The Challenges of Massively Parallel Computing

Rich Loft
Director, Technology Development
Computational and Information Systems Laboratory
National Center for Atmospheric Research
loft@ucar.edu

Logarithmic Units of Fast

- 10^{24} yotta Y
- 10^{21} zetta Z
- 10^{18} exa E
- 10^{15} peta P
- 10^{12} tera T
- 10^{9} giga G
- 10^{6} mega M
- 10^{3} kilo K
- 10^{0} mono U

History of Computing So Far

We've had a pretty good run...

Technology Trends…
ITRS Roadmap

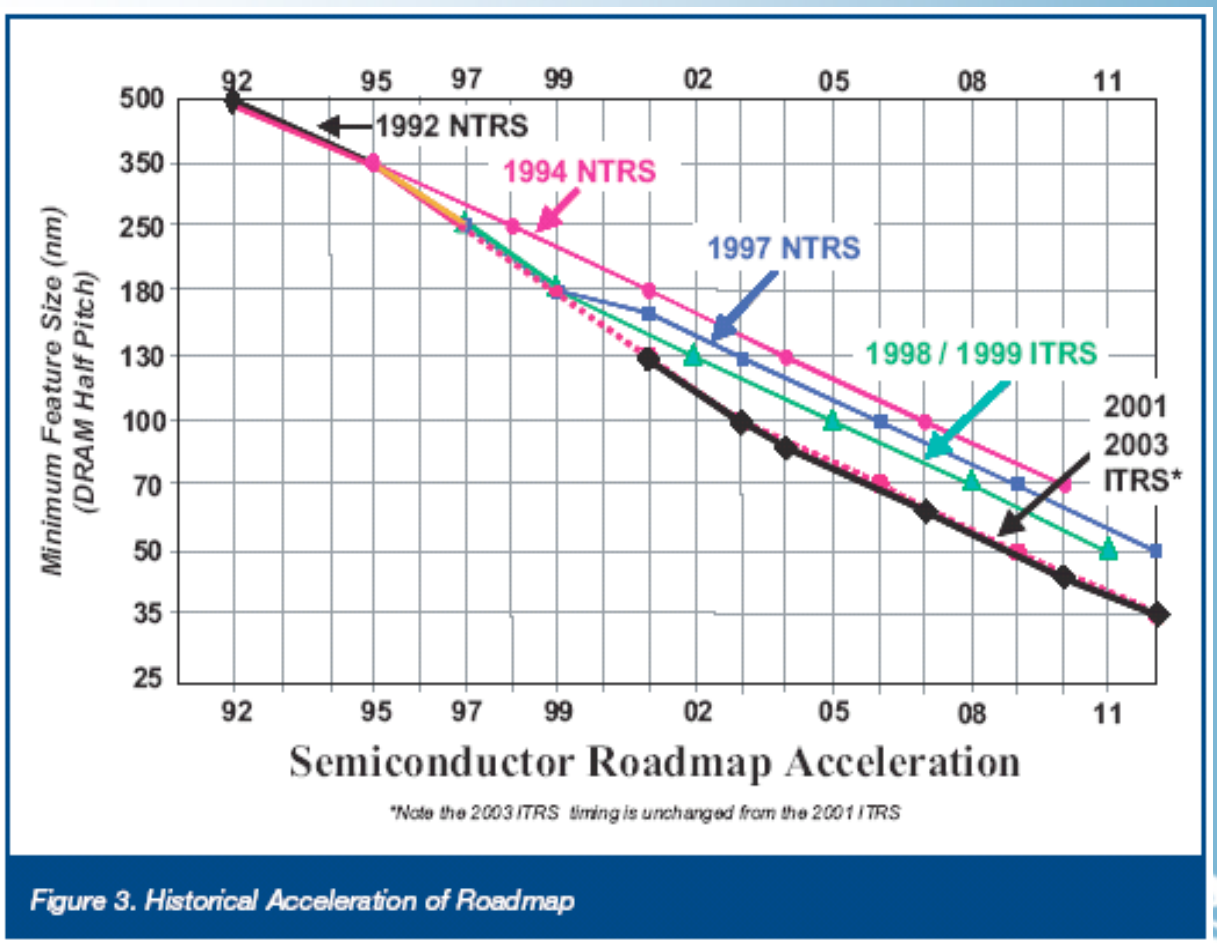
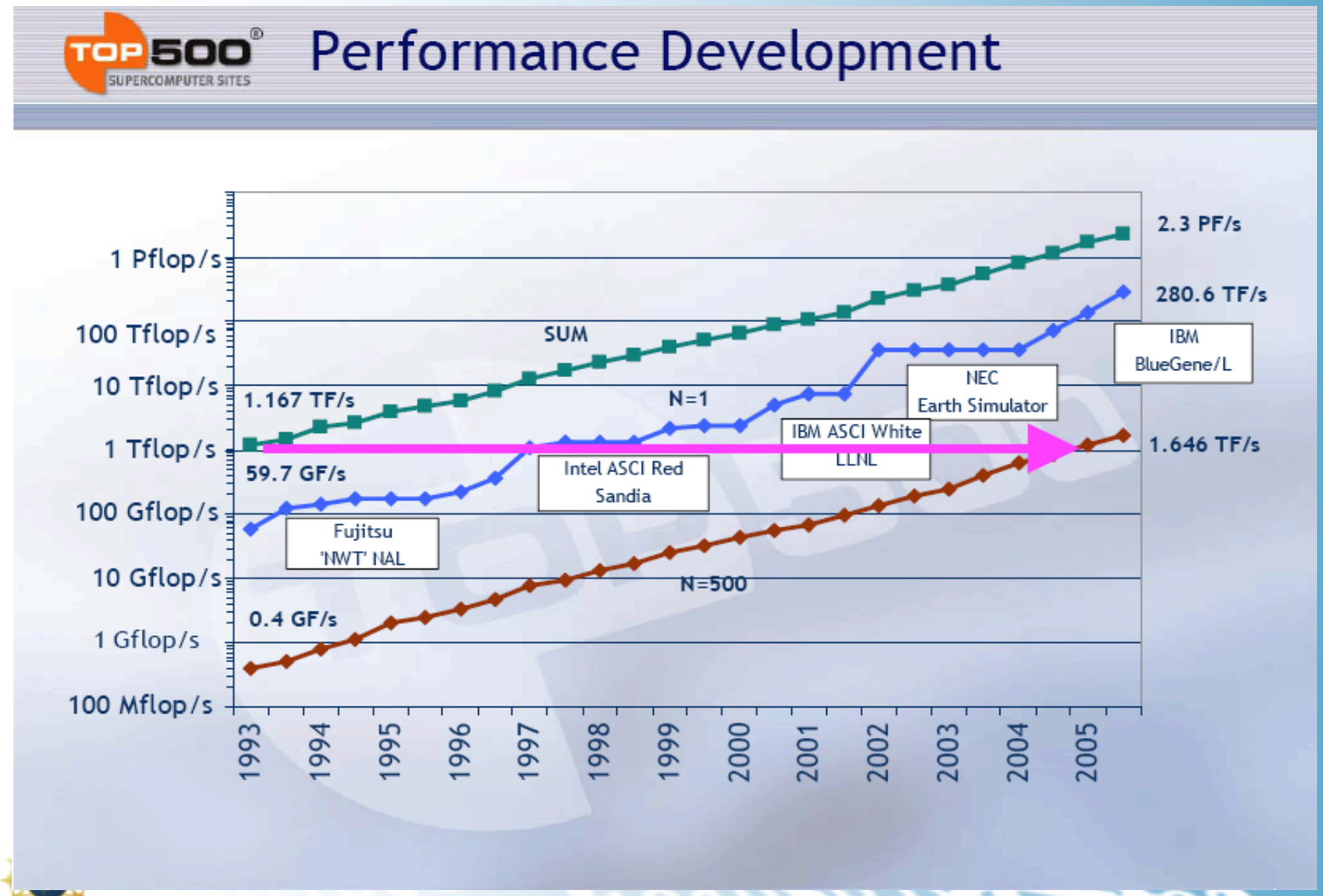


Figure 3. Historical Acceleration of Roadmap

By 2050 reaches the size of an atom

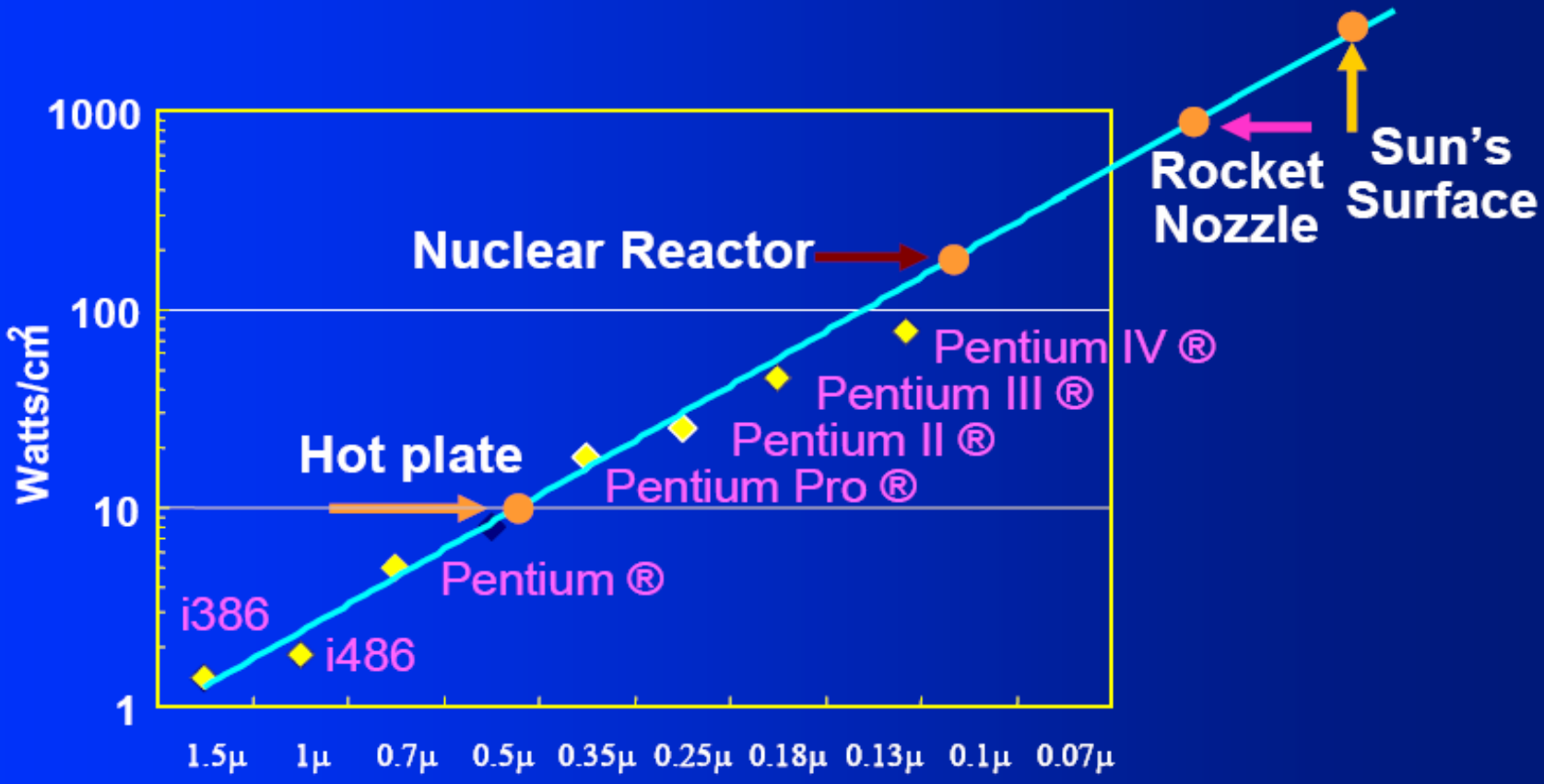
Moore's Law is not fast enough!



...suggests 10^4 to 10^6 improvement will take 20 years

However, there's been an underlying paradigm shift in how progress in HPC technology is occurring

Relentless rise of power density

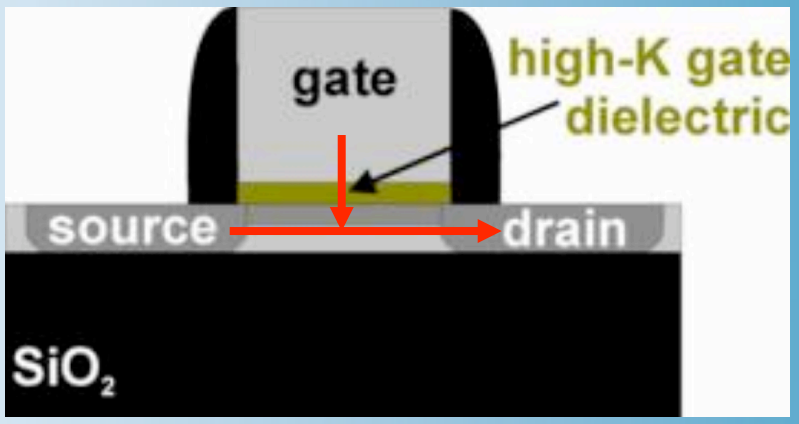


- 80% increase in power density/generation
- Voltage scales by ~0.8
- 225% increase in current consumption/unit area !

