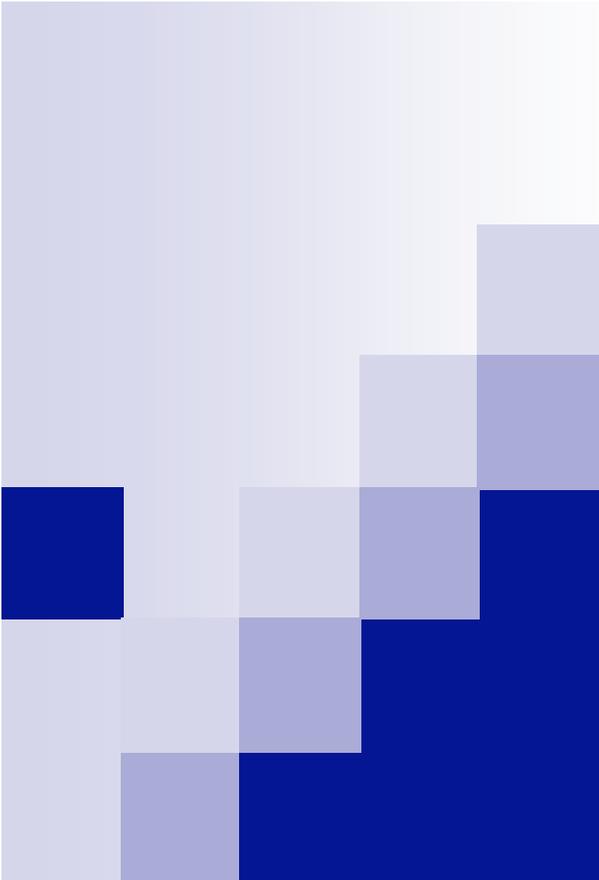


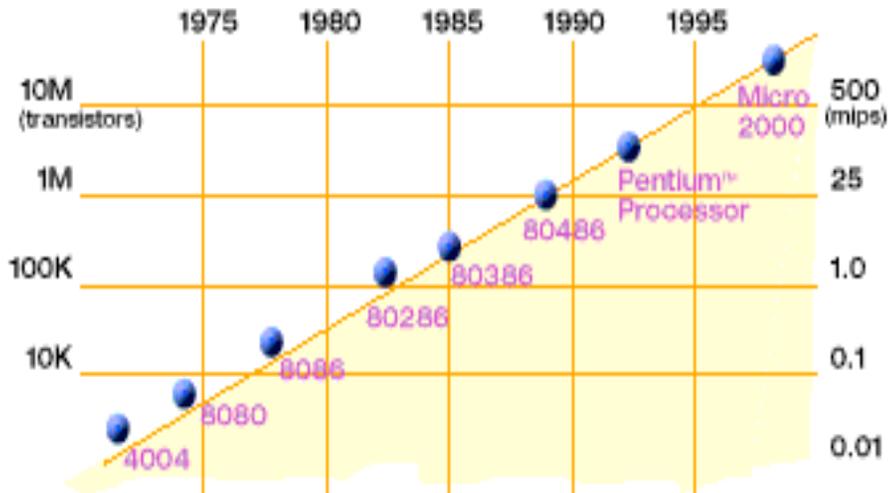
Multicore: Fallout of a Hardware Revolution

Katherine Yelick, NERSC Division Director
and a cast of thousands from
U.C. Berkeley and Lawrence Berkeley National Laboratory



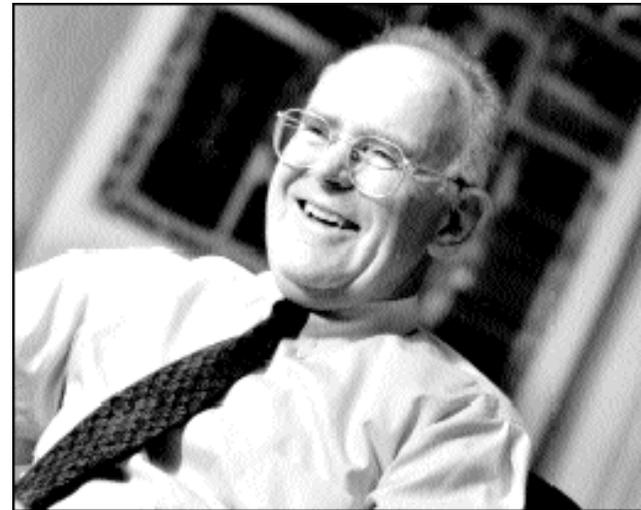
What has changed (and
what has not)

Old & New: Moore's Law is Alive and Well



2X transistors/Chip Every 1.5 years
Called "**Moore's Law**"

Microprocessors have become smaller, denser, and more powerful.

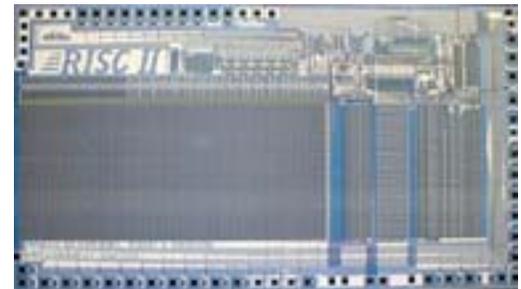
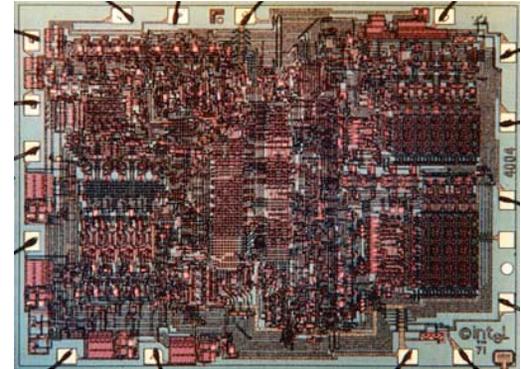


Gordon Moore (co-founder of Intel) predicted in 1965 that the transistor density of semiconductor chips would double roughly every 18 months.

Slide source: Jack Dongarra

Sea Change in Chip Design

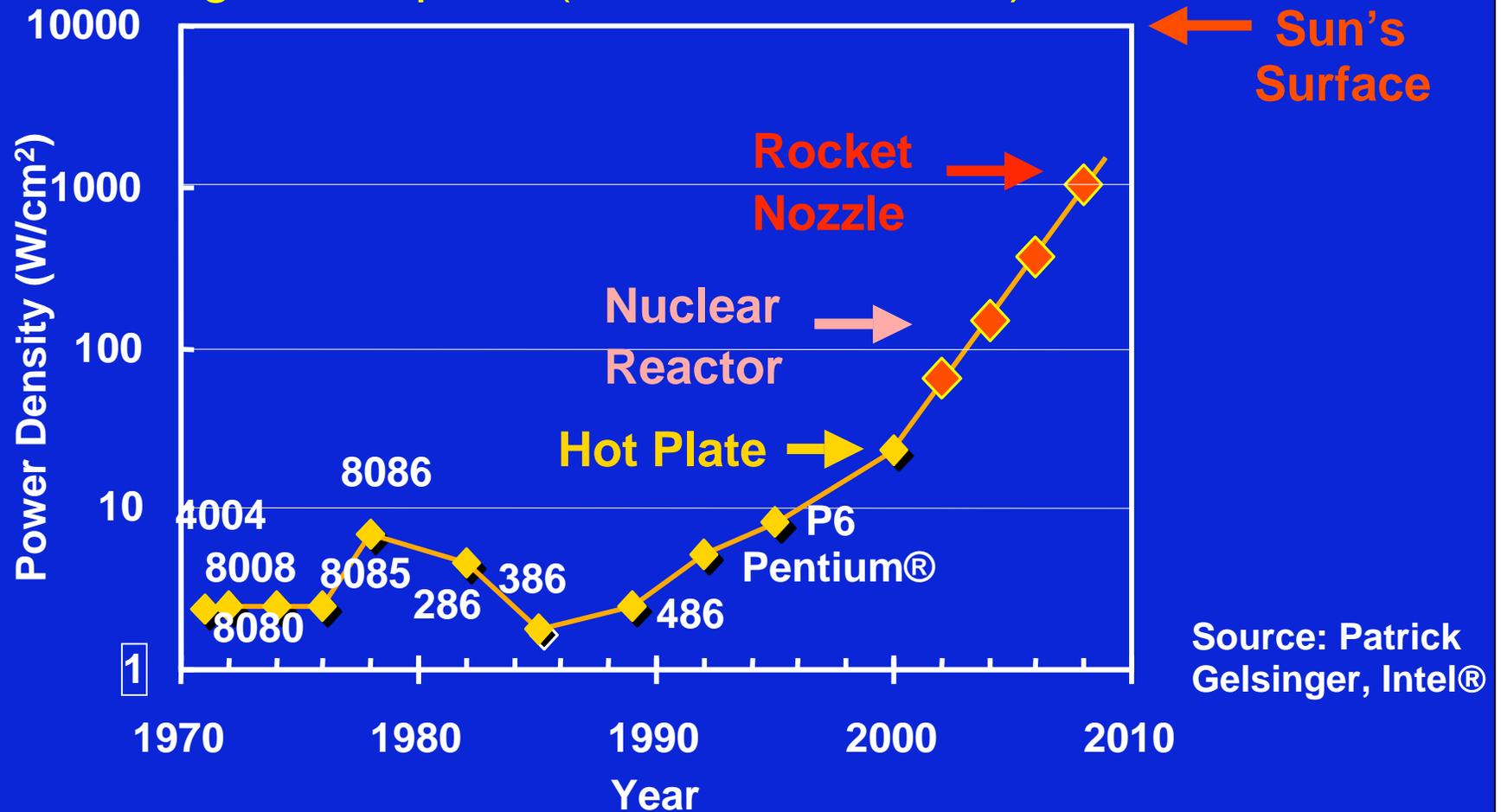
- Intel 4004 (1971): 4-bit processor, 2312 transistors, 0.4 MHz, 10 micron PMOS, 11 mm² chip
 - RISC II (1983): 32-bit, 5 stage pipeline, 40,760 transistors, 3 MHz, 3 micron NMOS, 60 mm² chip
 - 125 mm² chip, 0.065 micron CMOS
= 2312 RISC II+FPU+Icache+Dcache
 - RISC II shrinks to ≈ 0.02 mm² at 65 nm
 - Caches via DRAM or 1 transistor SRAM or 3D chip stacking
 - Proximity Communication via capacitive coupling at > 1 TB/s ?
(Ivan Sutherland @ Sun / Berkeley)
- **Processor is the new transistor!**



