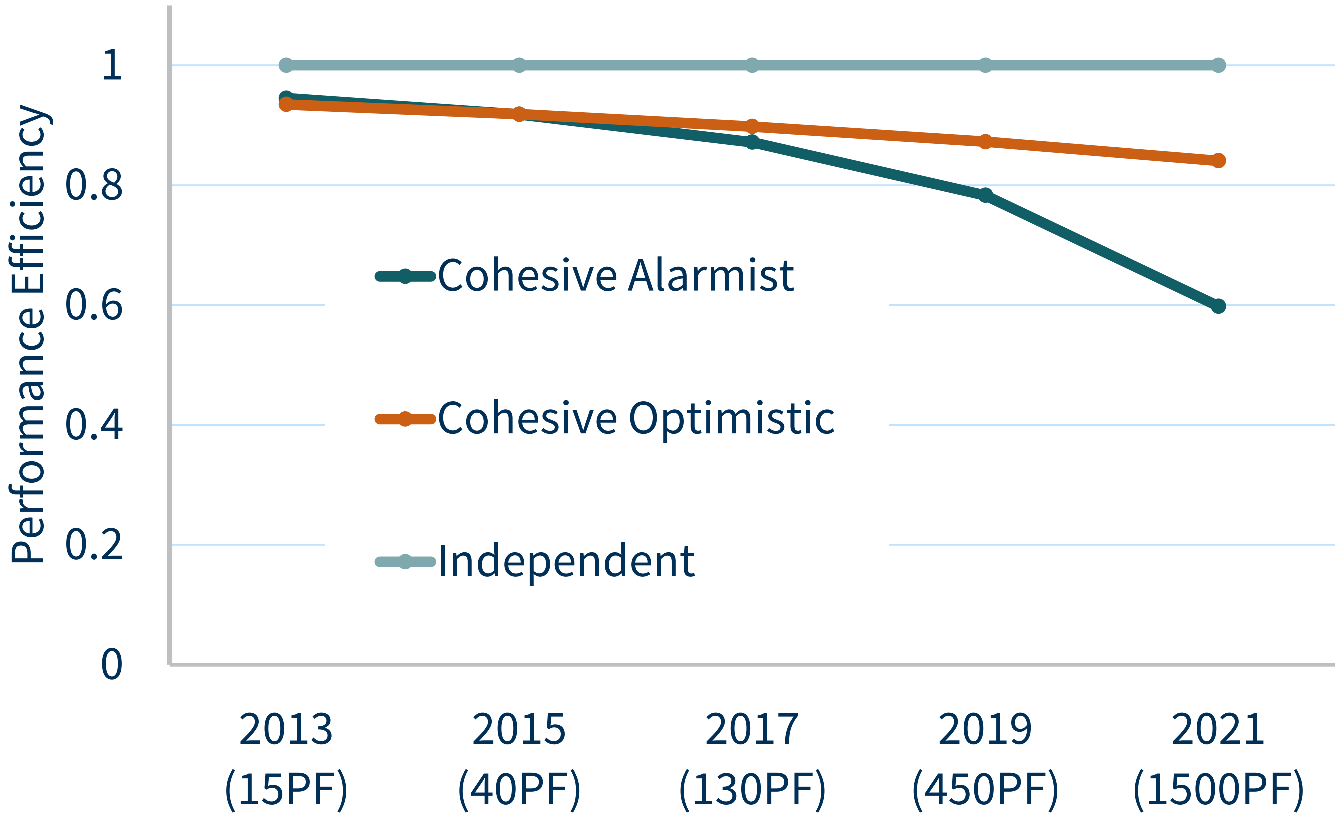
Do all **customers** care?

– “Commercial” vs. scientific

Clouds vs. supercomputers

Games vs. supercomputers

The cohesiveness **problem**



– Performance efficiency is expected performance relative to perfectly-reliable system

Reliability is in the eye of the beholder

- Level of paranoia (life threatening?)
- Monetary cost of failure (and legal implications)
- Tolerance to different results
- Can someone else be blamed?

No “right answer”