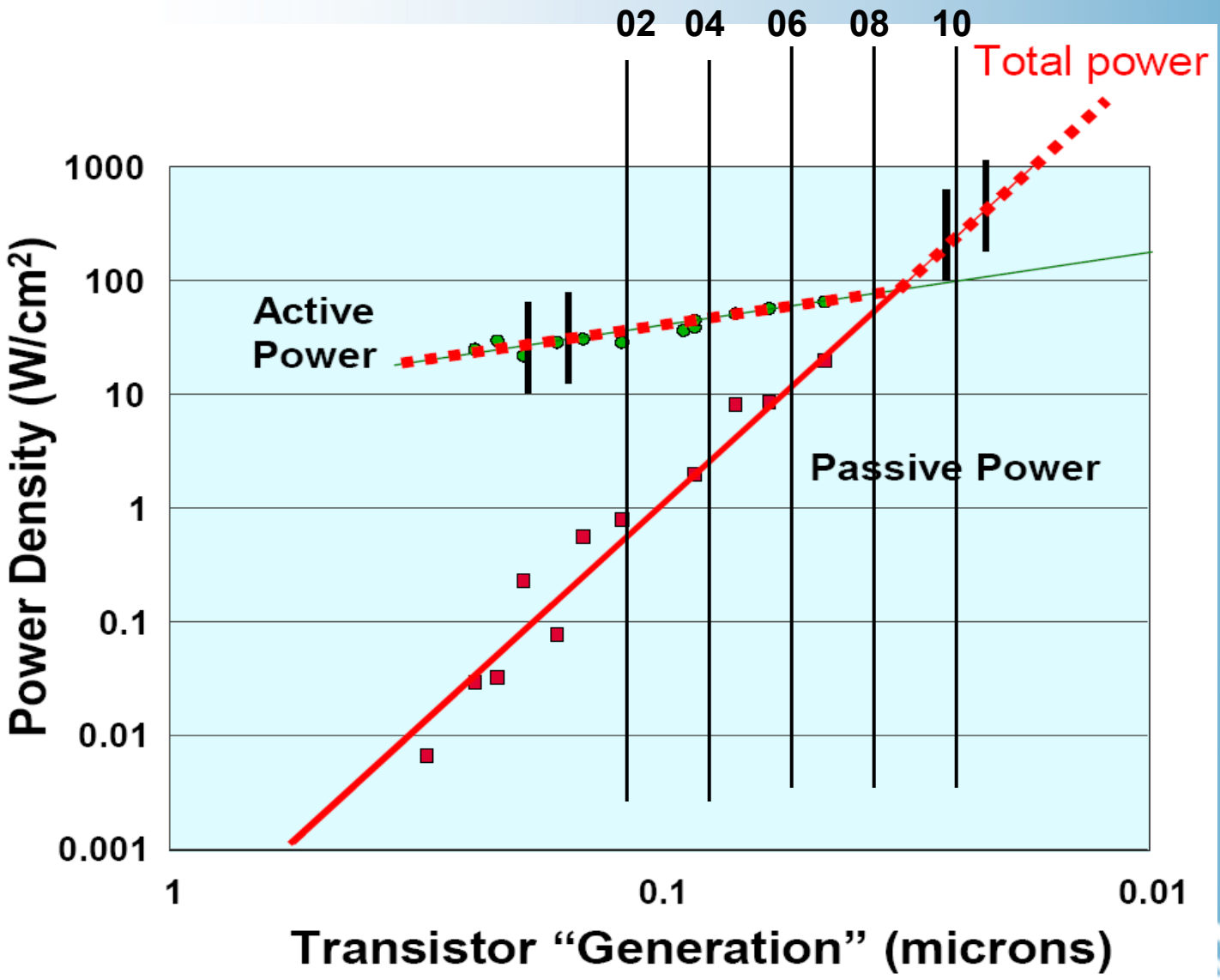
Source: Shekhar Borkar. Intel

It gets even worse: passive power Leaks due to Quantum Tunneling

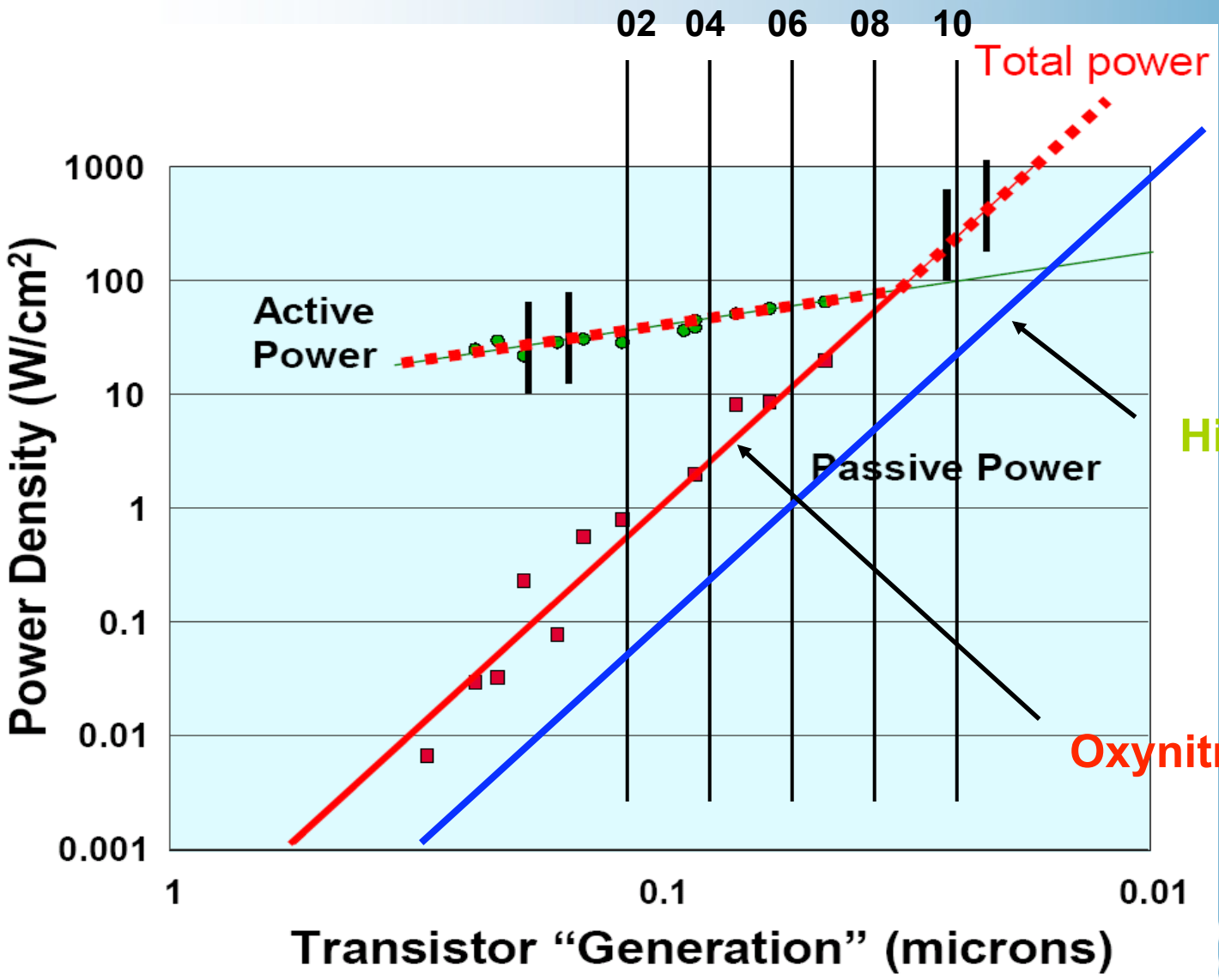
- Long before we hit atomic scales, quantum mechanics will start working against current design
- Exponential increase of gate direct tunneling currents
- short channel effects (SCE) for sub-micron gate-lengths has lead to increasing source-drain leakage currents.
- Leakage currents can be controlled by introducing high dielectric constant gate insulators.



Chip Power Densities



High-K reduces leakage currents…



Forestalling "power-death" of CMOS

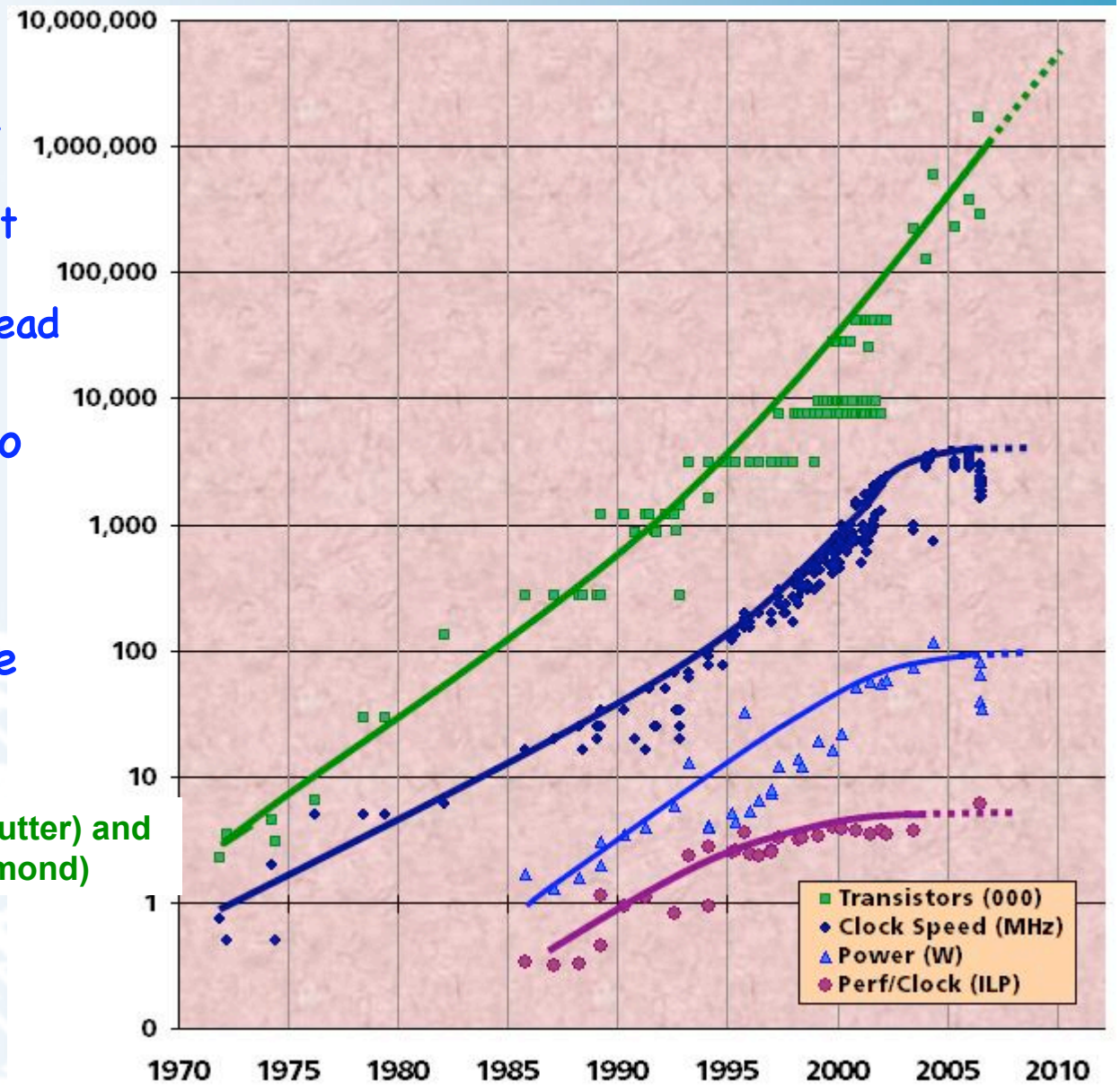
Further Problems: Long wires

- The problem is with sending signals longer distances. If the length of the wire does not scale down *faster* than transistor lengths, wire delays will come to dominate.
- In other words, no long wires to our devices!
- In the long run, the number of devices in a group of transistors that can communicate within 1 gate delay is going to shrink, and so architectures will need to become more and more highly *localized*.
- Some sort of parallel mesh (a.k.a. tile) chip architectures and mesh programming models will be needed in the long run.
- Already see this coming with multi-core chips (Intel, AMD, POWER-7, etc)
- Like it or not we're headed towards Massively parallel with a big-M.

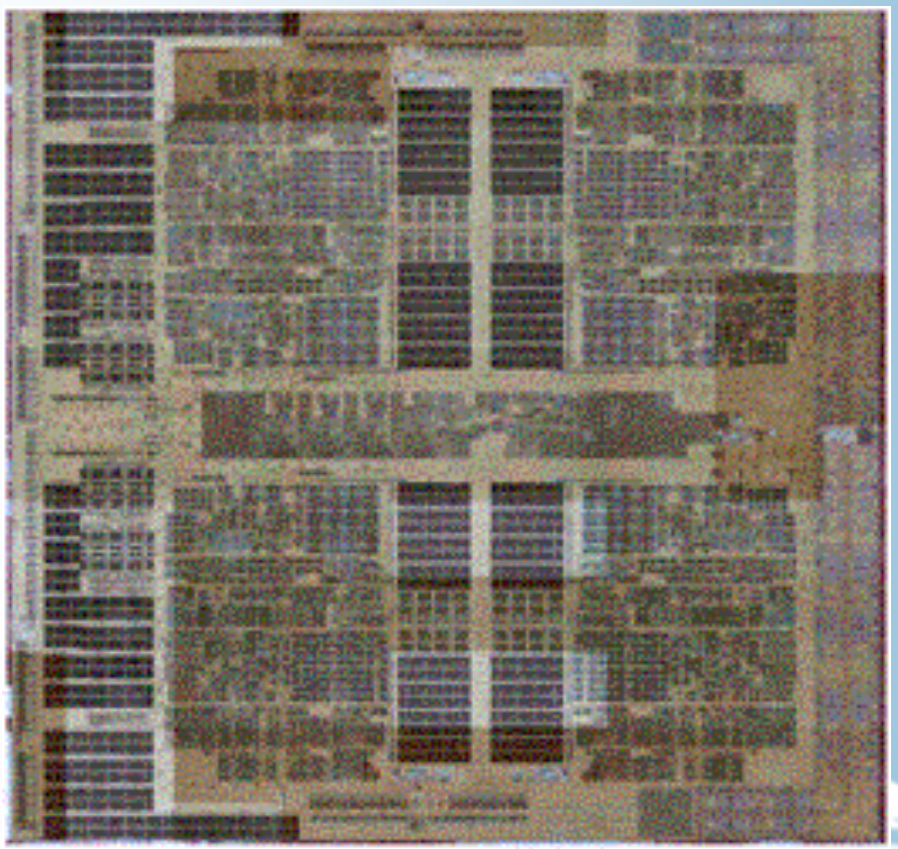
Chip Level Trends

- Chip density is continuing increase ~2x every 2 years
  - Clock speed is not
  - Number of cores are doubling instead
- There is little or no additional hidden parallelism (ILP)
- Parallelism must be exploited by software

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

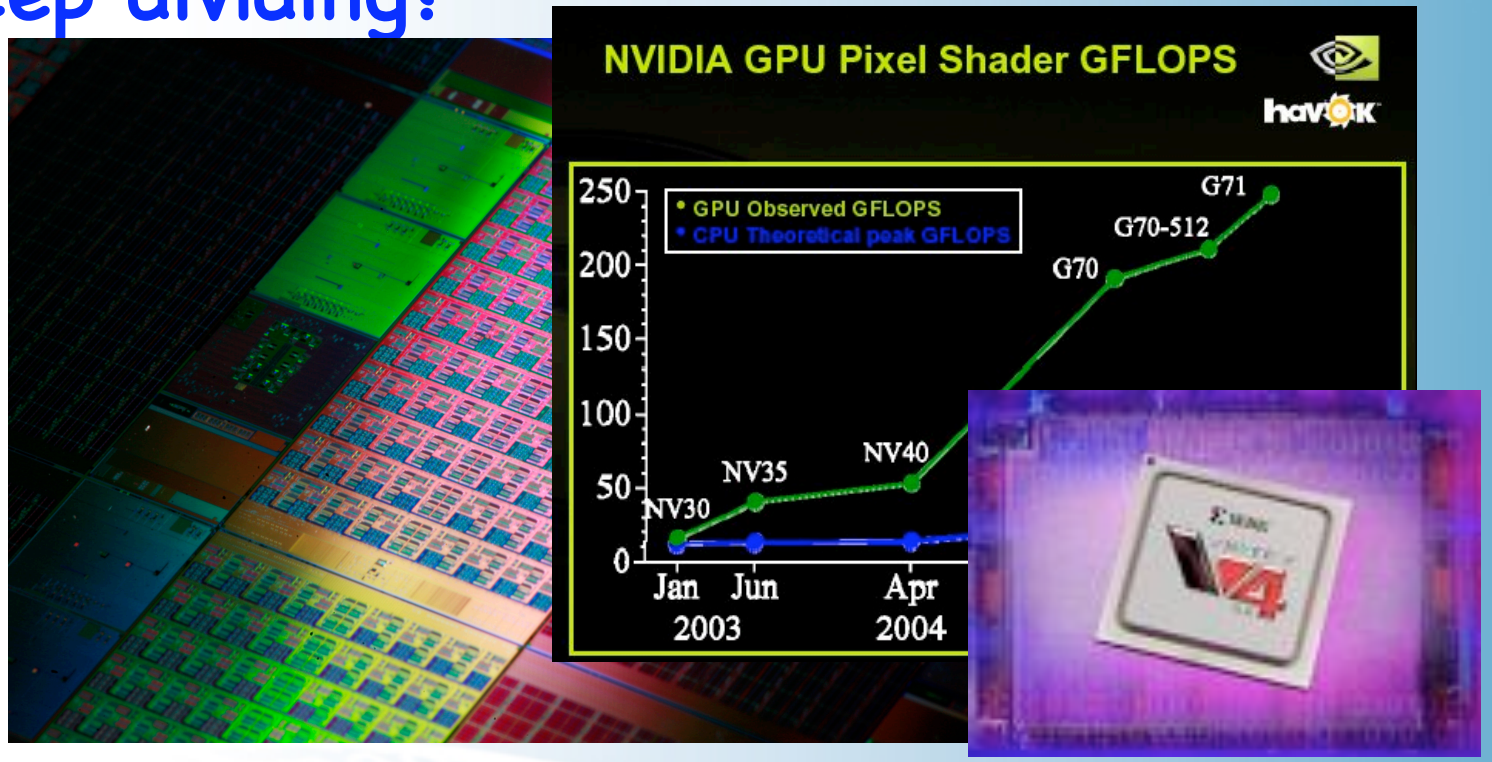


Moore's Law = More Cores: Quad Core "Barcelona" AMD Processor…



Can 8, 16, 32 cores be far behind?