New: Power Wall

Can put more transistors on a chip than can afford to turn on

Scaling clock speed (business as usual) will not work

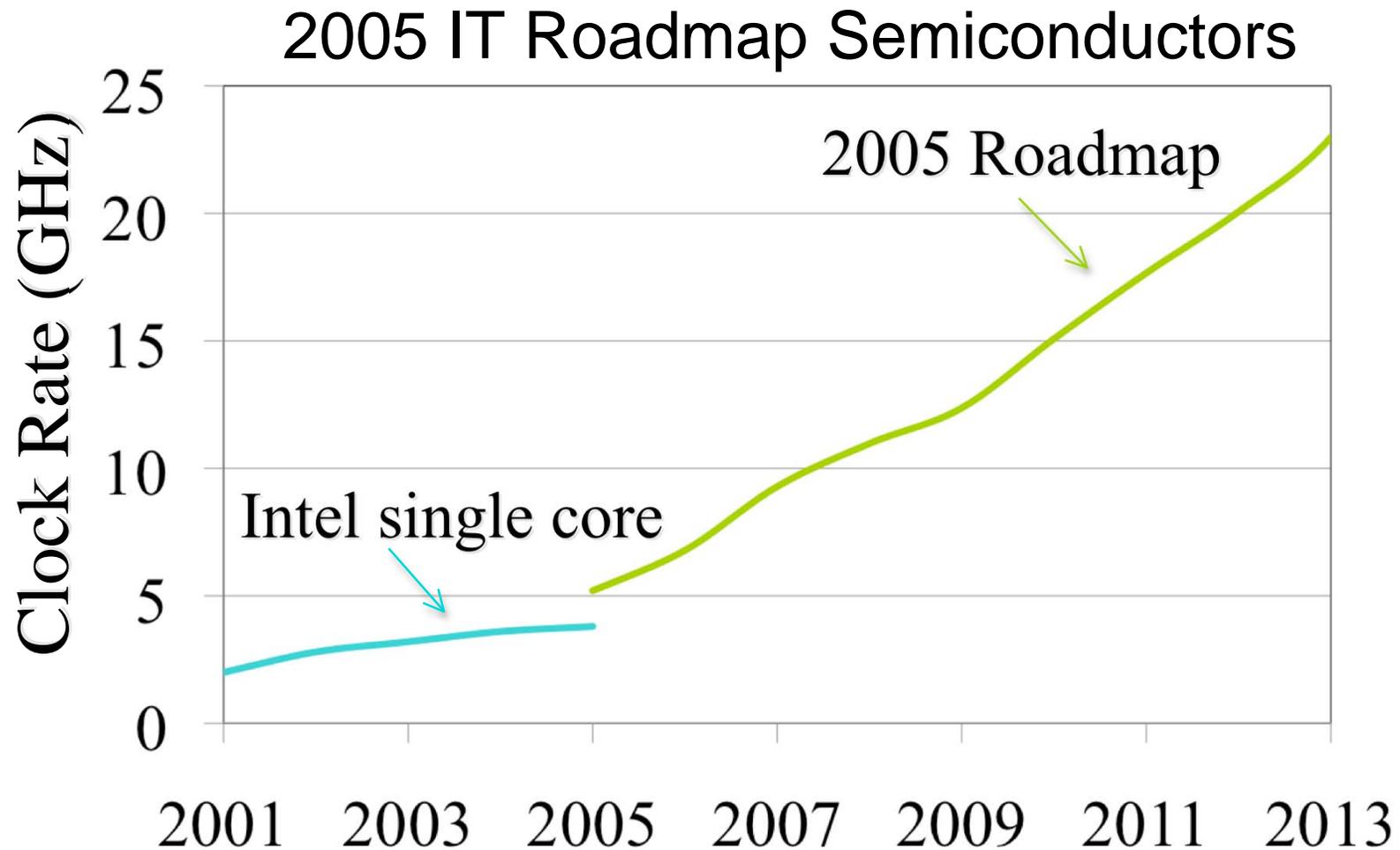




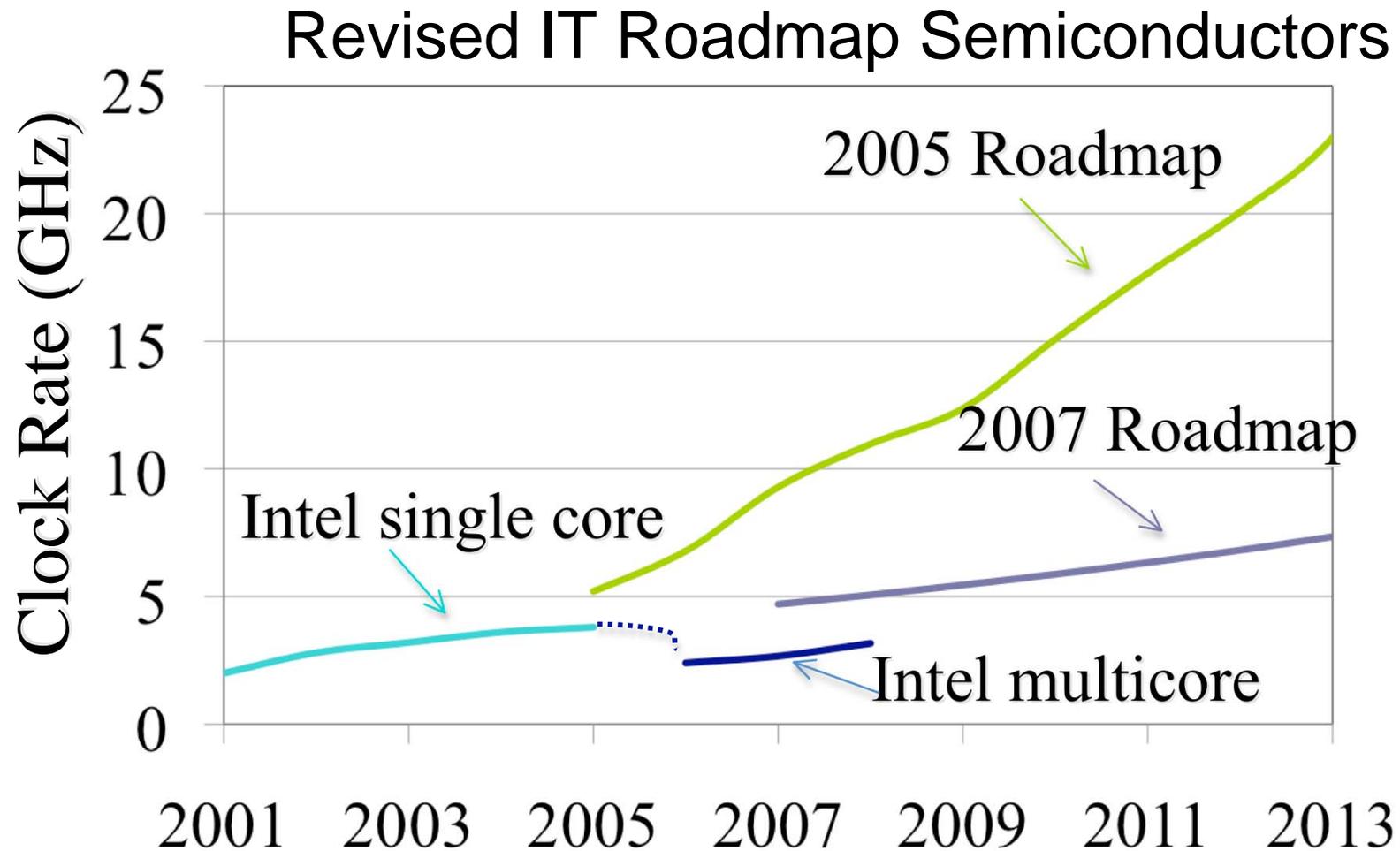
Why Parallelism Lowers Power

- Highly concurrent systems are more power efficient
 - *Dynamic power is proportional to V^2fC*
 - *Increasing frequency (f) also increases supply voltage (V): more than linear effect*
 - *Increasing cores increases capacitance (C) but has only a linear effect*
- Hidden concurrency burns power
 - Speculation, dynamic dependence checking, etc.
 - Push parallelism discovery to software (compilers and application programmers) to save power
- Challenge: *Can you double the concurrency in your algorithms every 2 years?*

Old: Clock speed double every 2 years



New: Cores/chip will double every 2 years

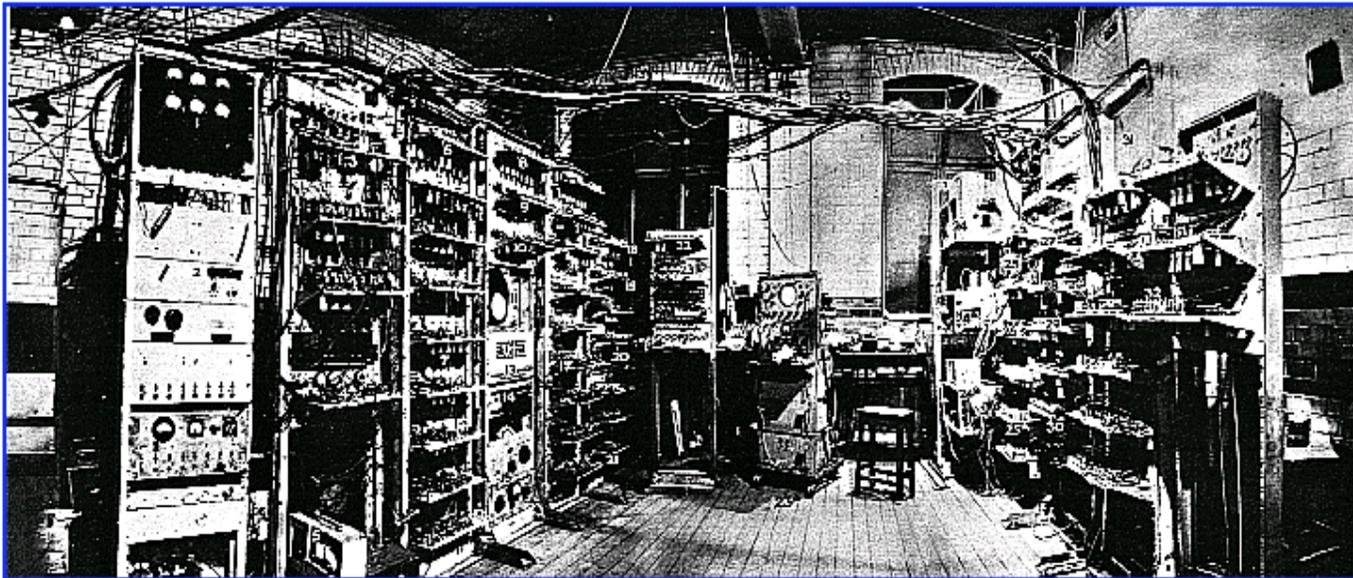


Old: Parallelism only for High End Computing

Address  <http://www.paralogos.com/DeadSuper/index.html>

 Go  Links >>

The Dead Supercomputer Society



The Passing of a Golden Age?