– Performance/efficiency and reliability tradeoffs

Short course goals

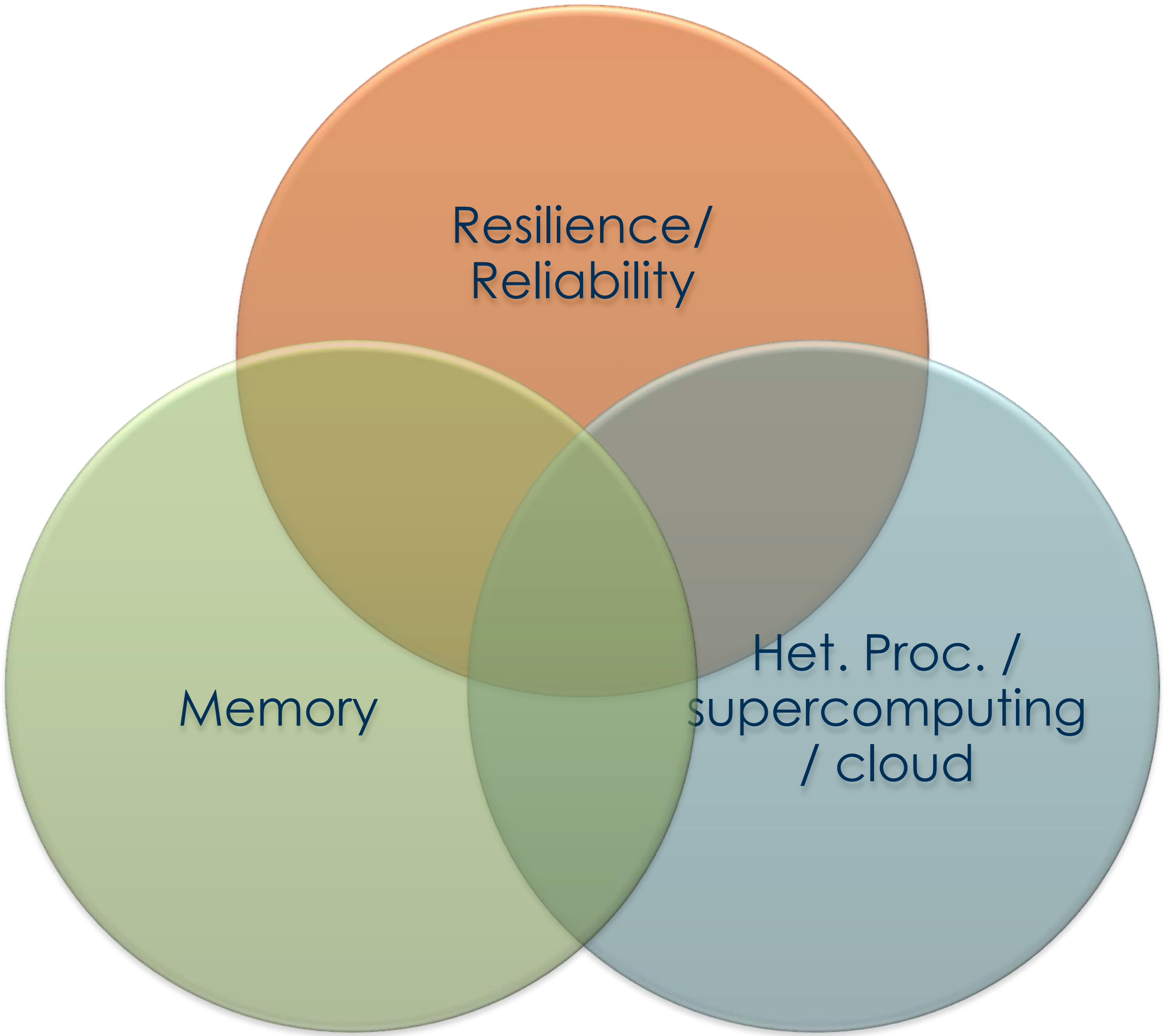
- For hardware, system, and application:
  - Understand the state of the practice
  - Understand the state of the art
  - Understand the sources of overhead
  - Understand how to improve resilience within a layer
- Understand cross-layer resilience
  - Better balance of overheads and needs

Syllabus (basics, then bottom-up)

- Overall system architecture
- Fundamentals of resilience
  - Terminology and actions
- Fault/error/failure modes and models
- Past, present, and near future approaches
  - HW techniques + checkpoint/restart
- Supercomputers aren't clouds
  - But what can we learn?
- Containment Domains and other future cross-layer approaches
  - With thoughts on approximate computing
- Reporting and forecasting

AN ASIDE ABOUT ME

Big problems and emerging systems



Arch-focused whole-system approach

Efficiency requirements require crossing layers

Algorithms are key

- Compute less, move less, store less

Proportional systems

- Minimize waste

Utilize and improve emerging technologies

Explore (and act) up and down

- Programming model to circuits

Preferably implement at micro-arch/system

Big problems and emerging platforms

Memory systems

- Capacity, bandwidth, efficiency – impossible to balance
- Adaptive and dynamic management helps
- New technologies to the rescue?
- Opportunities for in-memory computing

GPUs, supercomputers, clouds, and more

- Throughput oriented designs are a must
- Centralization trend is interesting

Reliability and resilience

- More, smaller devices – danger of poor reliability
- Trimmed margins – less room for error
- Hard constraints – efficiency is a must
- Scale exacerbates reliability and resilience concerns

Current and graduated PhD students

- Benjamin Cho
- Jinsuk Chung
- Seong-Lyong Gong
- Cagri Eryilmaz
- Dong Wan Kim
- Jungrae Kim
- Evgeni Krimer (NVIDIA)
- Min Kyu Jeong (Oracle Labs)
- Ikhwan Lee
- Kyushick Lee
- Mike Sullivan (NVIDIA Research)
- Minsoo Rhu (NVIDIA Research)
- Doe Hyun Yoon (Google)
- Song Zhang
- Tianhao Zheng
- Haishan Zhu

Current and graduate MS students

- Mehmet Basoglu (Broadcom)
- Esha Choukse
- Nick Kelly
- Mahnaz Sadoughi (Apple)

Collaborators at

- AMD
- Cray
- Intel
- NVIDIA
- Samsung
- LBNL, PNNL
- Stanford

Lots of great insightful feedback

- Grew up mostly in Israel
- BSc EE and BS Physics at Technion
- Part-time researcher at Intel Haifa
- PhD at Stanford (w/ Bill Dally)
- At UT Austin since 2007
  - Going well (great students and collaborators)
  - ~7 architects + ~5 systems
  - Very strong collaborations with others
  - TACC supercomputing center