How long can the "chip blastula" keep dividing?



- Intel tested 1 TFLOPS O(80) cores/socket
- Paradigm shift?
  - GP-GPU
  - FPGA - 21.7 x in V5 simulator on CAM sw-radiation code.

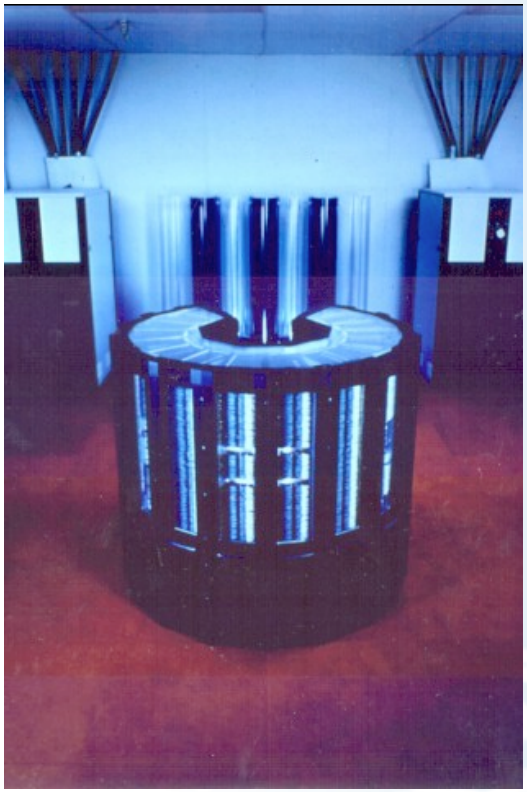
Are we near the Thermodynamic Limits of Irreversible Computing?

- Min energy required to flip a bit irreversibly:
  - kT ln(2) , k = 1.3806503x10^{-23} joule/K
  - Landauer's principle
- Min energy to reliably flip a bit and keep it flipped
  - ~80*kT
- Min bit flips to perform one FLOP:
  - ~100,000
- Min energy to compute one irreversible FLOP at 60C:
  - 3.678x10^{-14} joules/FLOP -> Watts/FLOPS
  - 27 TFLOPS/Watt
  - 27 ExaFLOPS/MWatt

Irreversible Computing: Good News/Bad News

- Good News: Lots of Room for Improvement!
- Current IBM Blue Gene/L processor fuel efficiency is
  - 800 Rmax MFLOPS/Watt
  - Improvement factor of 30K - 100K possible!
  - Brain Synapse: 200×10^6 MSyOPs/Watt
    - ~10x better than FLOPS limit derived from Landauer's
- Bad news: we have Exponential Expectations!
  - 30 years to reach this theoretical limit if fuel efficiency increases at current rates (doubling every two years).
  - Stronger constraint than reaching the atomic scale.

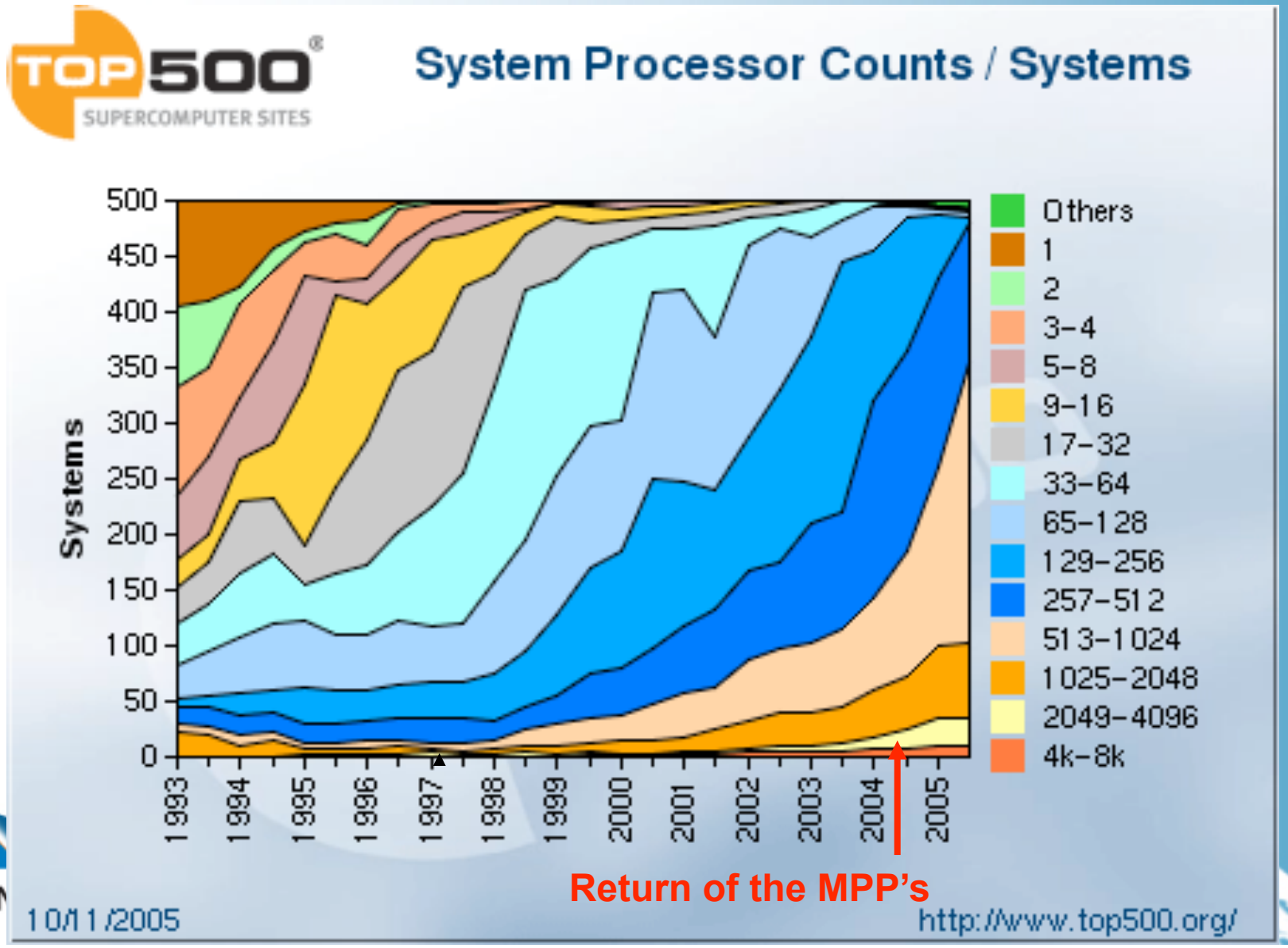
The future is massively parallel…



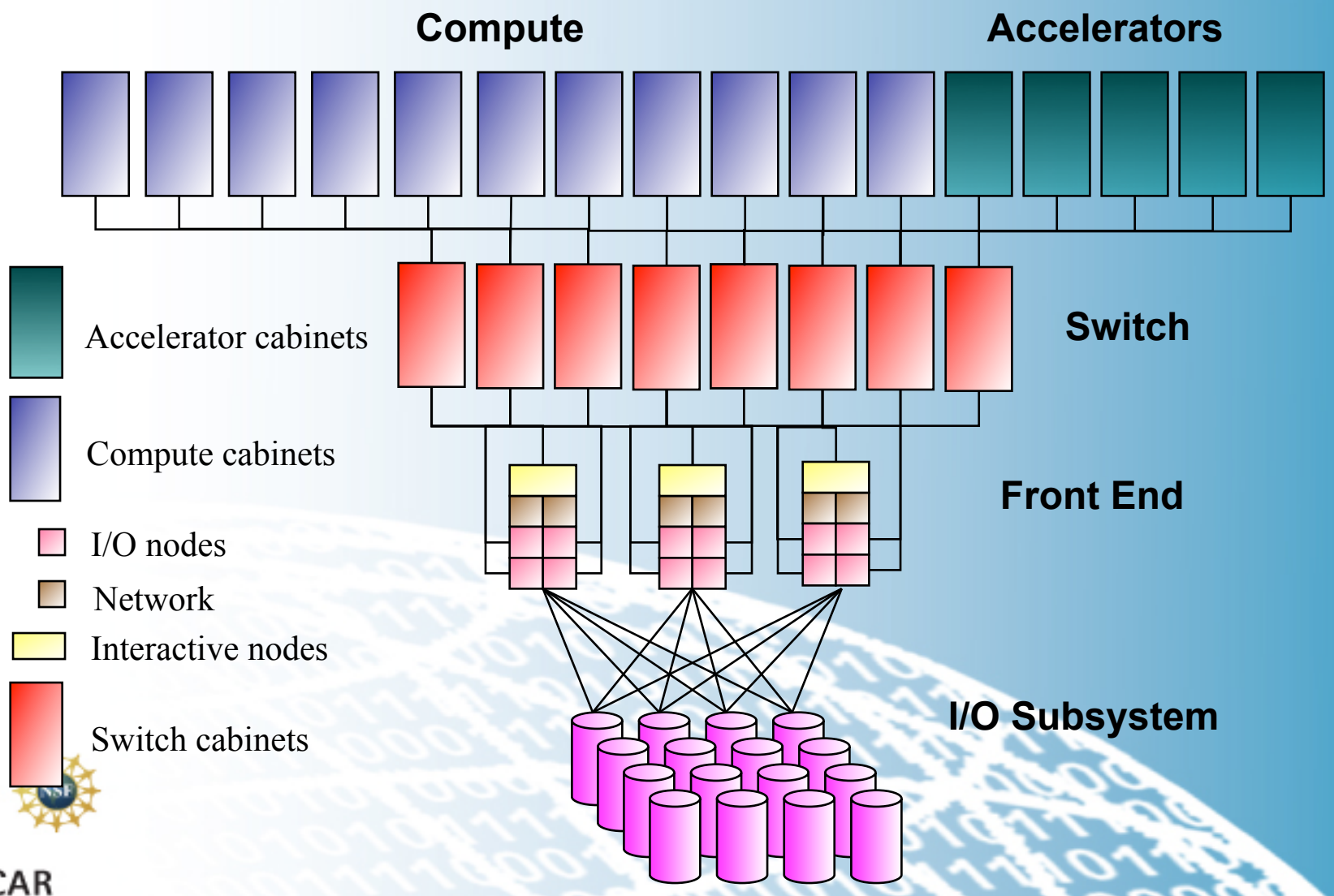
Cray-2/8 1986
3.9 GFLOPS

Blue Gene/L 2006
130K CPU's
367 TFLOPS

The history of parallelism in supercomputing…



Generic Petascale Computer Architecture



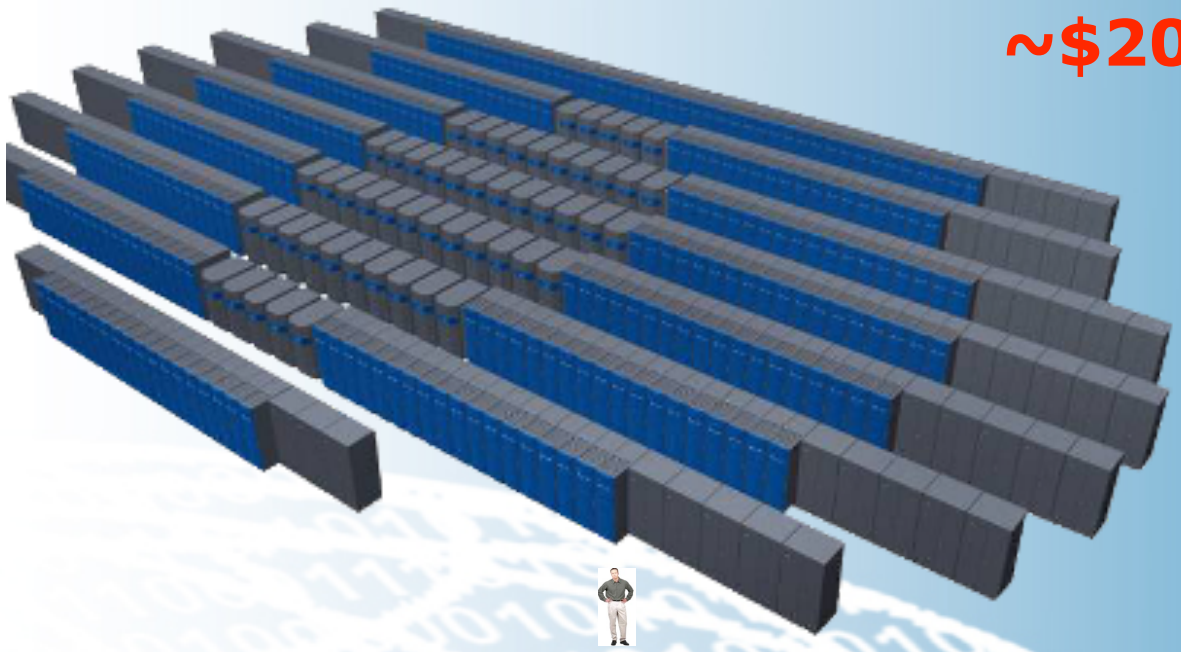
Petascale System Design Issues: Performance Means Heat

- However, achievable performance has been increasingly gated by the memory hierarchy performance not CPU peak
  - Peak is basically a poor predictor of application performance
- Aggregate memory bandwidth =
  - Signaling rate/pin x pins/socket x sockets
- To increase aggregate bandwidth you can increase
  - signaling rate - fundamental technology issue
  - pins/socket - packaging technology
  - sockets - more communications
- Consequences
  - More heat
  - Higher heat density
  - More heat from the interconnect
- System power requirements
  - Track-1 O(10 MW)
  - Mid next-decade exascale system- O(100 MW)