From the construction of the first programmed computers until the mid 1990s, there was always room in the computer industry for someone with a clever, if sometimes challenging, idea on how to make a more powerful machine. Computing became strategic during the Second World War, and remained so during the Cold War that followed. High-performance computing is essential to any modern nuclear weapons program, and a computer technology "race" was a logical corollary to the arms race. While powerful computers are of great value to a number



New: Parallelism by Necessity

“This shift toward increasing parallelism is not a triumphant stride forward based on breakthroughs in novel software and architectures for parallelism; instead, this **plunge into parallelism is actually a retreat from even greater challenges that thwart efficient silicon implementation of traditional uniprocessor architectures.**”

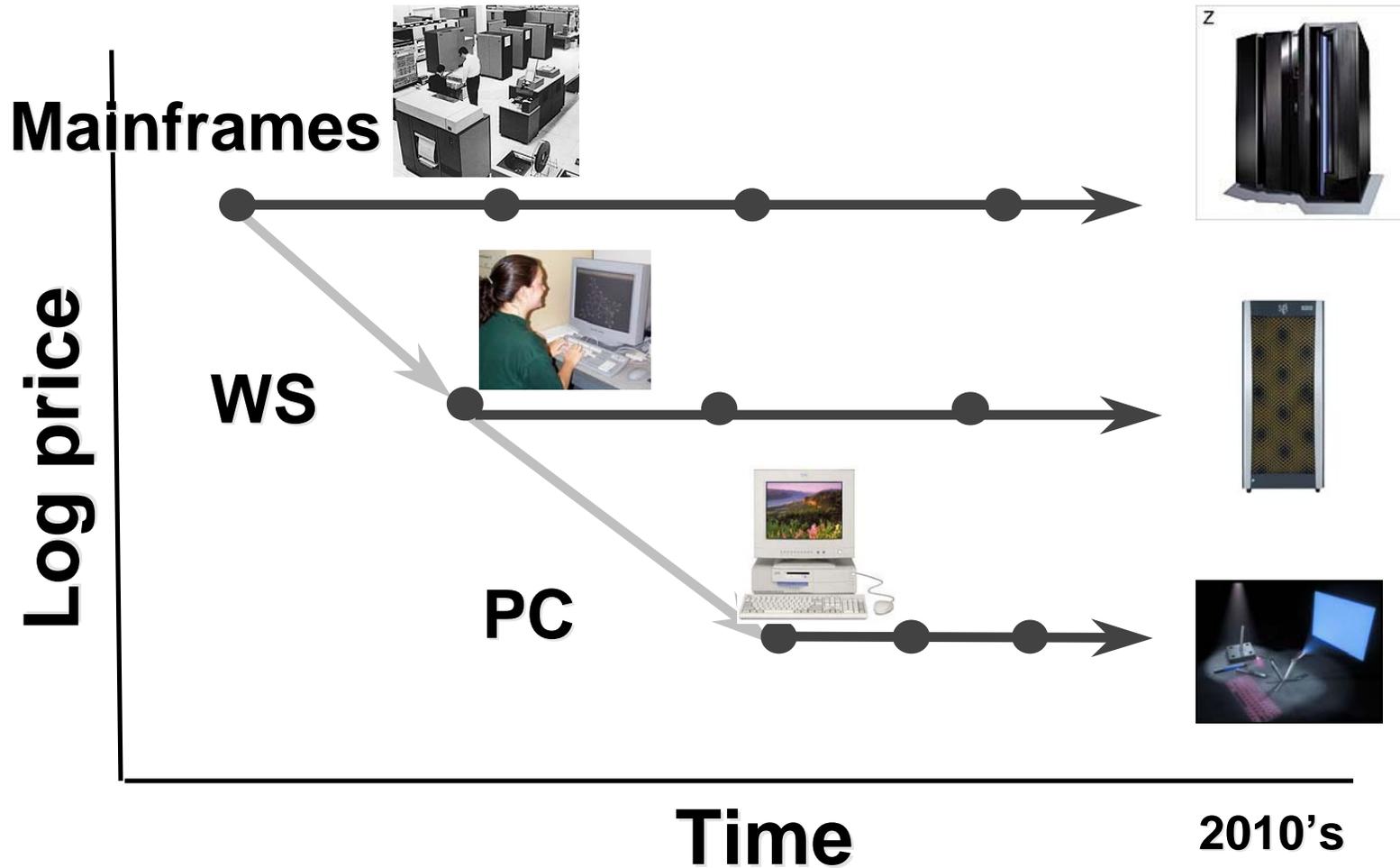
Kurt Keutzer, Berkeley View, December 2006

- HW/SW Industry bet its future that breakthroughs will appear before its too late

Bell's Evolution Of Computer Classes

Technology enables two paths:

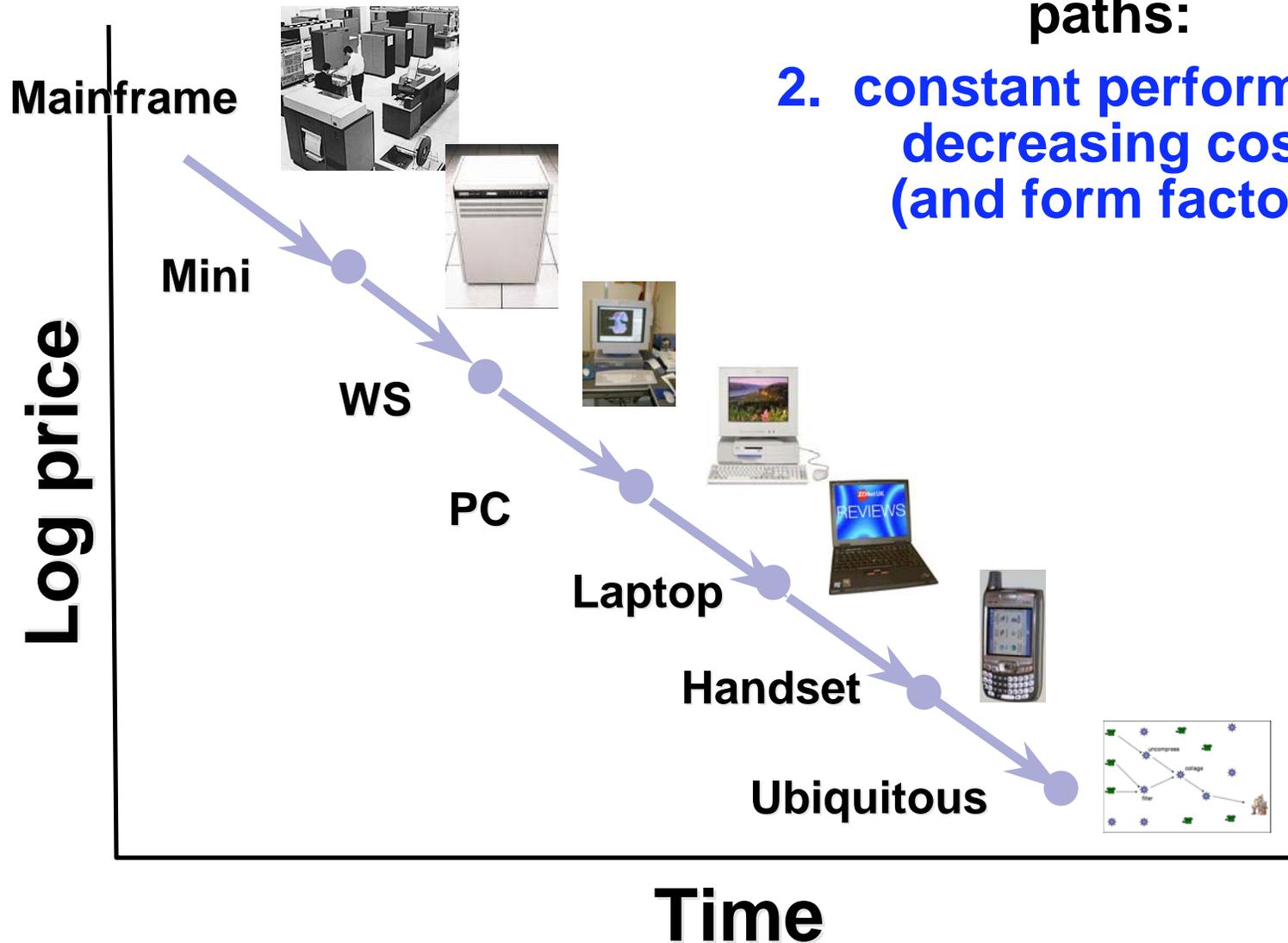
1. increasing performance, same cost (and form factor)



Bell's Evolution Of Computer Classes

Technology enables two paths:

2. constant performance, decreasing cost (and form factor)



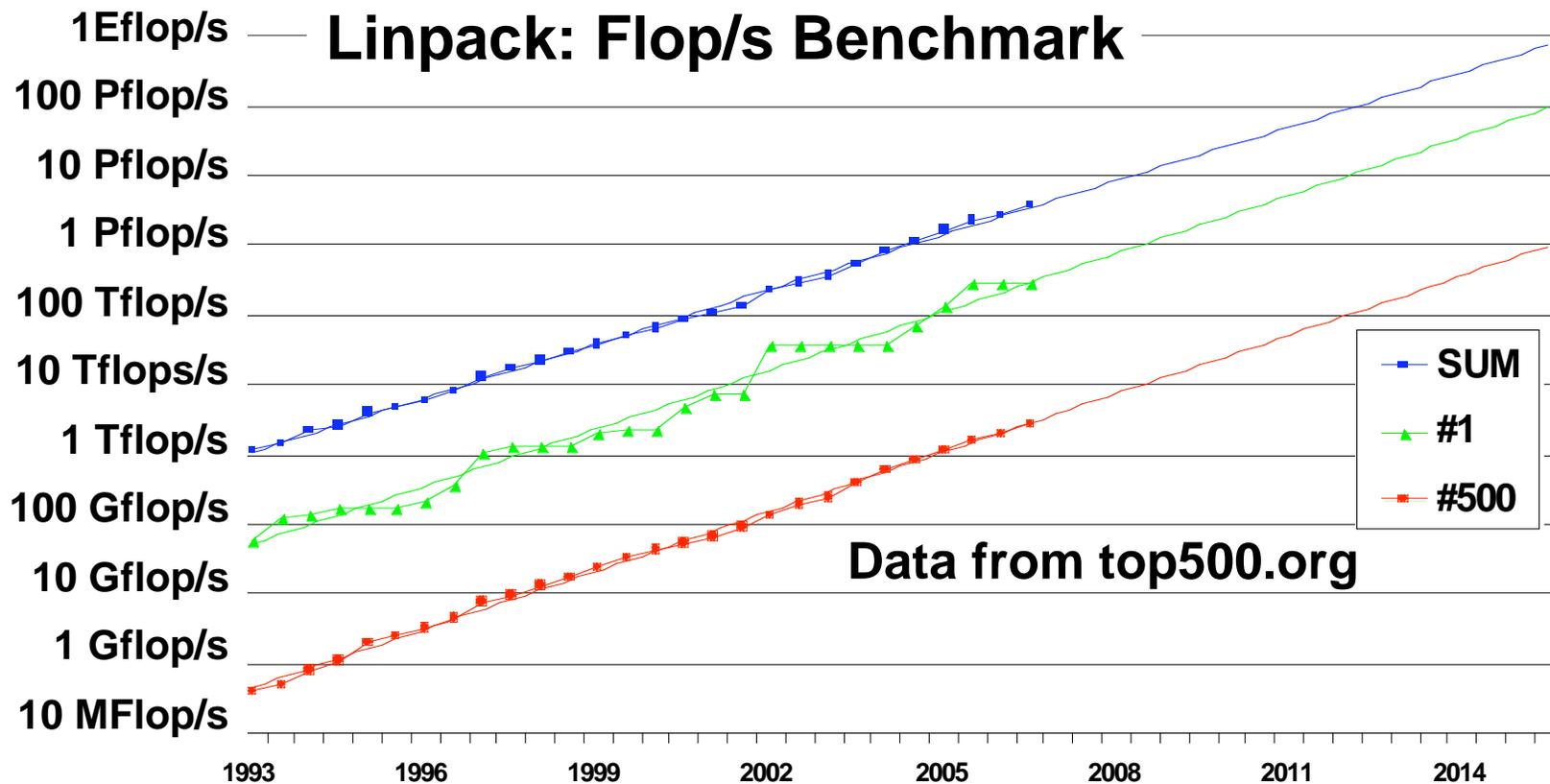
Re-inventing Client/Server

- “The Datacenter is the Computer”
 - Building sized computers: Google, MS ..
- “The Laptop/Handheld is the Computer”
 - '08: Dell # laptops > # desktops?
 - 1B Cell phones/yr, increasing in function
 - Will desktops disappear? Laptops?
- Laptop/Handheld as future client,
Datacenter or “cloud” as future server



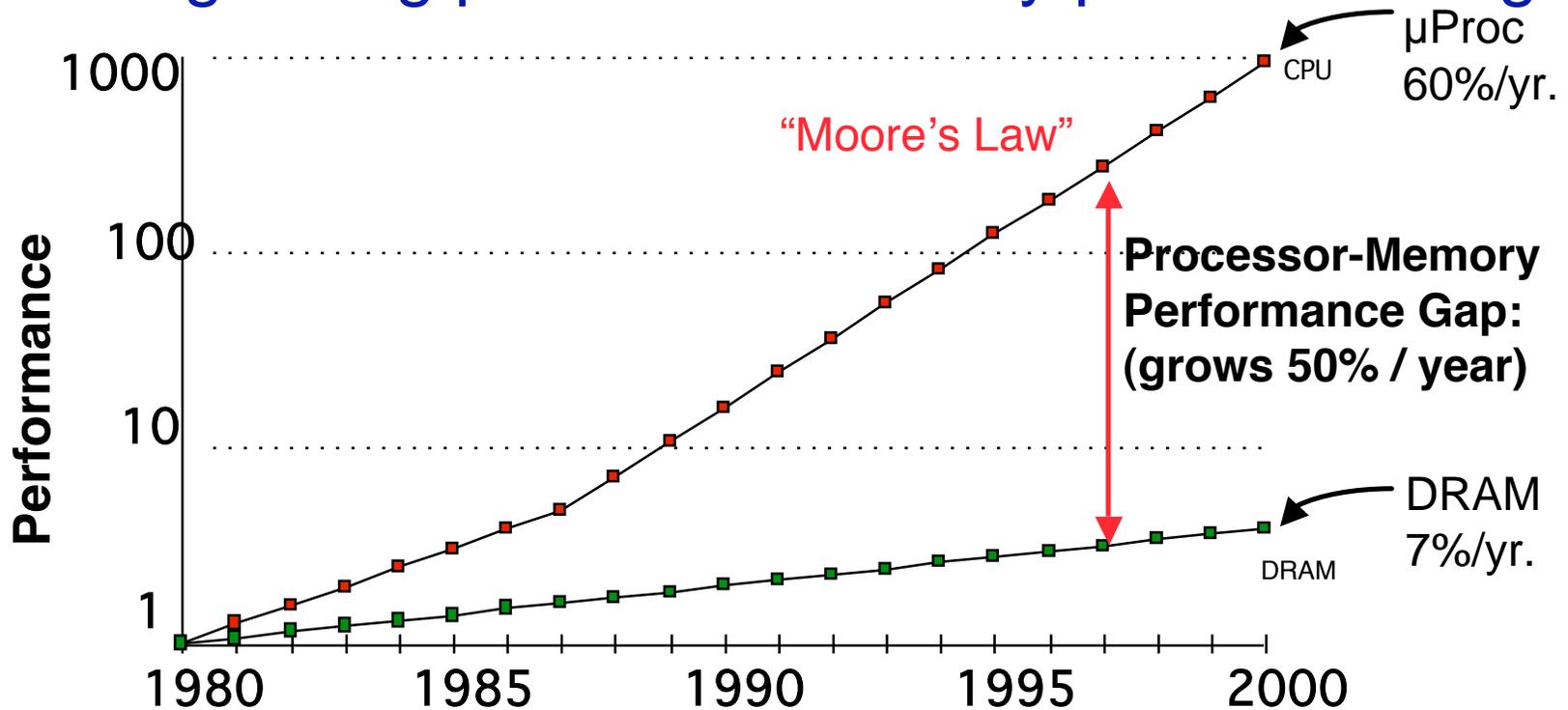
Very Old: Multiplies Slow, Loads fast

- Design algorithms to reduce floating point operations
- Machines measured on peak flop/s



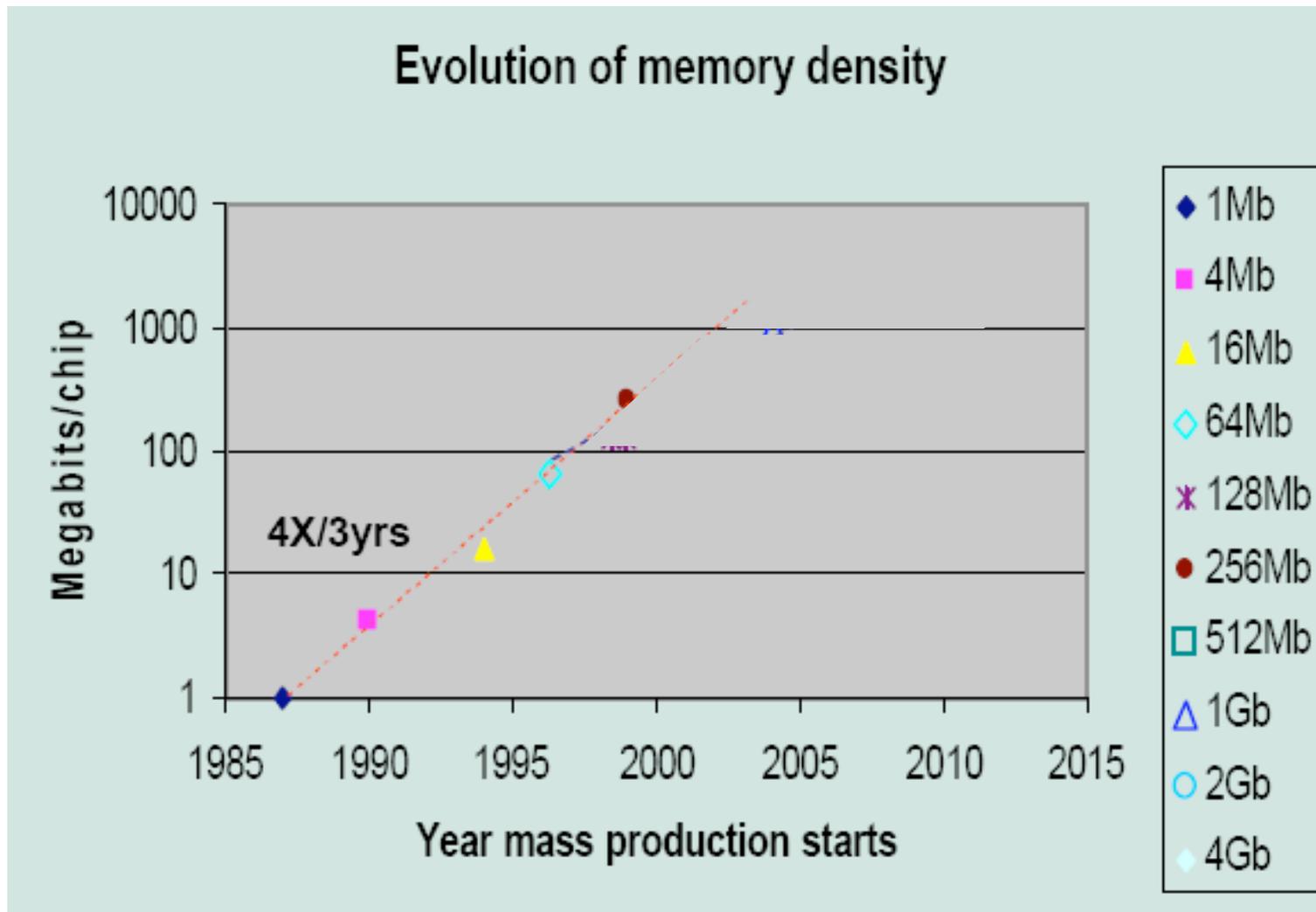
New: Memory Performance is Key

Ever-growing processor-memory performance gap

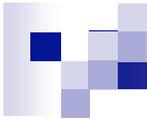


- Total chip performance still growing with Moore's Law
- Bandwidth rather than latency will be growing concern