A Petaflops Sustained System in 2011… will be big by any measure

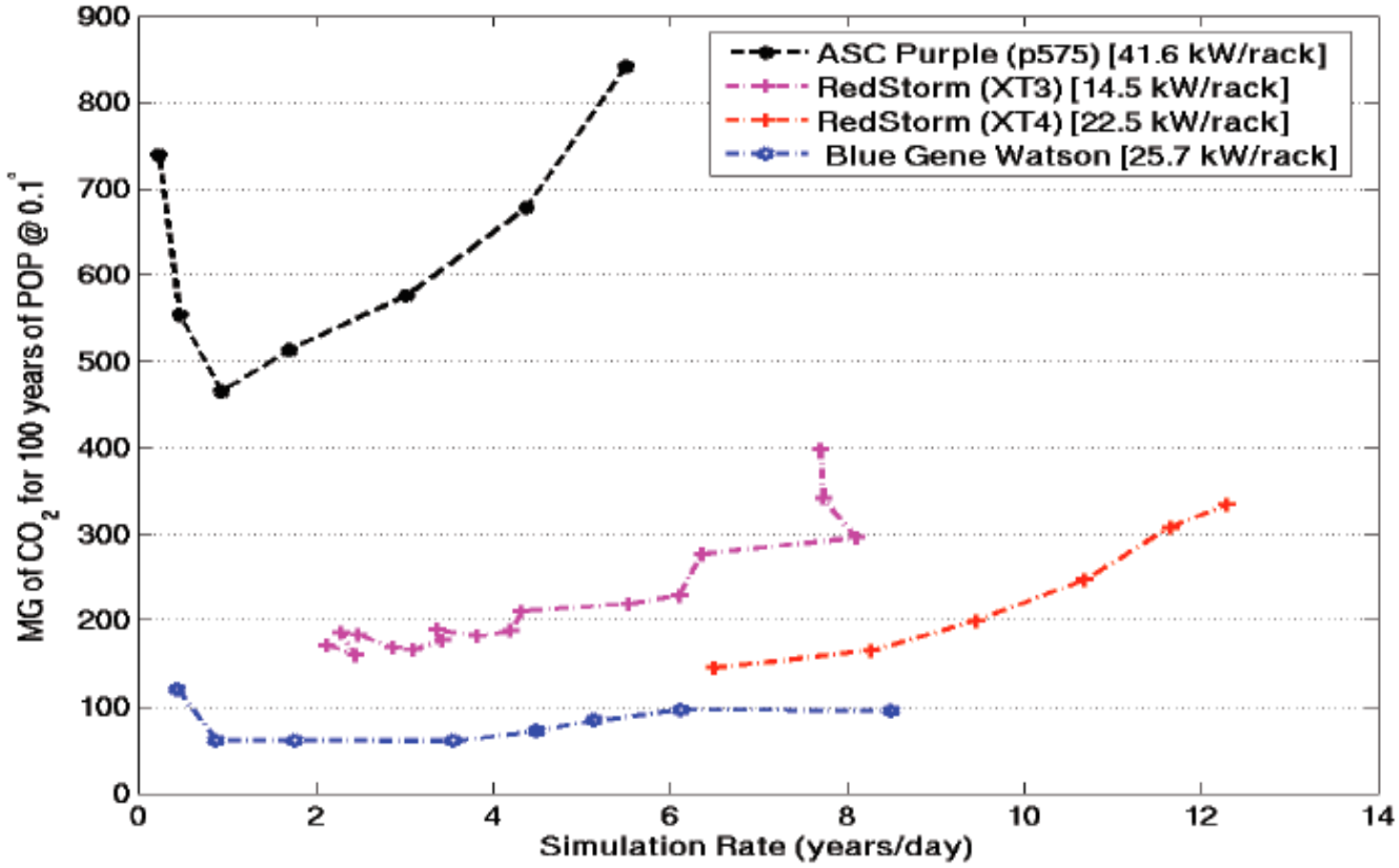
10-20 MW

~$200 M



O(10^5 - 10^6) CPU's

Not all systems have the same carbon footprint

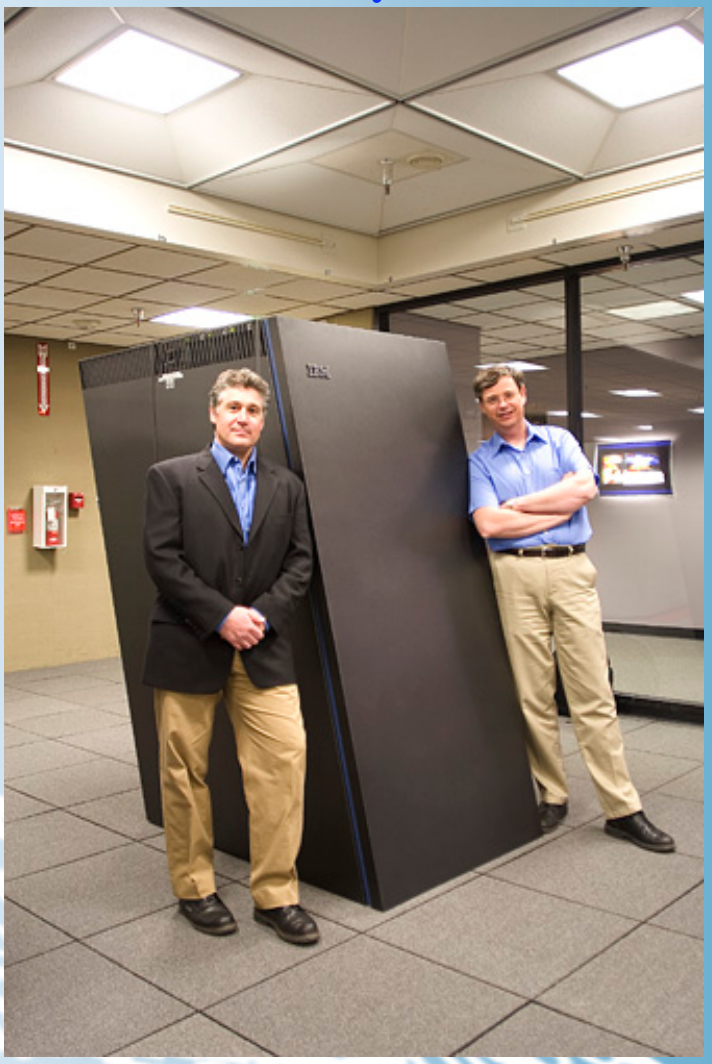


Getting applications to scale is green!

NCAR and University Colorado Partner to Experiment with Blue Gene/L

Characteristics:

- 2048 Processors/5.7 TF
- PPC 440 (750 MHz)
- Two processors/node
- 512 MB memory per node
- 6 TB file system



Dr. Henry Tufo and myself with "frost" (2005)

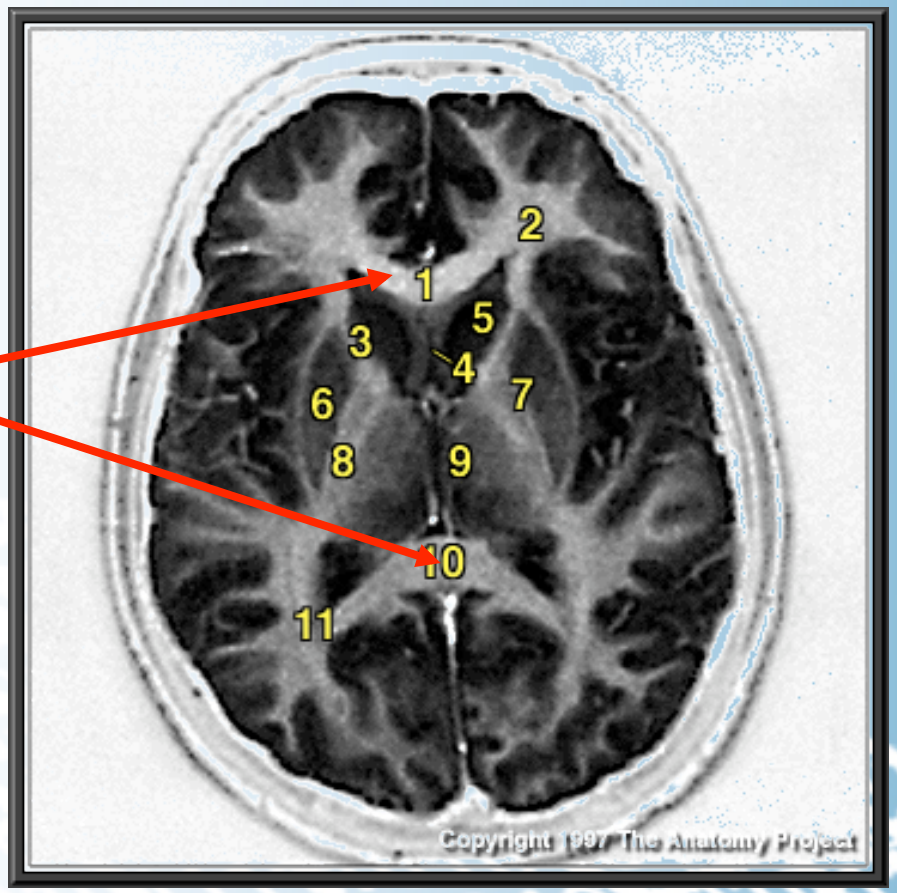
The Brain as a Computer (I)…

The brain as a computer

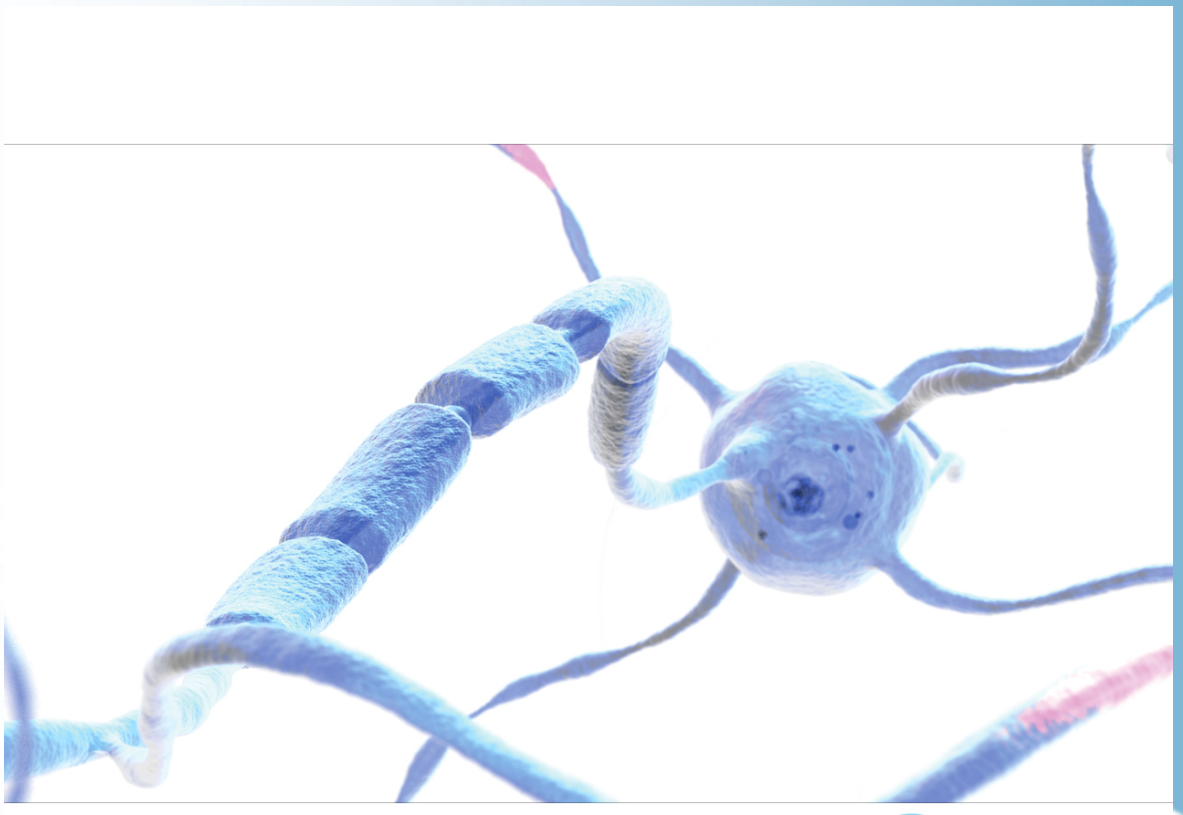
- 1 m^2 compute surface (Cerebral Cortex) with 10^{10}-10^{12} neurons
- Fault Tolerant/Denying!
- Power consumption
  - overall 25 W
  - compute 10 W
- Estimated raw compute: 10^{13}-10^{15} SyOPs
- 50% White Matter - myelin sheathed axons
  - Ranvier nodes - act as signal repeaters and increase propagation speeds 100x
  - Signals jump node to node
- Half the brain is white matter
- Effectively, half the brain is interconnect