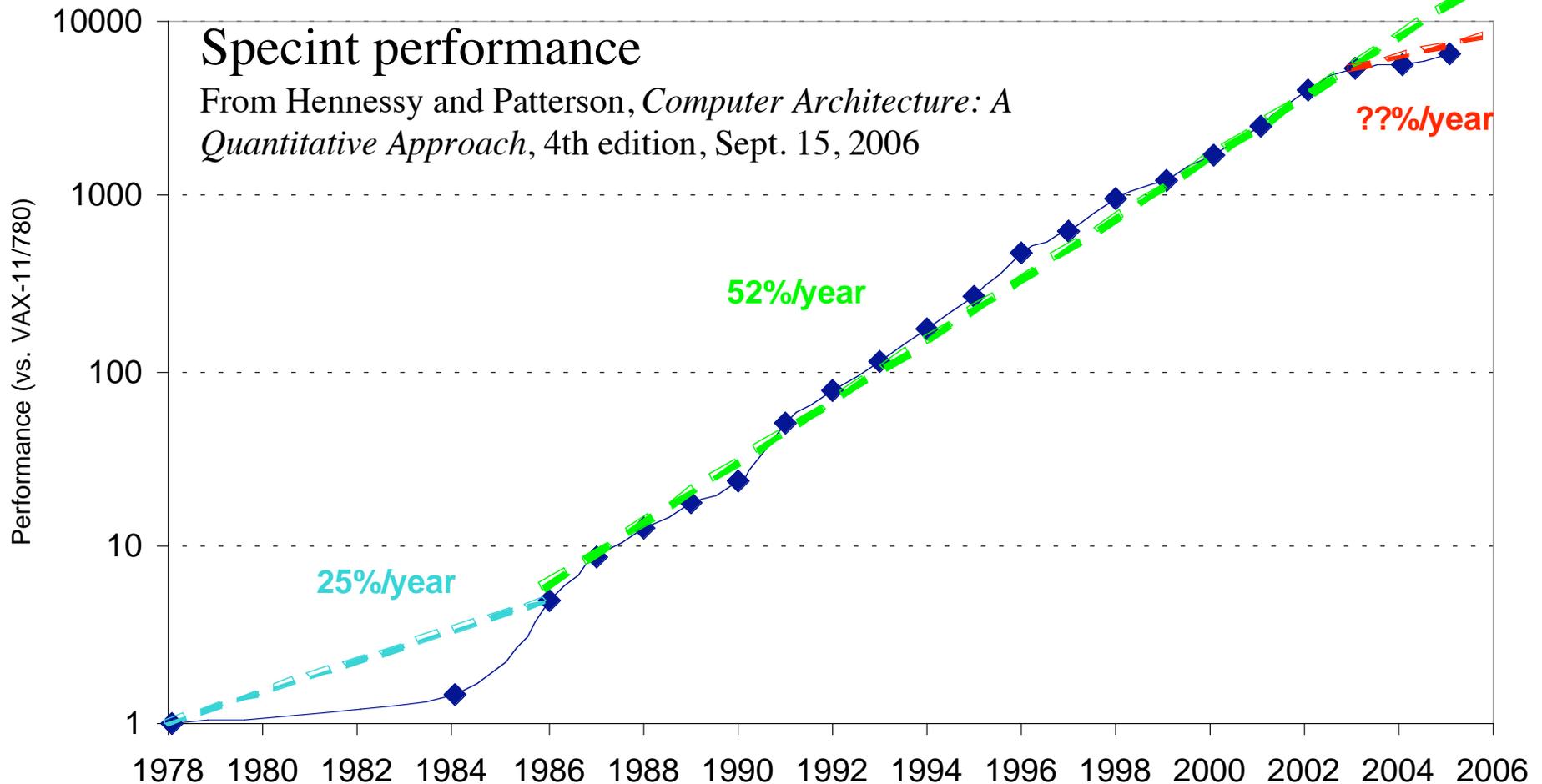
New: Fixed memory capacity will more chips/\$



Source: IBM



Old: Moore's Law → Clock 2x in 18 months

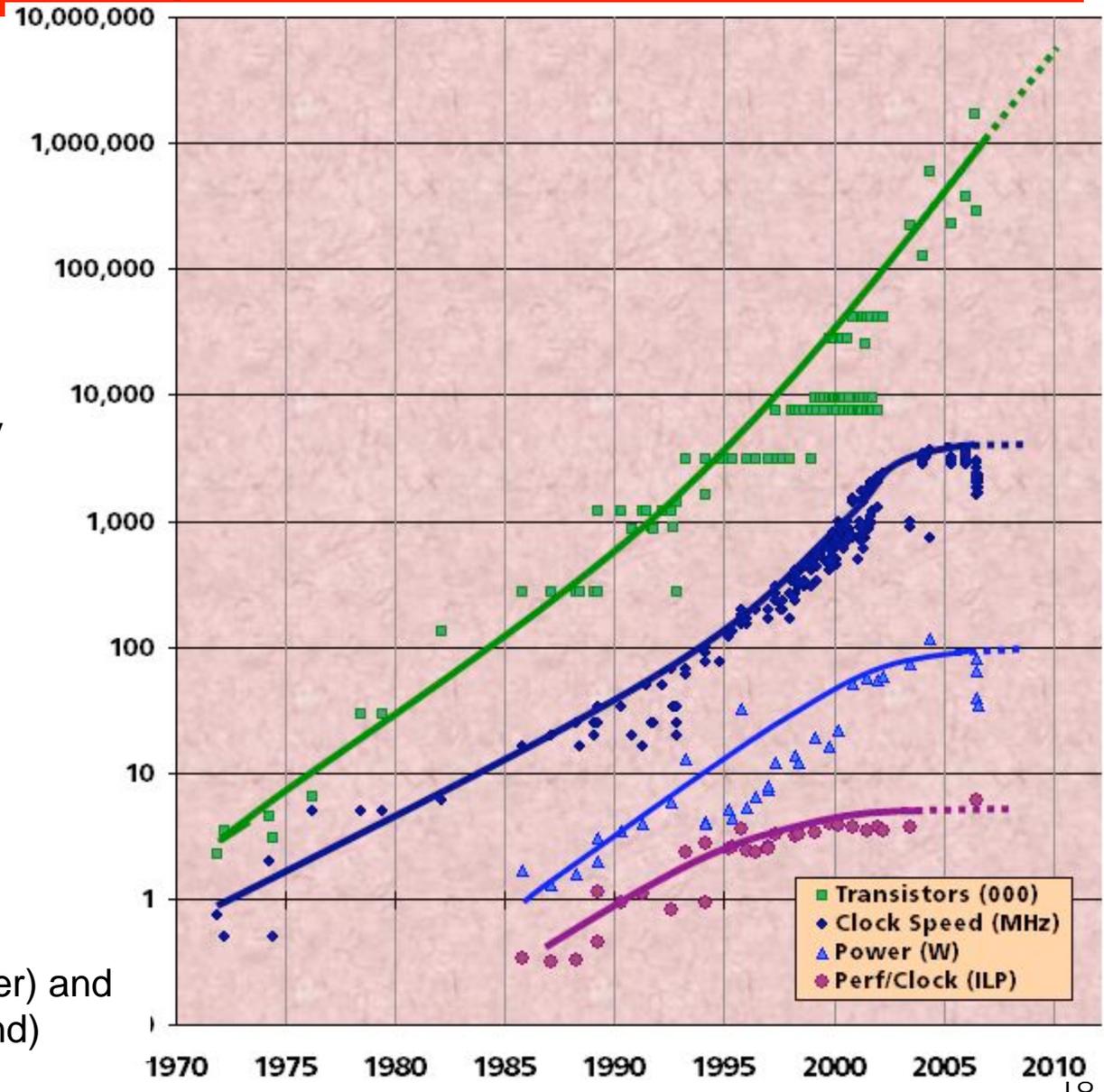


- VAX : 25%/year 1978 to 1986
- RISC + x86: 52%/year 1986 to 2002

New: Cores per chip will double instead

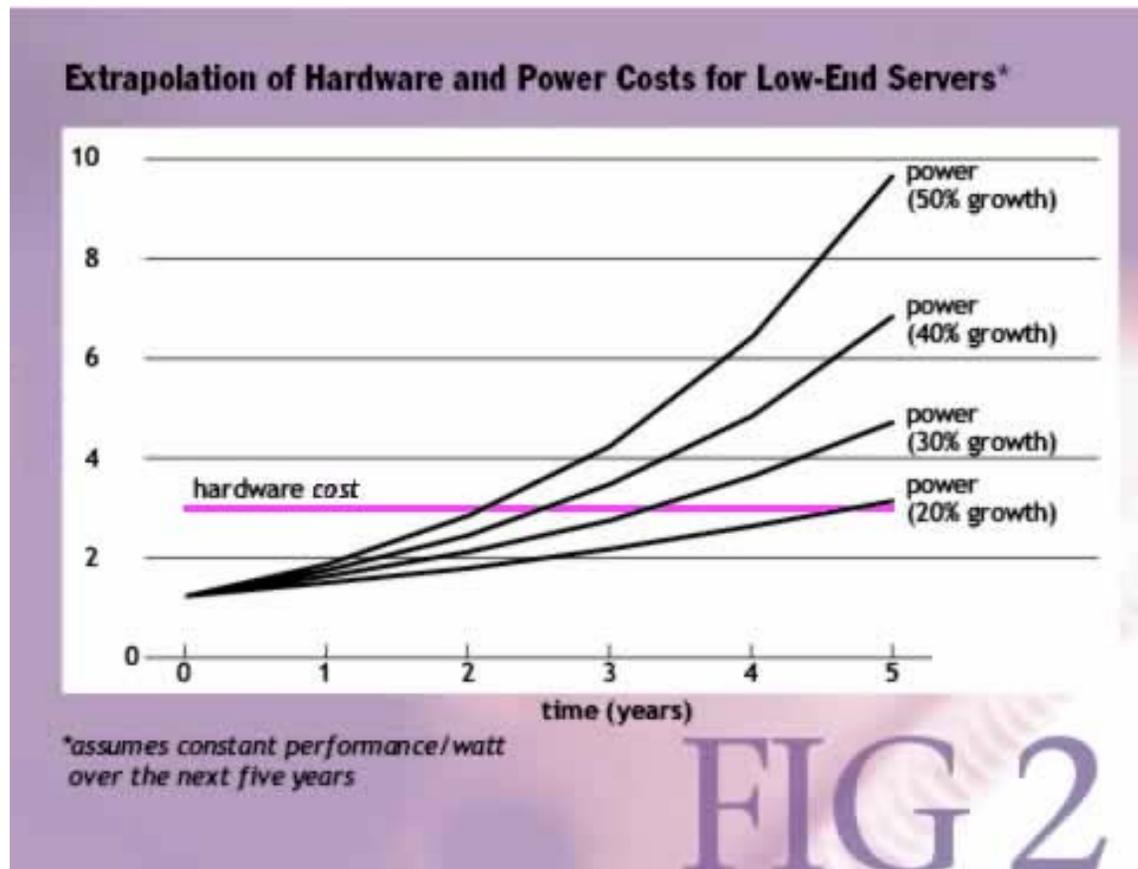
- Chip density is continuing increase ~2x every 2 years
 - Clock speed is not
 - Number of processor cores may double instead
- There is little or no hidden parallelism (ILP) to be found
- Parallelism must be exposed to and managed by software

Source: Intel, Microsoft (Sutter) and Stanford (Olukotun, Hammond)



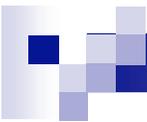
New: Energy budgets could soon dominate facility costs

Estimated 20-200 MWatts for exascale (10^{18} ops/sec) systems in 2018



Unrestrained IT power consumption could eclipse hardware costs and put great pressure on affordability, data center infrastructure, and the environment.

Source: Luiz André Barroso, (Google) "The Price of Performance," *ACM Queue*, Vol. 2, No. 7, pp. 48-53, September 2005. (Modified with permission.)

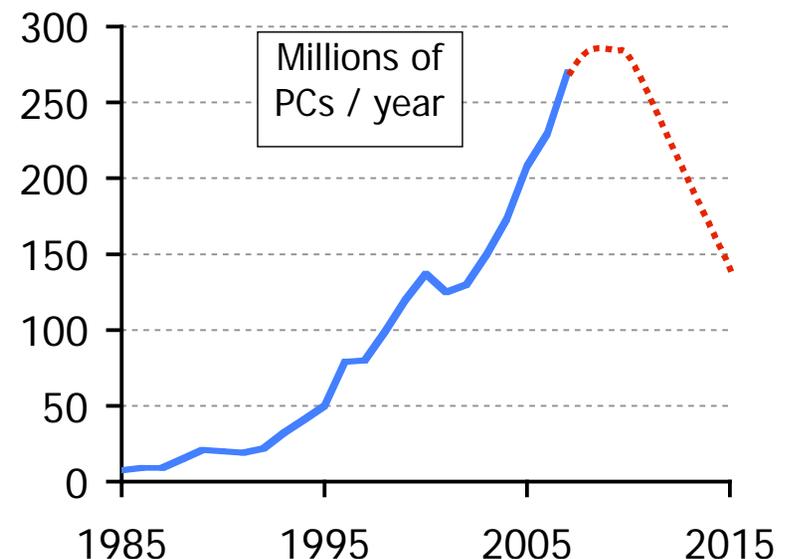


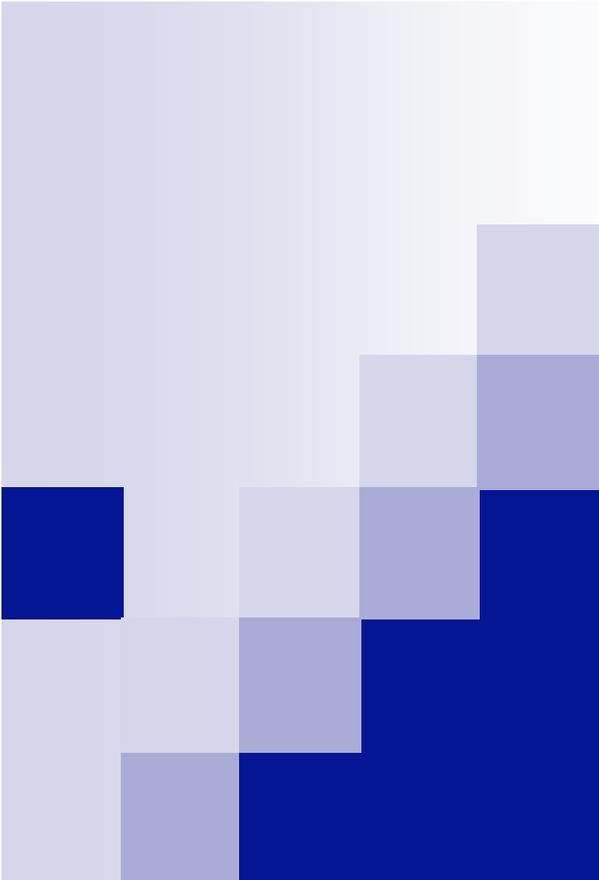
Summary of Trends

- Power is dominant concern for future systems
- Primary “knob” to reduce power is lower clock rates an increase parallelism
- The memory wall (latency and bandwidth) will continue to get worse
- Memory capacity will be increasingly limited/costly
- Entire spectrum of computing will need to address parallelism => performance is a software problem
 - Handheld devices: to keep battery power
 - Laptops/desktops: each new “feature” requires saving time elsewhere
 - High end computing facilities and data centers: to reduce energy costs
- DOE is no long alone in dealing with parallelism

Parallel Revolution May Fail

- The industry is betting on parallelism:
“Jobs said Apple would focus principally on technology for the next generation of the industry’s increasingly parallel computer processors.” NYTimes, 6/10/08.
- Close to 100% failure rate of Parallel Computer Companies
 - Convex, Encore, Inmos (Transputer), MasPar, NCUBE, Kendall Square Research, Sequent, (Silicon Graphics), Thinking Machines, ...
- What if IT goes from a growth industry to a replacement industry?
 - If SW can’t effectively use 32, 64, ... cores per chip
 - ⇒ SW no faster on new computer
 - ⇒ Only buy if computer wears out

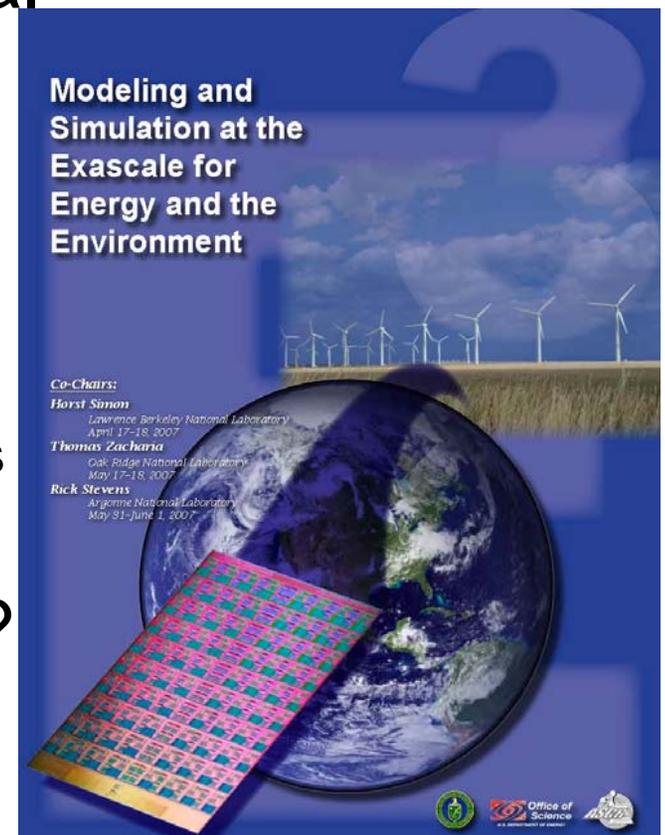


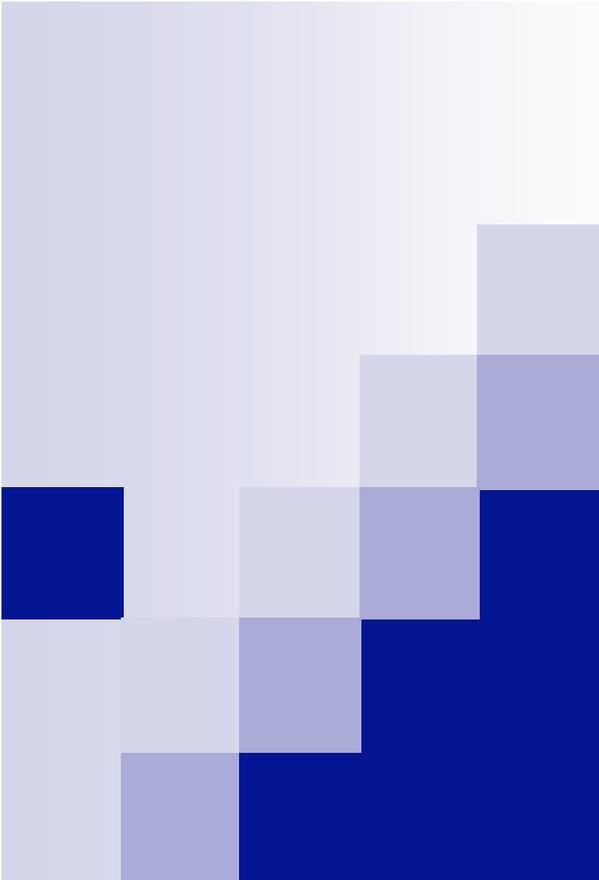


Multicore in High Performance Computing

Applications of Exascale

- Scientists will always find ways of using the largest computational systems
- Some applications highlighted in the 2008 Exascale report:
 - Climate modeling
 - Energy systems modeling with economics
 - Biological system model
- Can we leverage mainstream?
 - Hardware, e.g., embedded systems
 - Software solutions