The Brain's Interconnect

Corpus callosum

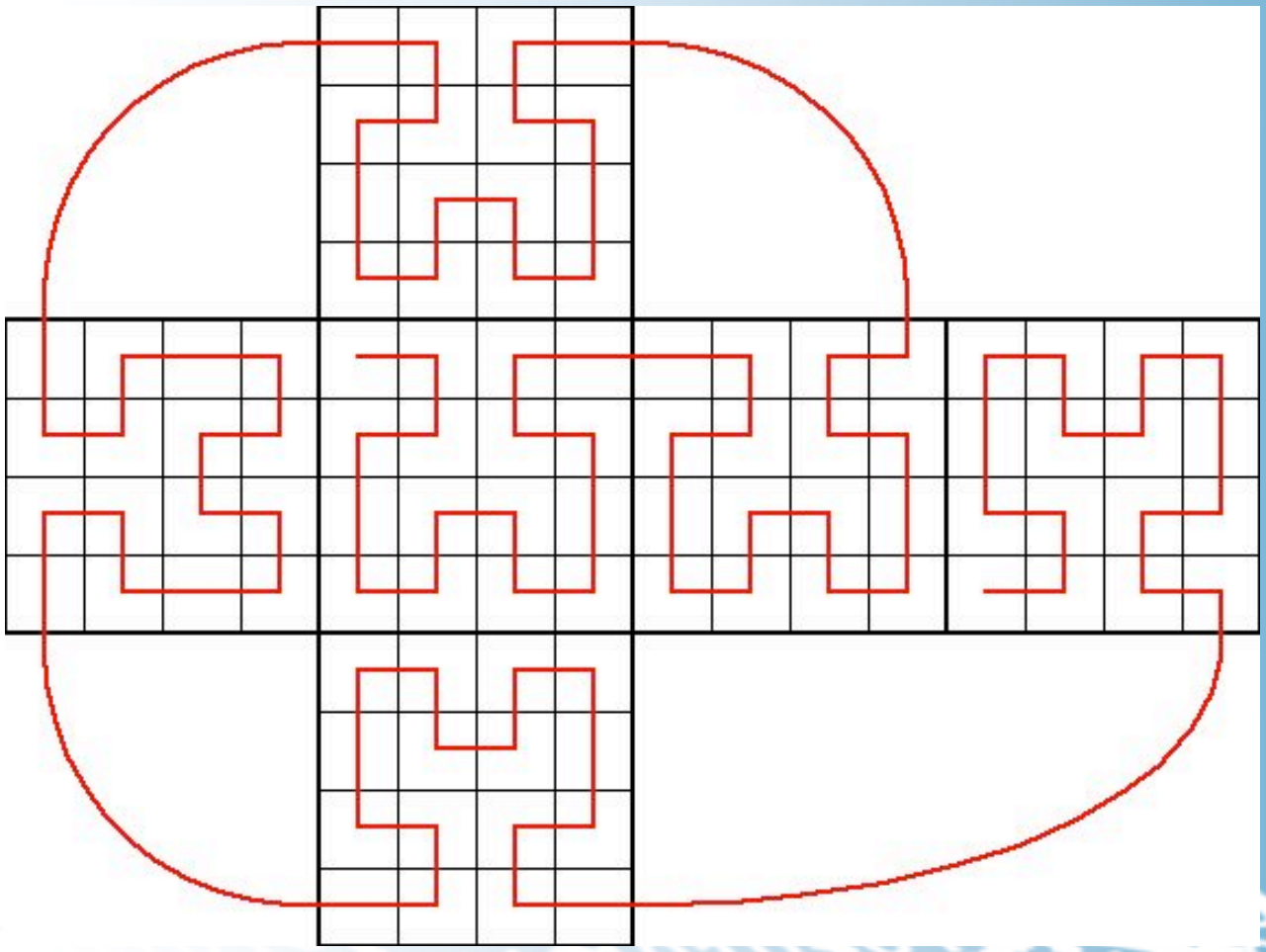


The brain's solution for long-haul signal propagation: Nodes of Ranvier

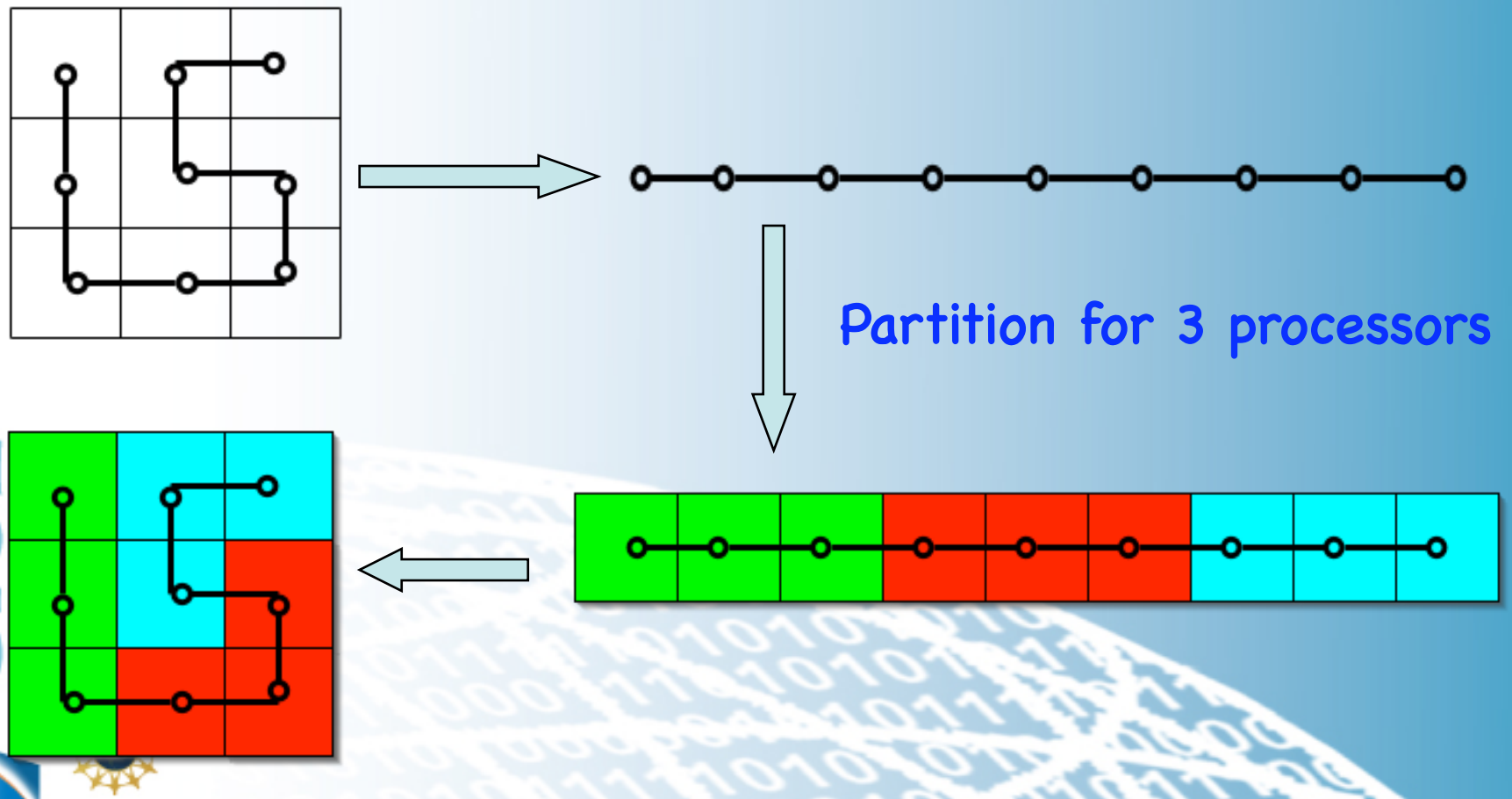


The Importance of Locality…

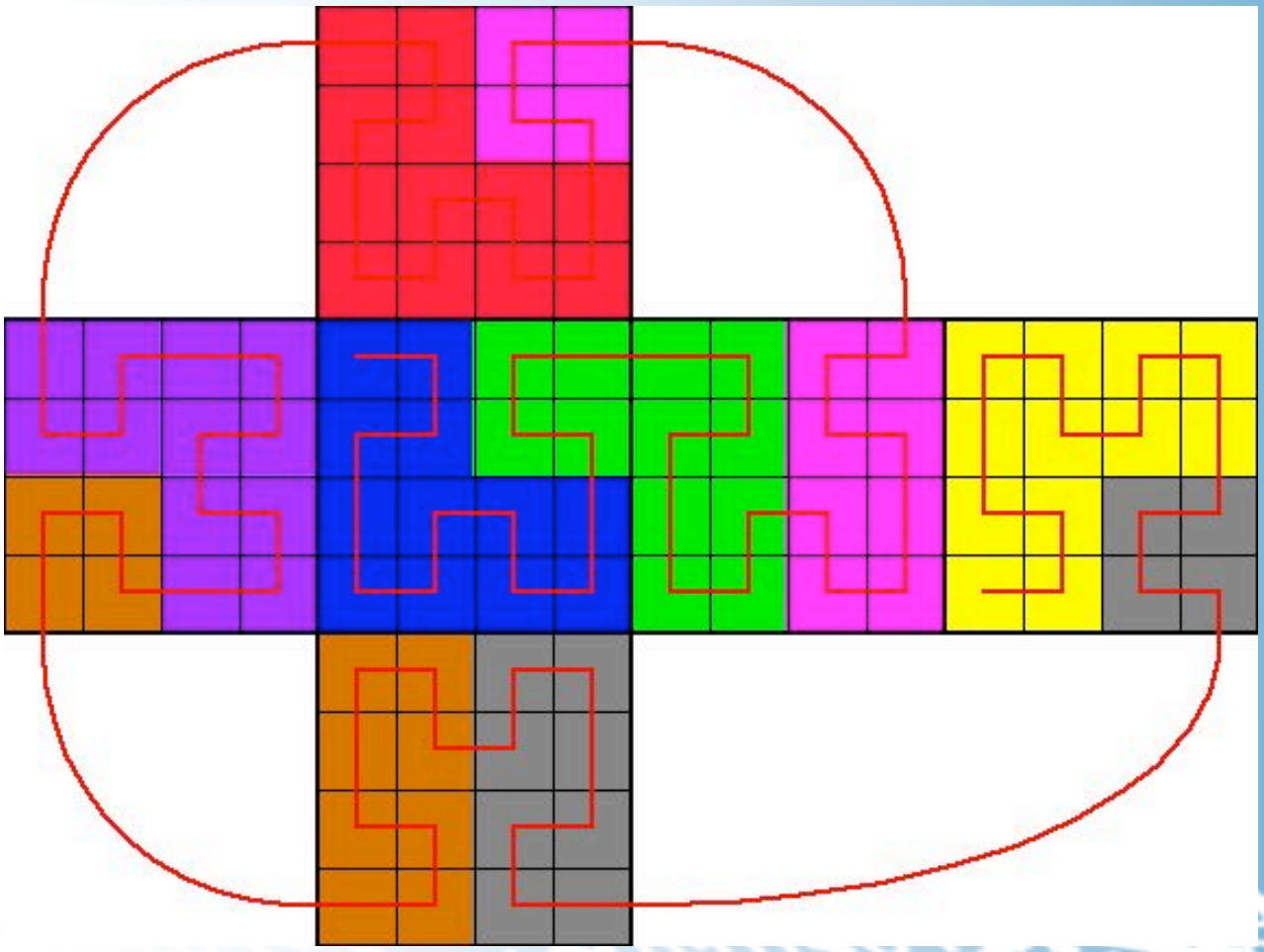
Partitioning a cube-sphere on 8 processors



Load Balancing: Partitioning with Space Filling Curves



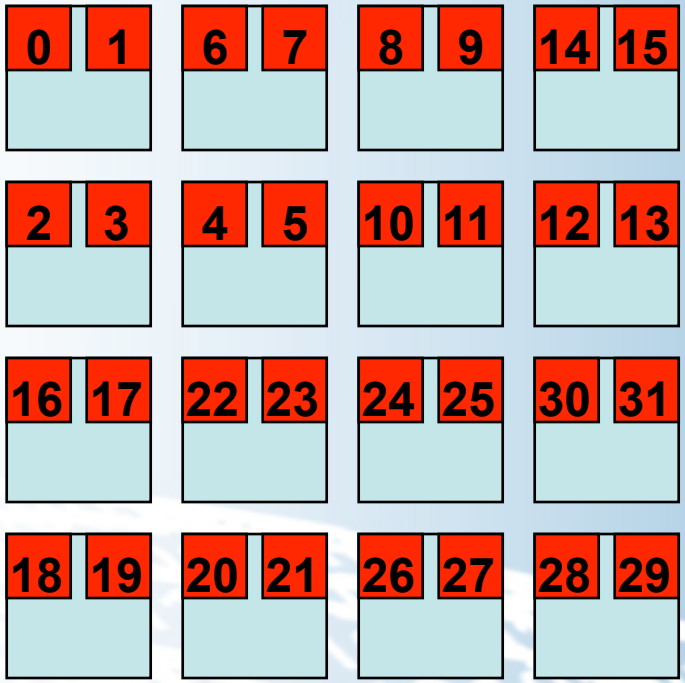
Partitioning a cubed-sphere on 8 processors



Mapping to SFC's to Torus Network

- Must map 1-D list of SFC domains to MPI processes on 3-D torus intelligently.
  - Need to maximize torus locality.
  - Need to minimize wire contention.
- Basic idea: snake through the torus as well.

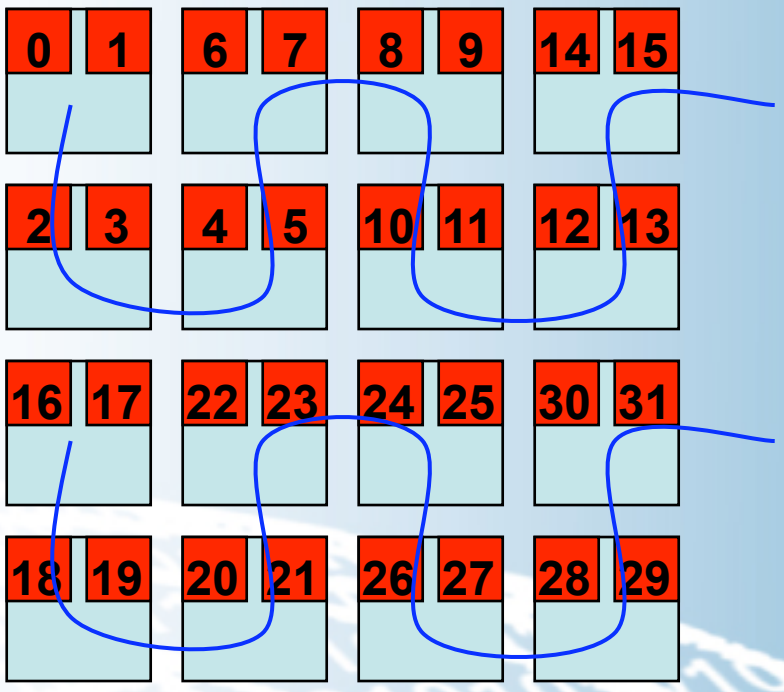
Mapping the SFC's to the IBM Blue Gene/L Torus: Even Better 2x2 "Snaked" Mapping



Virtual Node Mode

Shown in 2-D

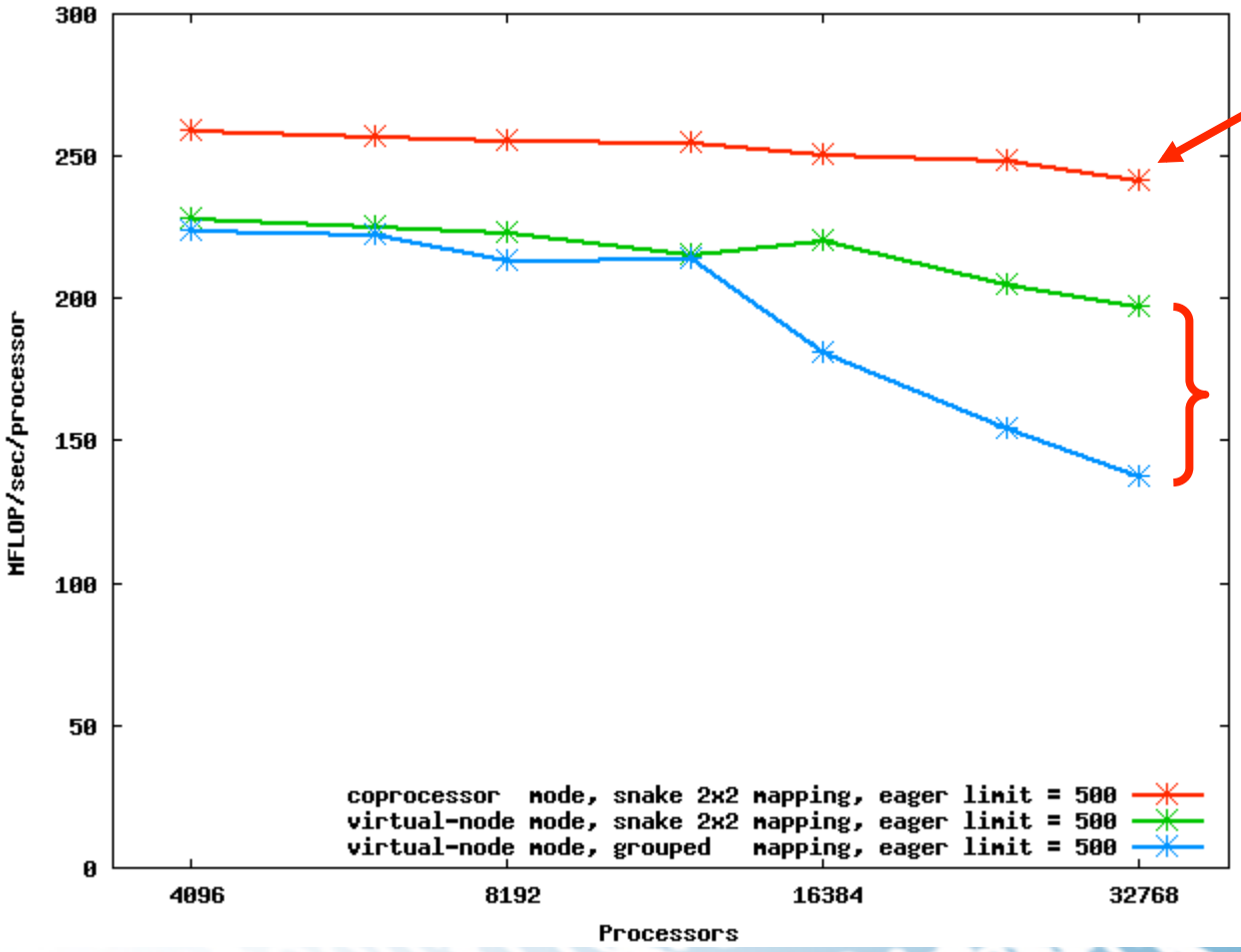
Mapping the SFC's to the IBM Blue Gene/L Torus: Even Better 2x2 "Snaked" Mapping



Virtual Node Mode

Shown in 2-D

HOMME/Held-Suarez Performance on Blue Gene/L



CPM gets 8 TFlops

VNM Improvement due to snaking

Sustained MFLOP per second per processor for moist Held-Suarez.
Explicit integration Dt = 4 seconds.
6 X 128 X 128 elements, 96 vertical levels.

Computational & Information Systems Laboratory

CISL

NCAR

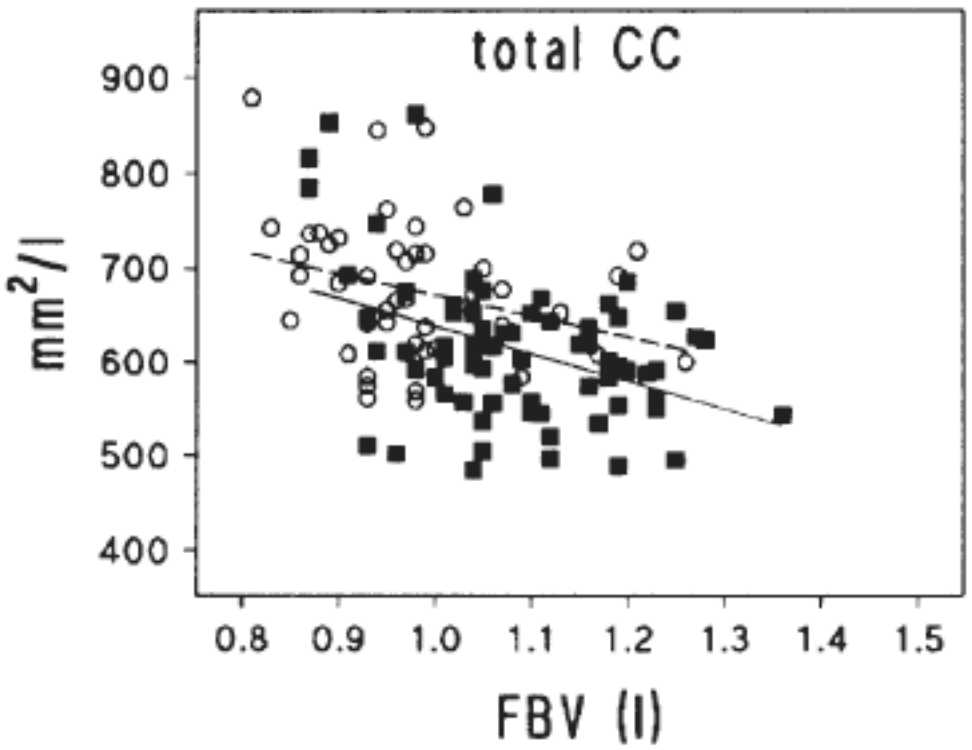
The Brain as a Computer (II)…

Locality optimization in brain design…

"…as brain size is scaled up there must be a fall in interhemispheric connectivity, due to the increasing time constraints of transcallosal conduction delay. Consequently, functionally related neuronal elements would cluster in one hemisphere, so that increasing brain size would be the driving force in the phylogeny of hemispheric specialization."

Jäncke, Staiger, Schlaug, et al., Cerebral Corex, 1997:7,48-56

Network vs compute trade-off in the human brain…



CISL Computational & Information Systems Laboratory

NCAR

Algorithms: the wise use of non-local communication…

The Role of Algorithms in Realizing the Future of Earth System Modeling

Modeling Goals

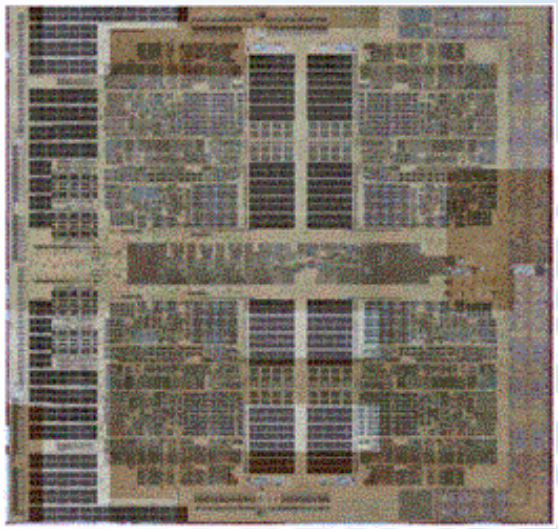
- Solve space scales
- Solve time scales
- Accuracy
- Efficient use of network resources
- Performance

Algorithms

- Adaptive Mesh refinement
- Implicit time-stepping: solvers
- High order methods
- Domain Decomposition
- Process Mapping
- Scalable methods

A Thought Experiment

- The road we're on says we'll get:
  - 2x CPU's every 18 months
  - But stagnant thread speed
- Suppose these idealized conditions exist:
  - Perfectly scalable system
  - Its infinite extensibility (for a price)

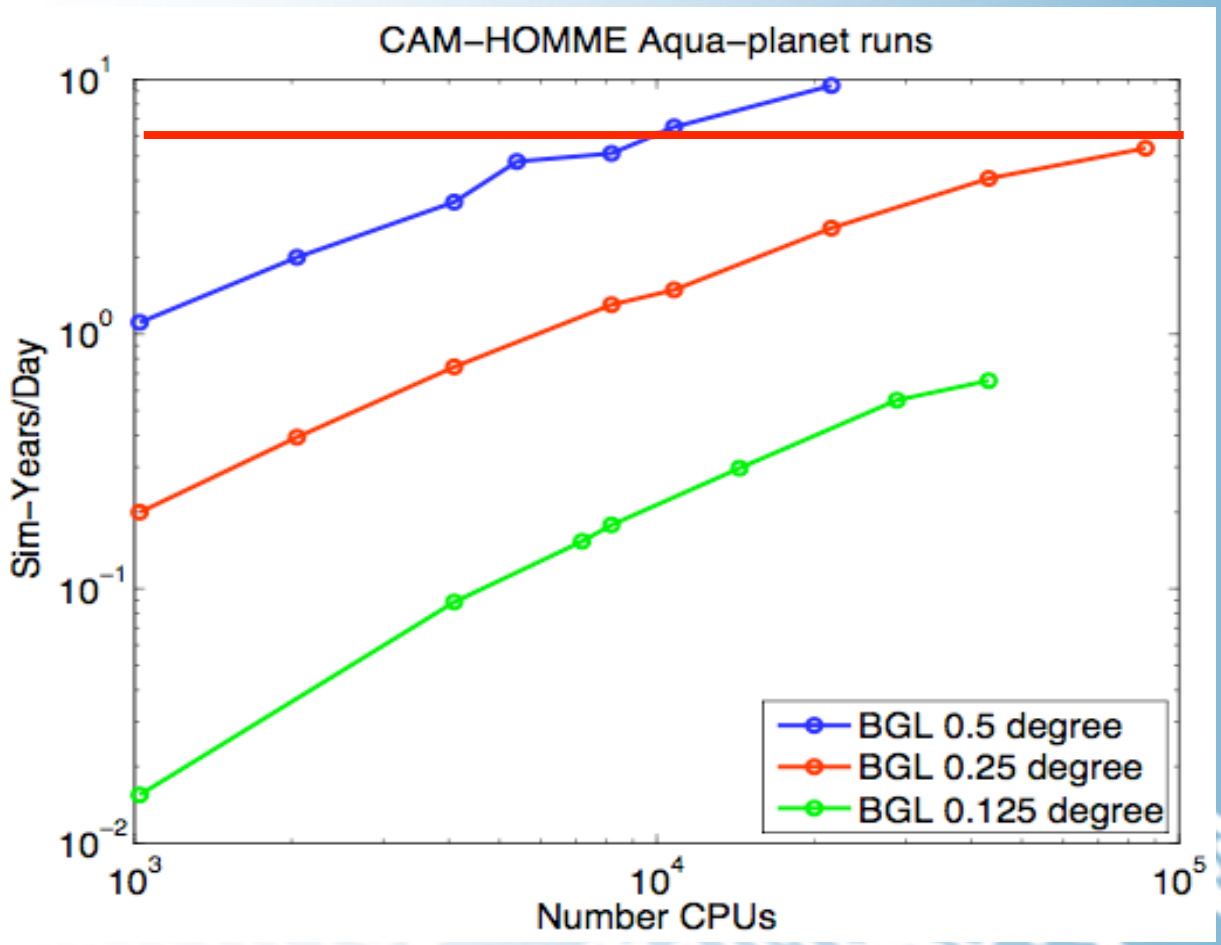


Merciless Effects of CFL

- Dynamics timestep goes like N^{-1}
  - The cost of dynamics relative to physics increases as N
  - e.g. if dynamics takes 20% at 25 km it will take 86% of the time at 1 km
- Option 1: Look at Algorithmic Acceleration
  - Semi-Lagrangian Transport
    - cannot ignore CFL with impunity
    - Increasingly non-local and dynamic communication patterns
  - Implicit or semi-implicit time integration - solvers
    - Non-local/quasi-local communications
  - Adaptive methods
- Option 2: Faster threads - find more parallelism in code
  - Architecture - old tricks, new tricks… magic tricks
    - Vector units, GPU's, FPGA's
  - device innovations (high-K)

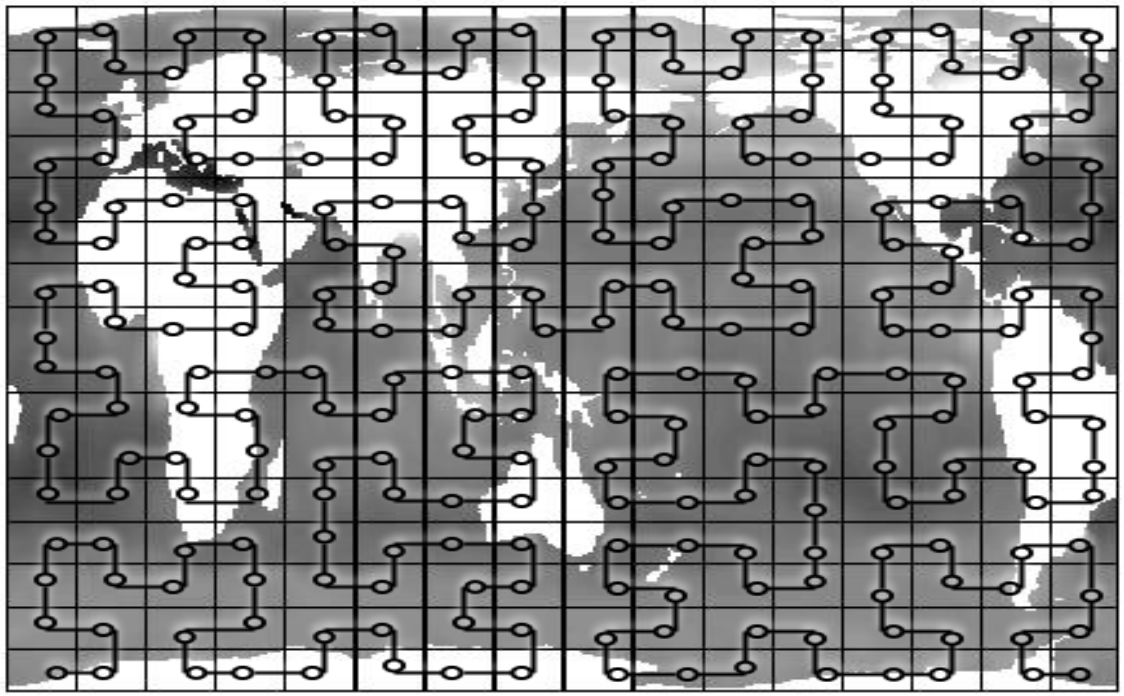
Example: Aqua-Planet Experiment with CAM/HOMME Dycore

Integration Rate Drops of as Resolution Increases



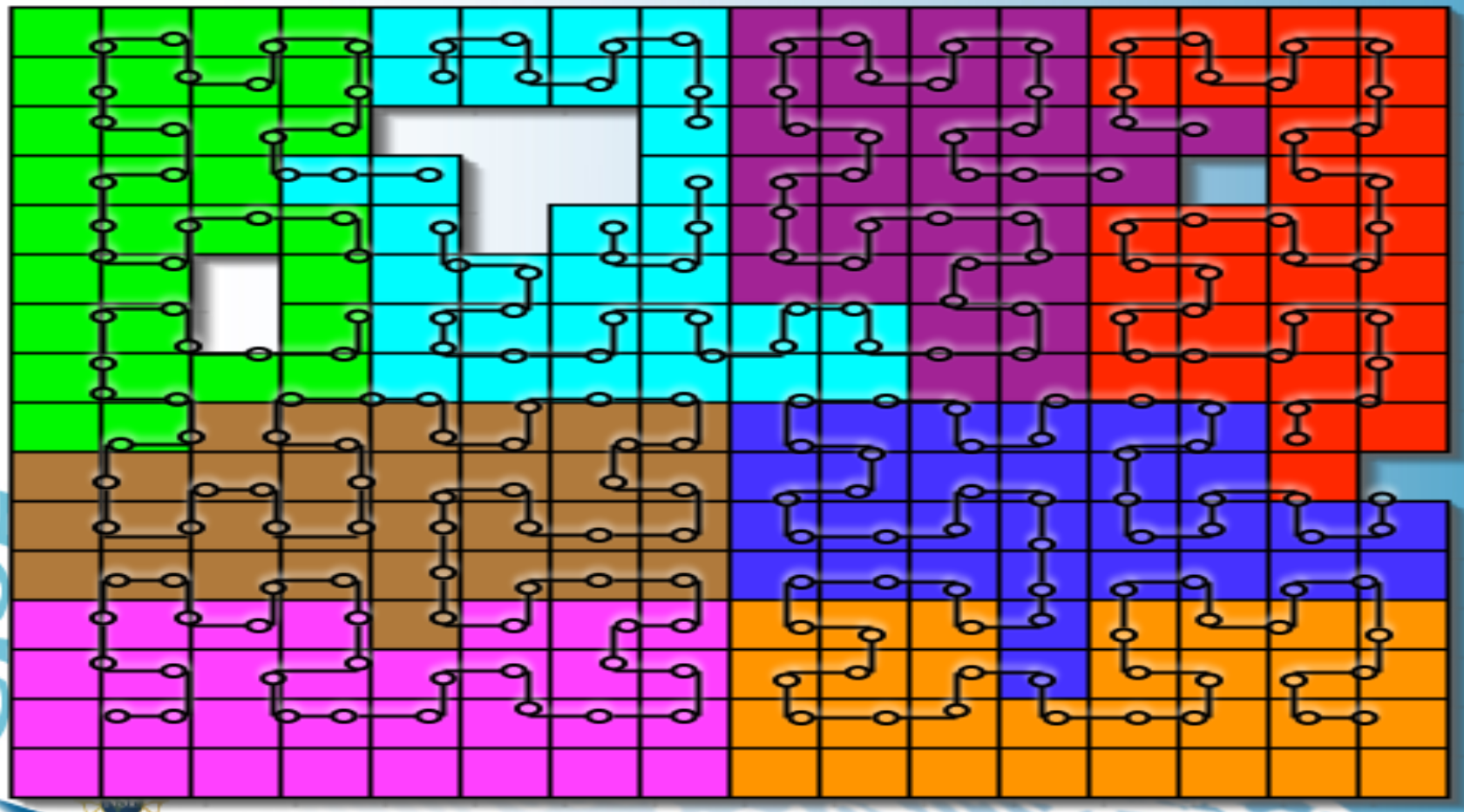
5 years/day

POP (gx1v3) with space-filling curves(Hilbert Nb=2^4)