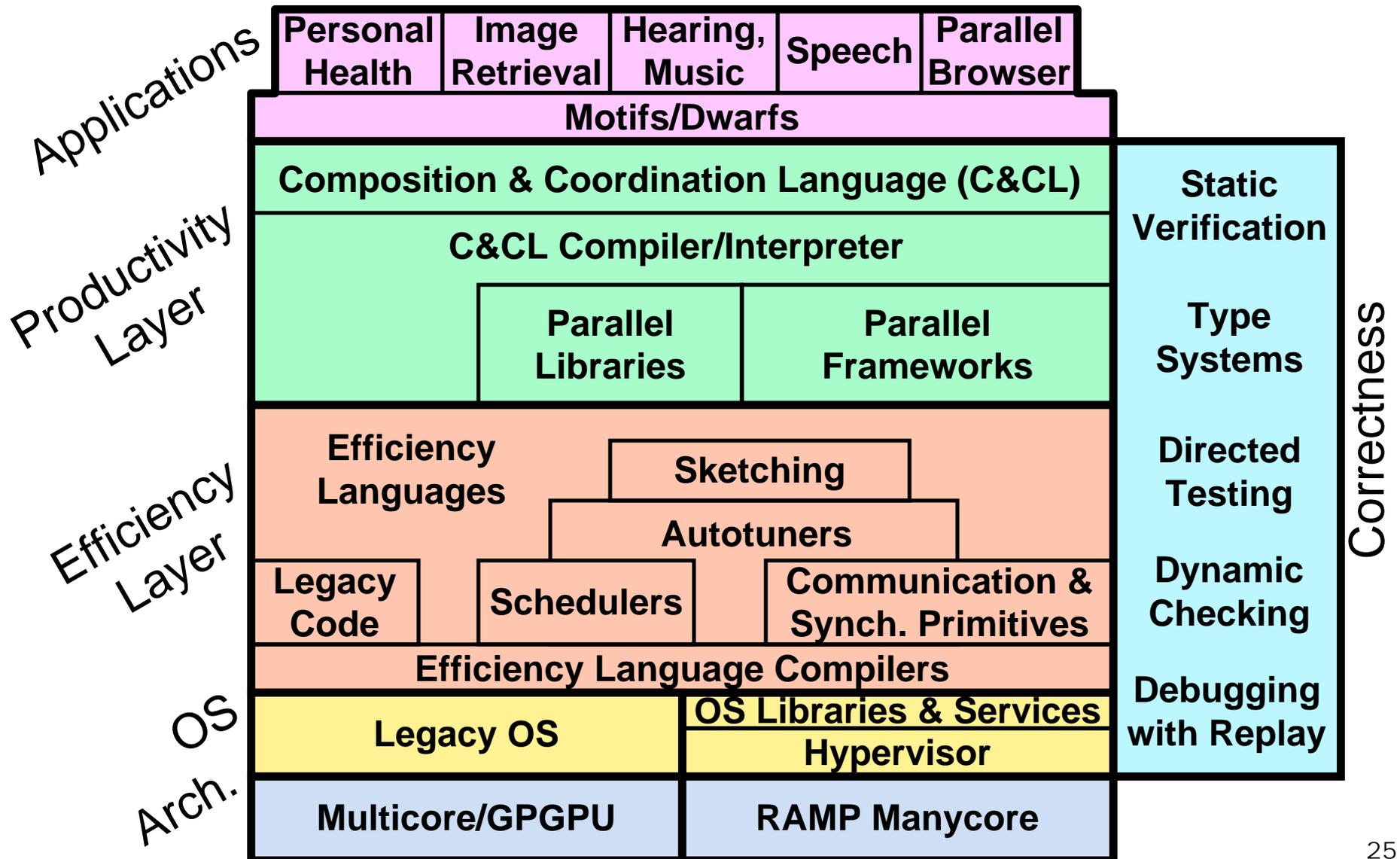
ParLab Project

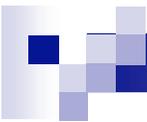
funded by Microsoft, Intel
and UC Discovery

Krste Asanovic, Ras Bodik, Jim Demmel,
John Kubiawicz, Kurt Keutzer, Edward Lee,
George Necula, [Dave Patterson](#), Koushik Sen,
John Shalf, John Wawrzynek, and Kathy Yelick

Par Lab Research Overview

Easy to write correct programs that run efficiently on manycore





Applications: What are the problems?

“Who needs 100 cores to run M/S Word?”

- Need compelling apps that use 100s of cores

How did we pick applications?

1. Enthusiastic expert application partner, leader in field, promise to help design, use, evaluate our technology
2. Compelling in terms of likely market or social impact, with short term feasibility and longer term potential
3. Requires significant speed-up, or a smaller, more efficient platform to work as intended
4. As a whole, applications cover the most important
 - Platforms (handheld, laptop, games)
 - Markets (consumer, business, health)

Compelling Laptop/Handheld Apps (David Wessel)

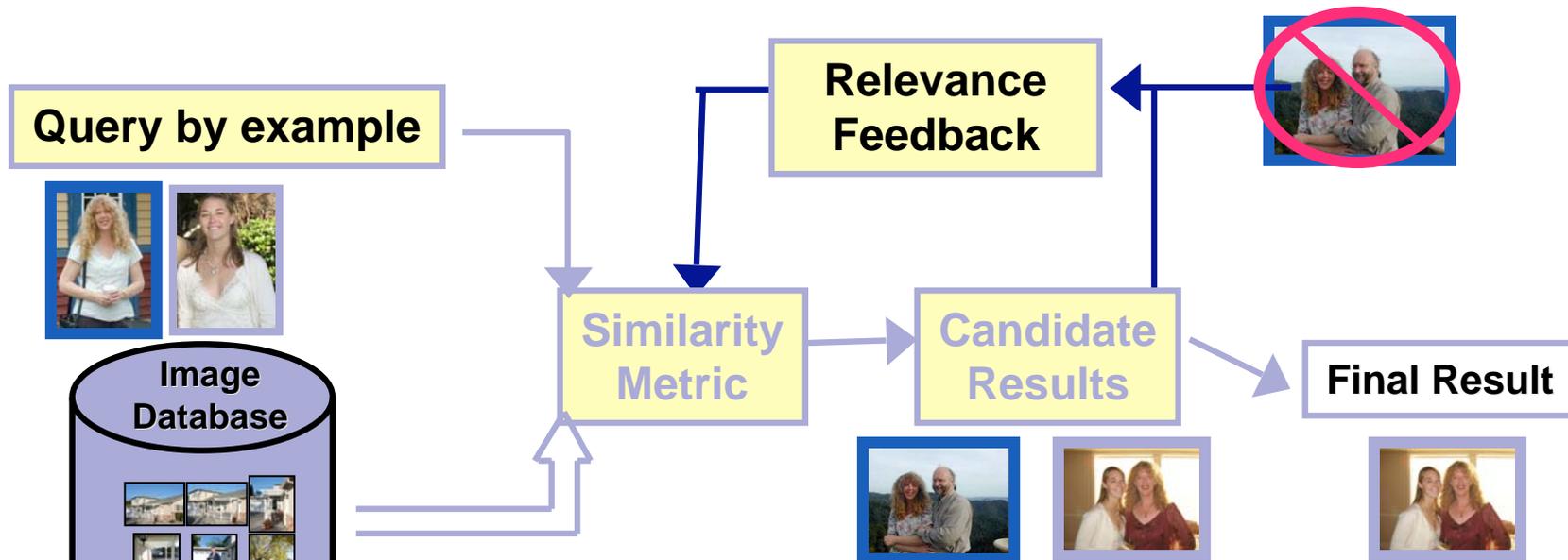
- Musicians have an insatiable appetite for computation
 - More channels, instruments, more processing, more interaction!
 - Latency must be low (5 ms)
 - Must be reliable (No clicks)
- Music Enhancer
 - Enhanced sound delivery systems for home sound systems using large microphone and speaker arrays
 - Laptop/Handheld recreate 3D sound over ear buds
- Hearing Augmenter
 - Handheld as accelerator for hearing aid
- Novel Instrument User Interface
 - New composition and performance systems beyond keyboards
 - Input device for Laptop/Handheld



Berkeley Center for New Music and Audio Technology (CNMAT) created a compact loudspeaker array: 10-inch-diameter icosahedron incorporating 120 tweeters.

Content-Based Image Retrieval

(Kurt Keutzer)



1000's of images

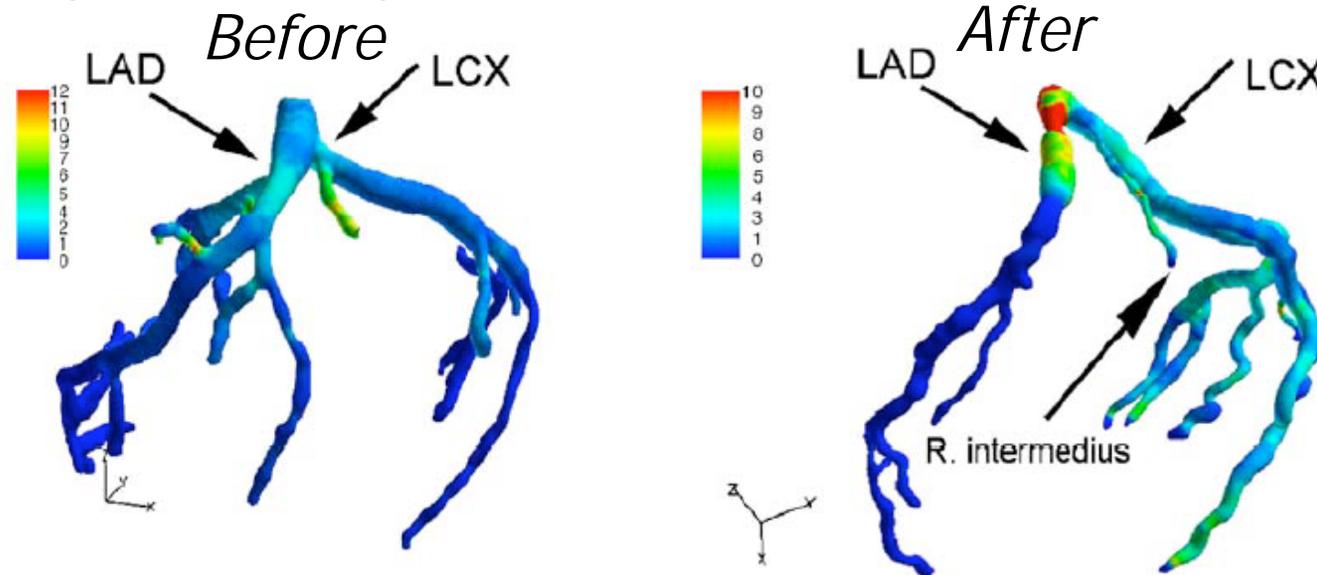
■ Built around Key Characteristics of personal databases

- Very large number of pictures (>5K)
- Non-labeled images
- Many pictures of few people
- Complex pictures including people, events, places, and objects



Coronary Artery Disease

(Tony Keaveny)

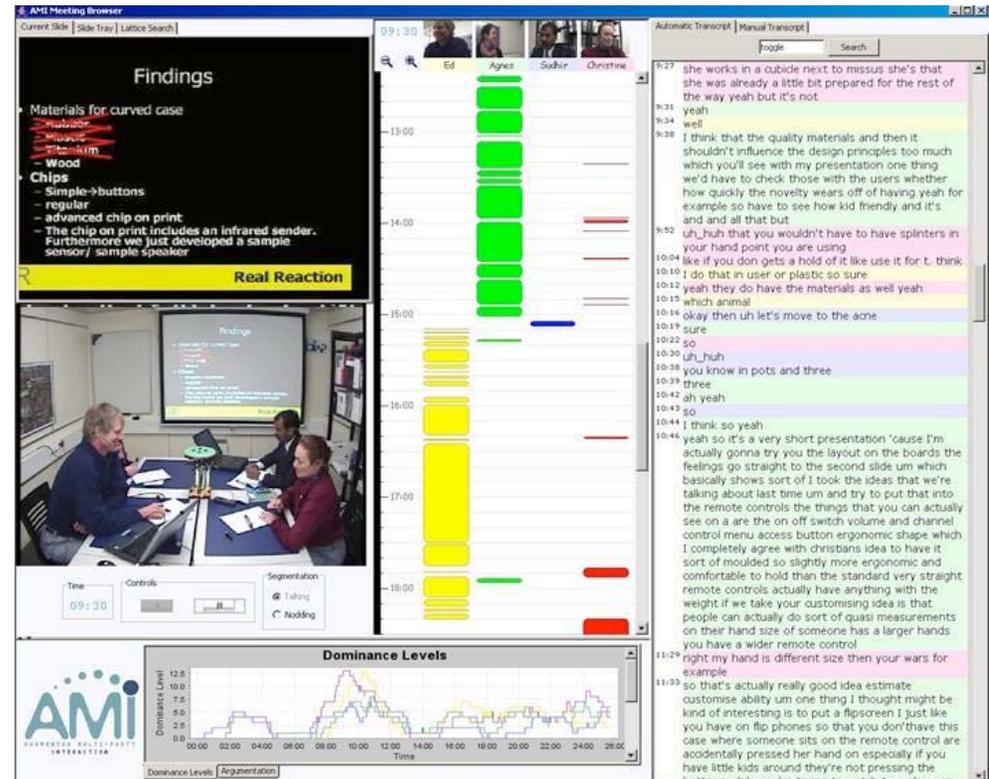


- Modeling to help patient compliance?
 - 450k deaths/year, 16M w. symptom, 72M↑BP
- Massively parallel, Real-time variations
 - CFD FE solid (non-linear), fluid (Newtonian), pulsatile
 - Blood pressure, activity, habitus, cholesterol