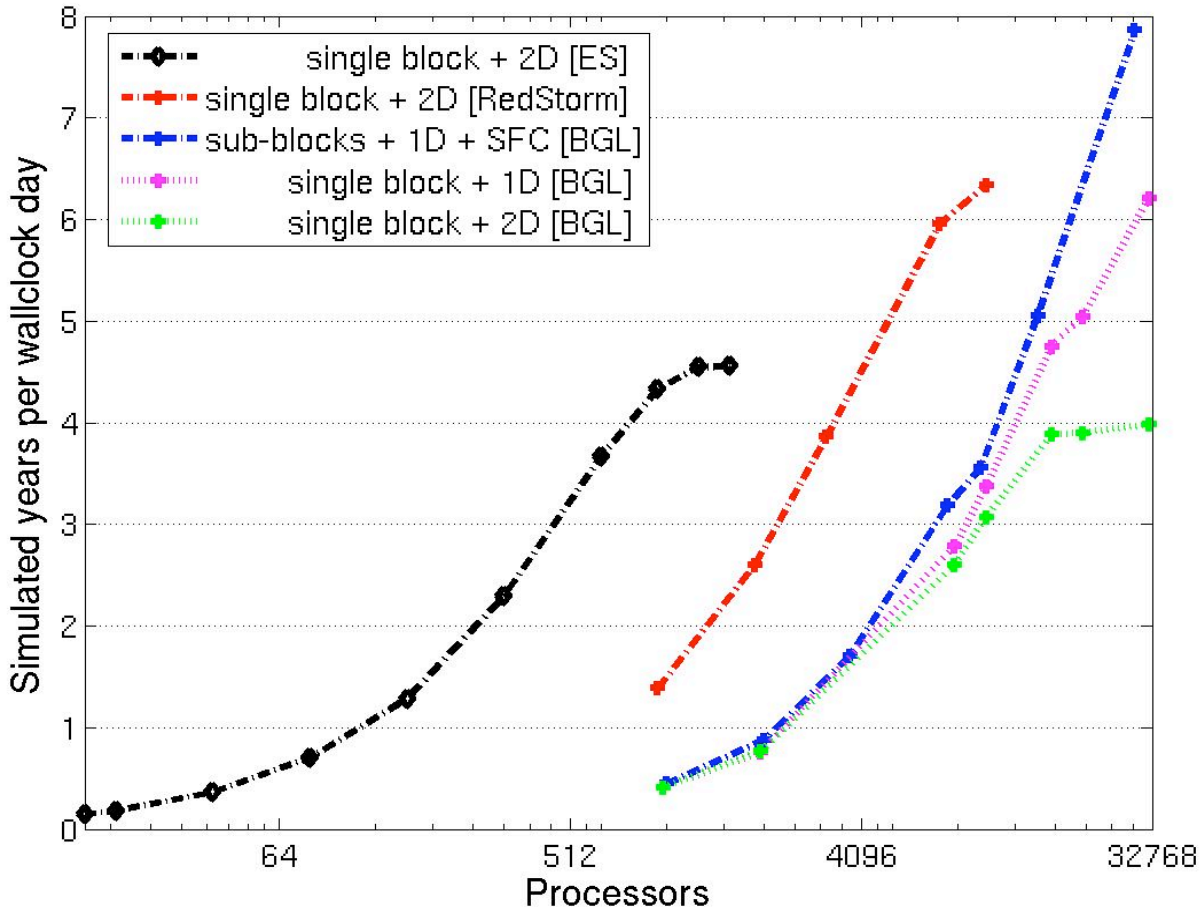
Space-filling Curve Partitioning for 8 Processors

Static Load Balancing…



Key concept: no need to compute over land!

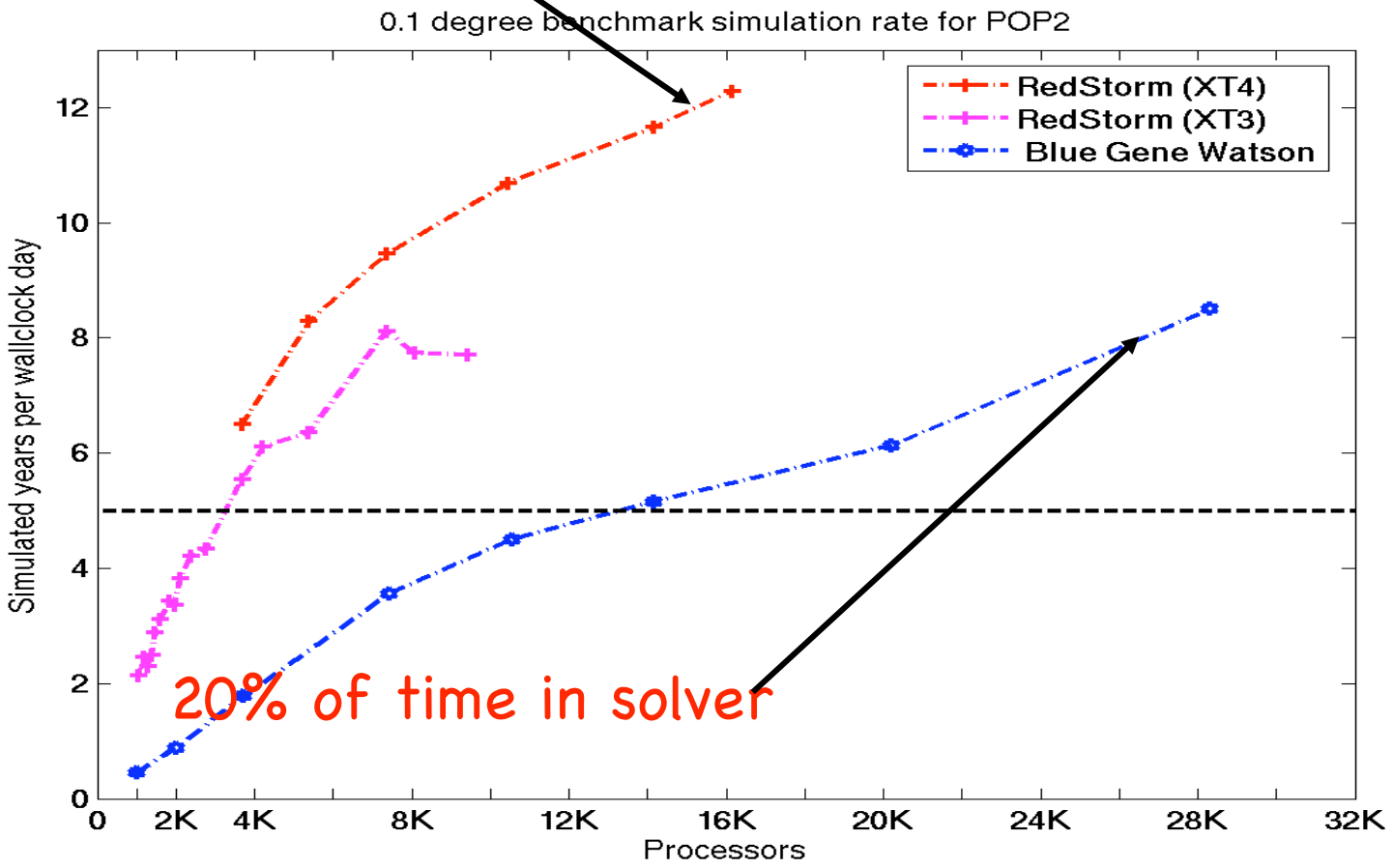
POP 1/10 Degree performance



Key concept: You need routine access to > 1k procs to discover true scaling behaviour!

Improved Scalability of the POP-2 0.1° benchmark

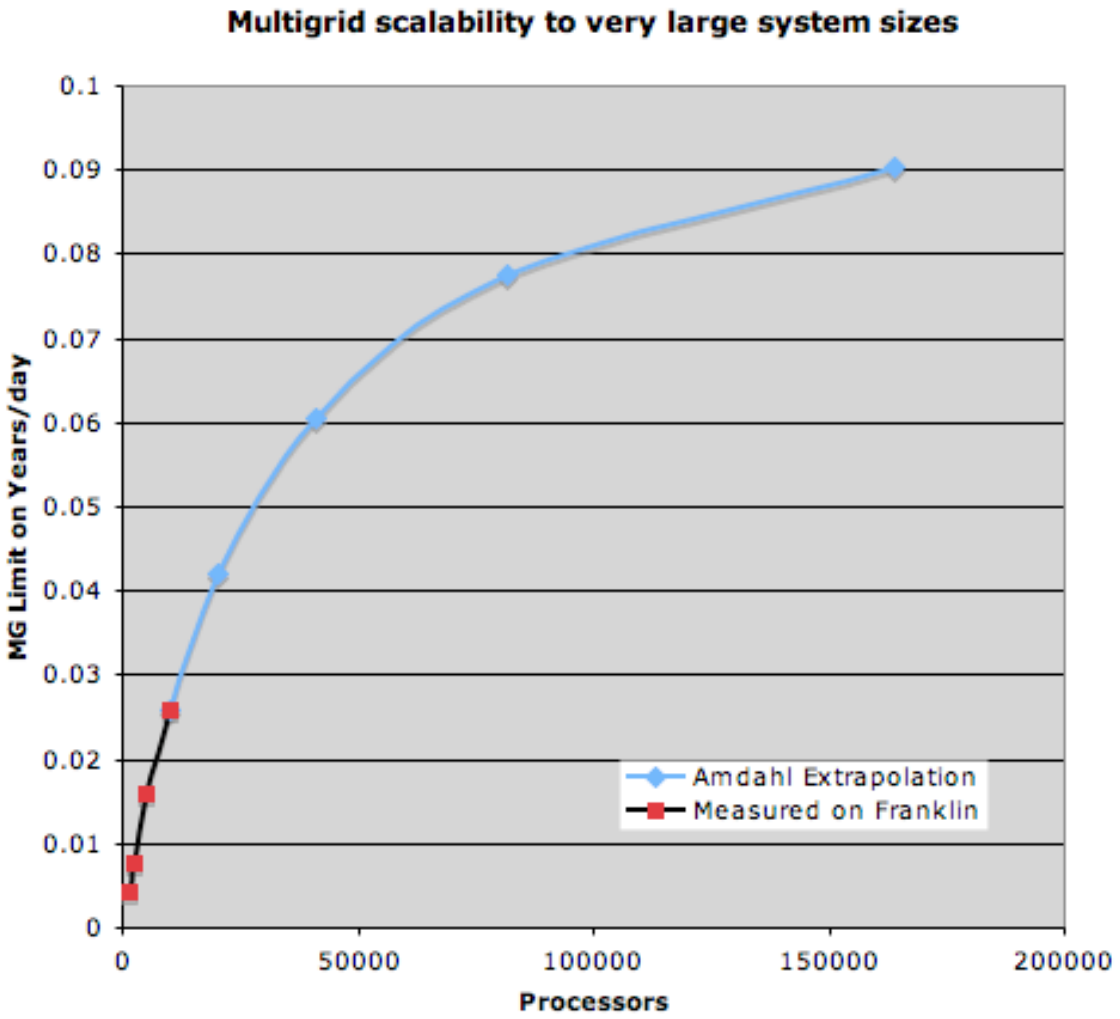
71% of time in solver



20% of time in solver

Key concept: different architectures have different scalability properties.

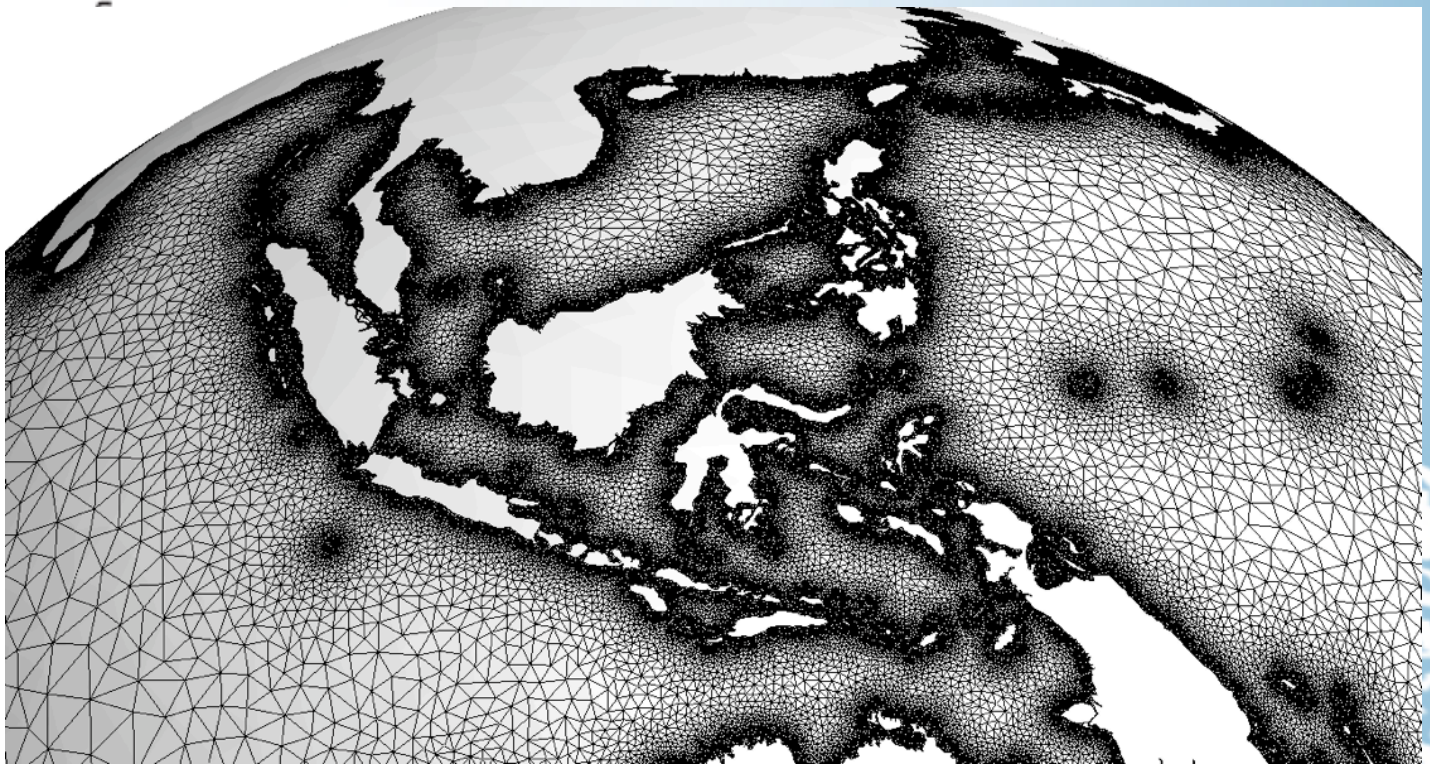
Applied Math vs Amdahl's Law- Could Solver Scalability Also Limit Integration Rate?