Meeting Diarist and Teleconference Aid (Nelson Morgan)

■ Meeting Diarist

- Laptops/ Handhelds at meeting coordinate to create speaker identified, partially transcribed text diary of meeting



• Teleconference speaker identifier

- L/Hs used for teleconference, identifies who is speaking, “closed caption” hint of what being said

Parallel Browser: Web 2.0 in 2 Watts (Ras Bodik)

■ Why parallelizing a browser:

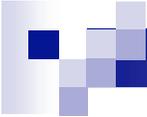
- 1) Web 2.0: Browser plays role of traditional OS
Resource sharing and allocation, Protection
- 2) Will handheld replace laptop?
Enabled by 4G networks, better output devices

■ Bottlenecks: Parsing, Rendering, Scripting

<i>machine</i>	<i>seconds</i>
a modern desktop (2Mbps network)	2
T40 1.6GHz (a very old laptop; 2Mbps network)	7
T40 1.6Ghz (laptop in battery mode, 2Mbps)	13
iPhone 600MHz (2Mbps network)	37
iPhone 600MHz (1Mbps network)	40

■ “SkipJax”

- Parallel replacement for JavaScript/AJAX



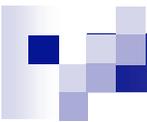
Developing Parallel Software

- 2 types of programmers \Rightarrow 2 layers
- **Efficiency Layer** (10% of today's programmers)
 - Expert programmers build **Frameworks & Libraries, Hypervisors, ...**
 - **“Bare metal” efficiency possible at Efficiency Layer**
- **Productivity Layer** (90% of today's programmers)
 - **Domain experts / Naïve programmers productively build parallel apps using frameworks & libraries**
 - **Frameworks & libraries composed to form app frameworks**
- Identify set of key computational methods
 - **Language for interdisciplinary project; focus for programming and architecture work**

Motif/Dwarf: Common Computational Methods

(Red Hot → Blue Cool)



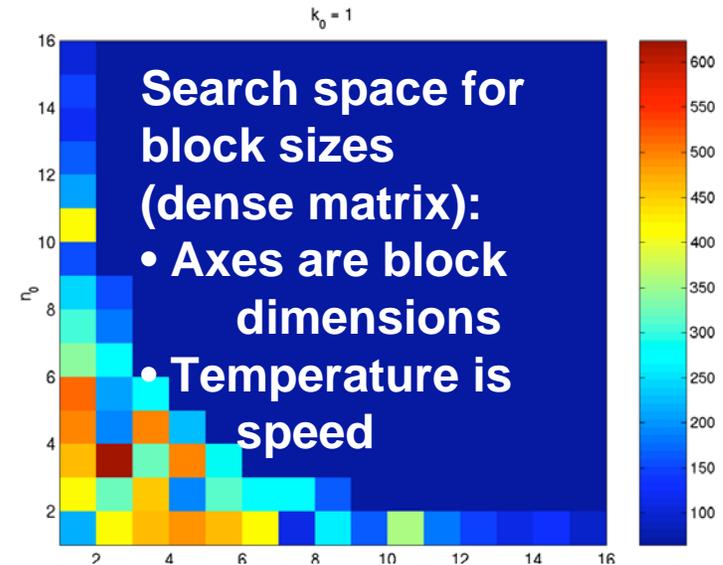


Productivity Language Strategy

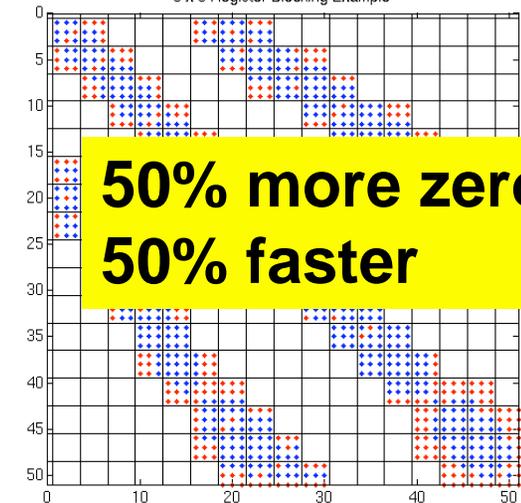
- **Application-driven: Domain-specific languages**
 - Ensure usefulness for at least one application
 - Music language
 - Image framework
 - Browser language
 - Health application language
- **Bottom-up implementation strategy**
 - Ensure efficiently implementable
 - “Grow” a language from one that is efficient but not productive by abstraction levels
- **Identify common features across domains**
 - Cross-language meetings/discussions

Autotuning: 21st Century Code Generation

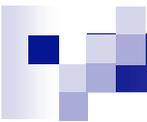
- Problem: generating optimal code is like searching for needle in a haystack
- Manycore \Rightarrow even more diverse
- New approach: “Auto-tuners”
 - 1st generate program variations of combinations of optimizations (prefetching, ...) and data structures
 - Then compile and run to search for best code for *that* computer
- Examples: PHiPAC (Dense LA), Atlas (Dense LA), Spiral (DSP), FFTW (FFT), OSKI (Sparse LA), Stencils (Convolutions, etc.)
 - Change data structures
 - Rely on high level properties
 - Write programs to generate code



3 x 3 Register Blocking Example



(688 true non-zeros) + (383 explicit zeros) = 1071 nz



Deconstructing Operating Systems

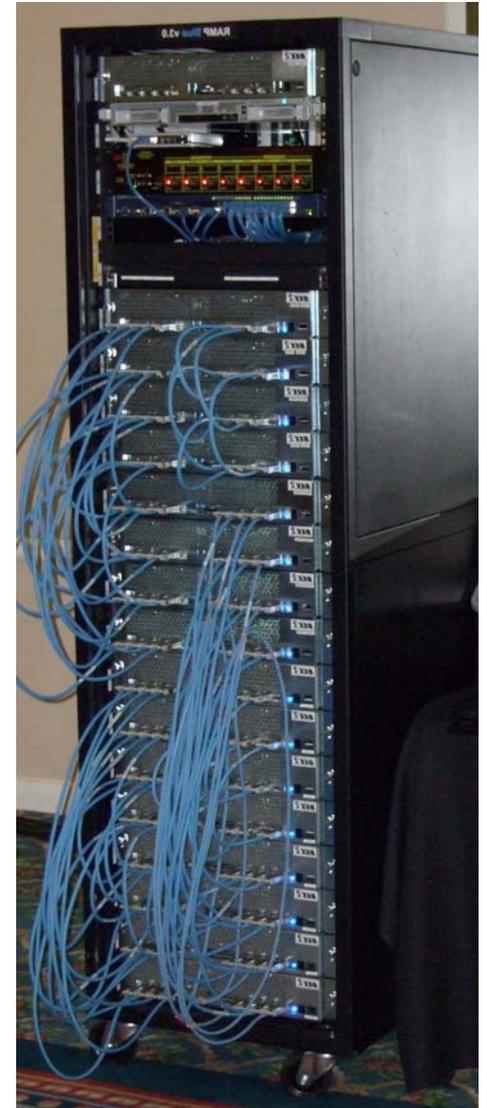
(Krste Asanovic, John Kubiatowicz)

- Resurgence of interest in virtual machines
 - Hypervisor: thin SW layer btw guest OS and HW
- Future OS: libraries where only functions needed are linked into app, on top of thin hypervisor providing protection and sharing of resources
 - Opportunity for OS innovation
- Very thin hypervisors, and to allow software full access to hardware within partition

Build Academic Manycore from FPGAs

(Krste Asanovic, John Wawrzynek)

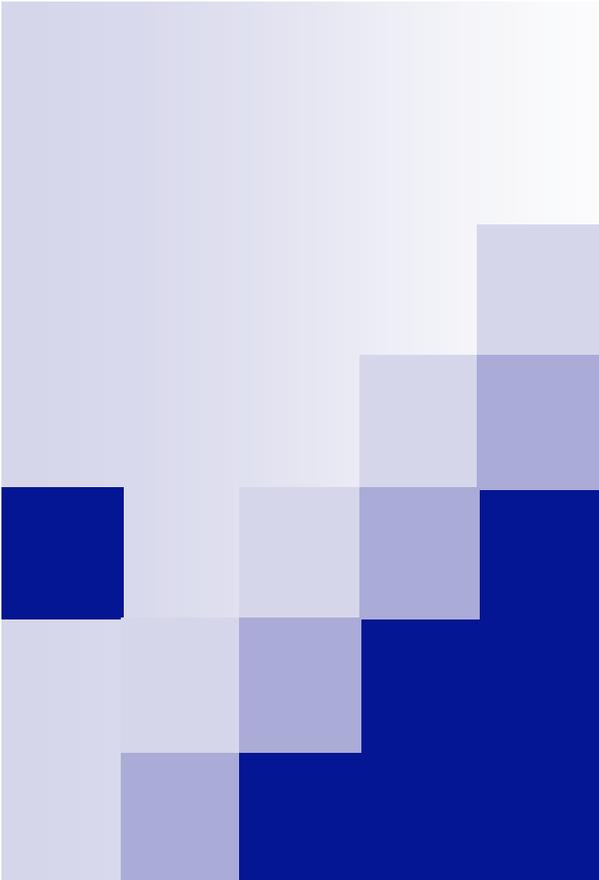
- As ≈ 10 CPUs will fit in Field Programmable Gate Array (FPGA), 1000-CPU system from ≈ 100 FPGAs?
 - 8 32-bit simple “soft core” RISC at 100MHz in 2004 (Virtex-II)
- HW research community does logic design (“gate shareware”) to create out-of-the-box, Manycore
 - E.g., 1000 processor, standard ISA binary-compatible, 64-bit, cache-coherent supercomputer @ ≈ 150 MHz/CPU in 2007
 - RAMPants: 10 faculty at Berkeley, CMU, MIT, Stanford, Texas, and Washington
- RAMP: Research Accelerator for Multiple Processors



Radical Co-Location: Part 1 & 2



Par Lab Opening Ceremony Mon. Dec. 1, 2008 11-2



Questions?