Adaptive Mesh Refinement

- SLIM - Louvain la Neuve University
- DG, implicit, AMR unstructured

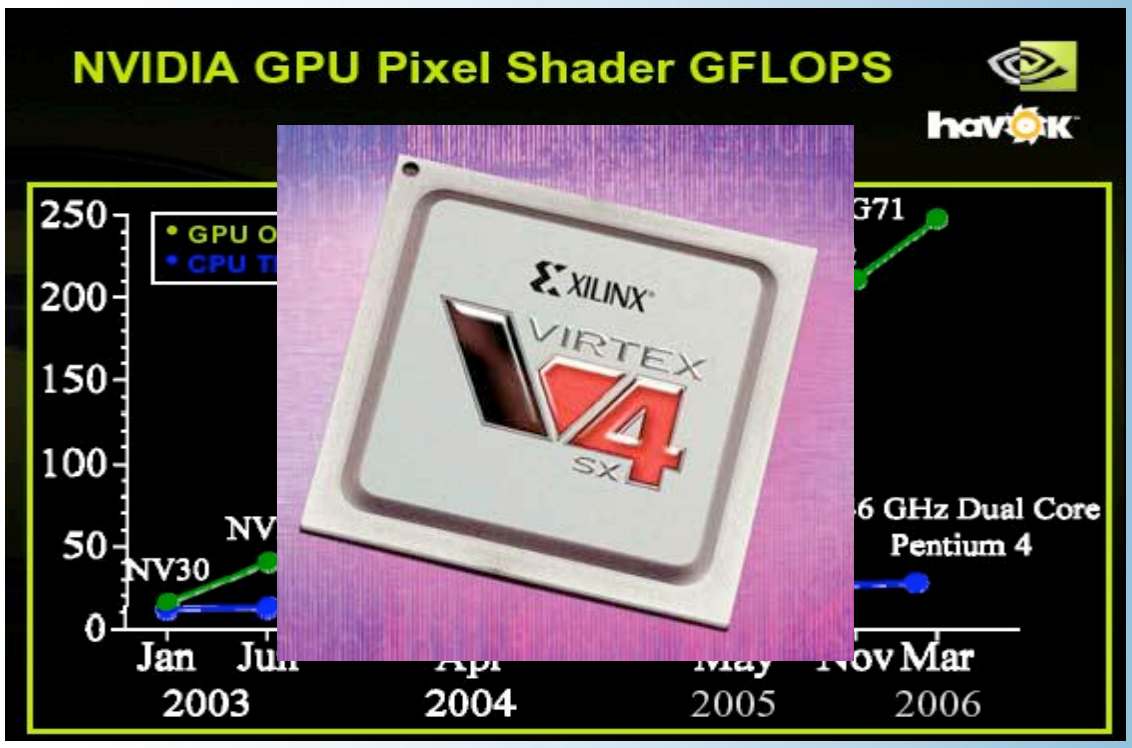
To be coupled to prototype unstructured ATM model



(Courtesy of Amik St. Cyr and J-F Remacle LNU)

Using Accelerators…

Leveraging the architectural paradigm shift?



- IBM Cell Processor - 8 cores
- Intel "concept chip" 1 TFLOPS 80 cores/socket
- Paradigm shift?
  - GP-GPU - 128 graphics pipes
    - Measured 20x on WRF microphysics
  - FPGA (data flow model)
    - Simulated 21.7x on Xilinx V5 CAM sw-radiation code.

Architecture is Important (Again)!

- Improvements in clock rates trumped architecture for 15 years
- Clock rates stall out -> architecture is back
- Accelerator space is wide open and poised for rapid increases in performance
- How do we exploit this?

Computational Intensity (CI)

- Compute Intensity:

  CI = Total Operations/(Input + Output data)
- GFLOPS = CI*Bandwidth
- Bandwidth expensive, flops cheap
- The higher the CI, the better we're able to exploit this state of affairs

Computational Intensity: Examples

- Saxpy: C= aX[]+B[], a = scalar, X, B vectors
  - CI = 1/3
- Matrix-Vector Multiply (N large)
  - CI = (2*N-1)*N/(N*(N+2)) ~ 2
- Radix 2 FFT -
  - CI = (5*log2(N)*N)/(2*N) = 2.5*log2(N)
  - 6.6 GFLOPS (low compute intensity)
- NxN - Matrix Multiply
  - CI = $(2*N^2-1)*N/(3*N*N) \sim 2*N/3$
  - 167 GFLOPS nVidia (high compute intensity)

Here Come the Accelerators: GPUs

- GPUs
  - SIMD find-grained parallelism
  - Also multi-level concurrency
  - Very fast, peak 520 GF/s
  - Cheap (< $500) commodity plug-in coprocessor for ordinary desktop systems
  - Programmability? Better tools on the way
- Approach used in WRF NWP Model
  - Incremental adoption of acceleration, module by module
  - Cloud microphysics (WSM5 testbed)
    - 25% of run time, < 1% of lines of code
    - 10x boost in microphysics
    - 20% increase in App performance overall versus high-end AMD opteron
  - Ongoing, adapt more of code to GPU

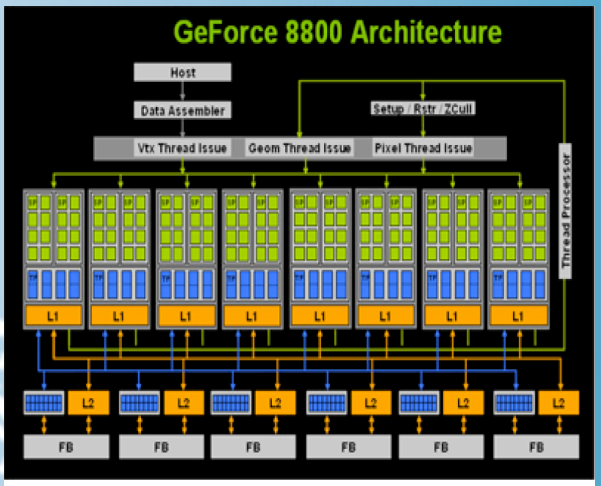
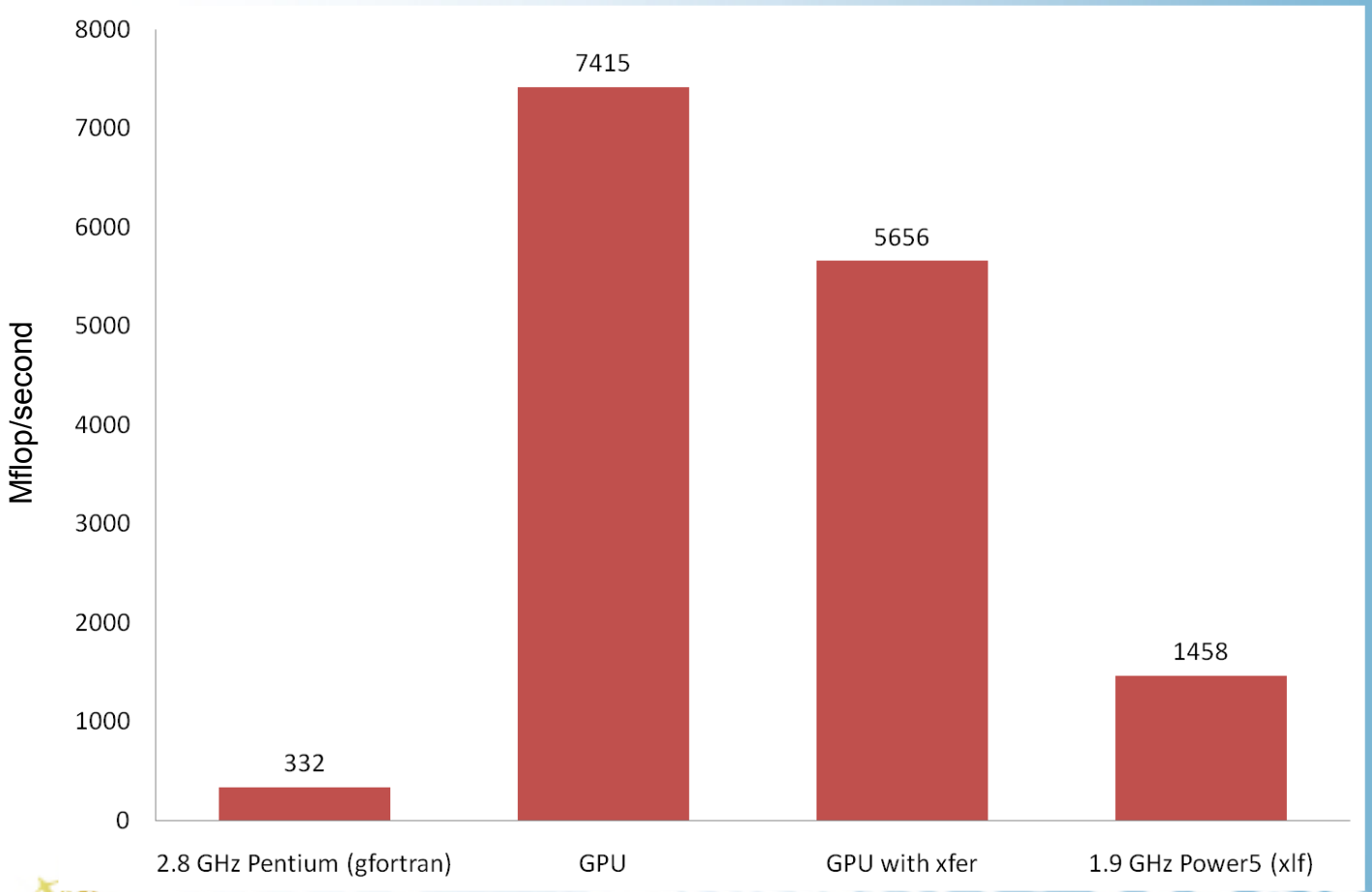


Figure 1. GeForce 8800 GTX block diagram

Here Come the Accelerators: WSM5 Kernel Performance

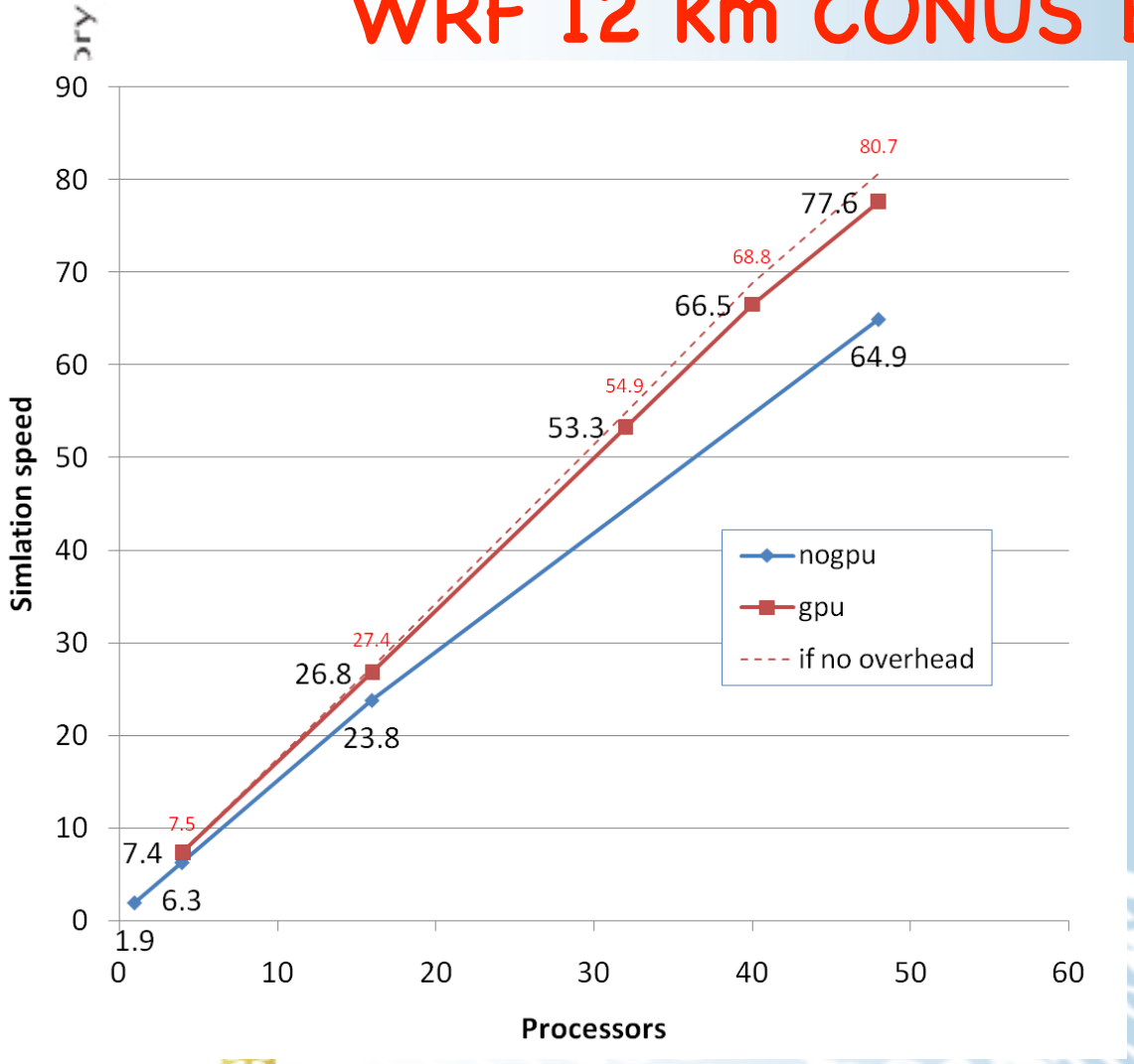
- Stand-alone microphysics testbed
- Workload: Eastern U.S. "Storm of Century" case
  - 74 x 61 (4500) threads
  - 28 cells/column
  - ~300 Mflop/invocation
  - 5 MB footprint
  - Moving 2 MB host<-> GPU in 15 milliseconds (130MB/sec)

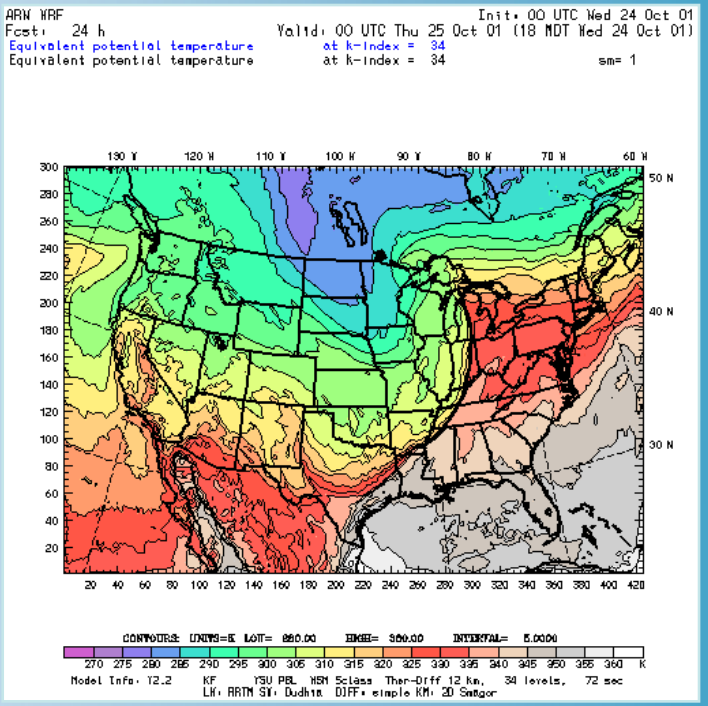
Here Come the Accelerators: WSM5 Kernel Performance



50 MFLOPS/W 38 MFLOPS/W 20 MFLOPS/W

Here Come the Accelerators: WRF 12 km CONUS Benchmark





qp.ncsa.uiuc.edu
16 Dual dual-core 2.4 GHz Opteron nodes, each with Four NVIDIA 5600 GTX GPUs

Credit: Wen-mei Hwu, John Stone, and Jeremy Enos

Courtesy of John Michalakes

NCAR

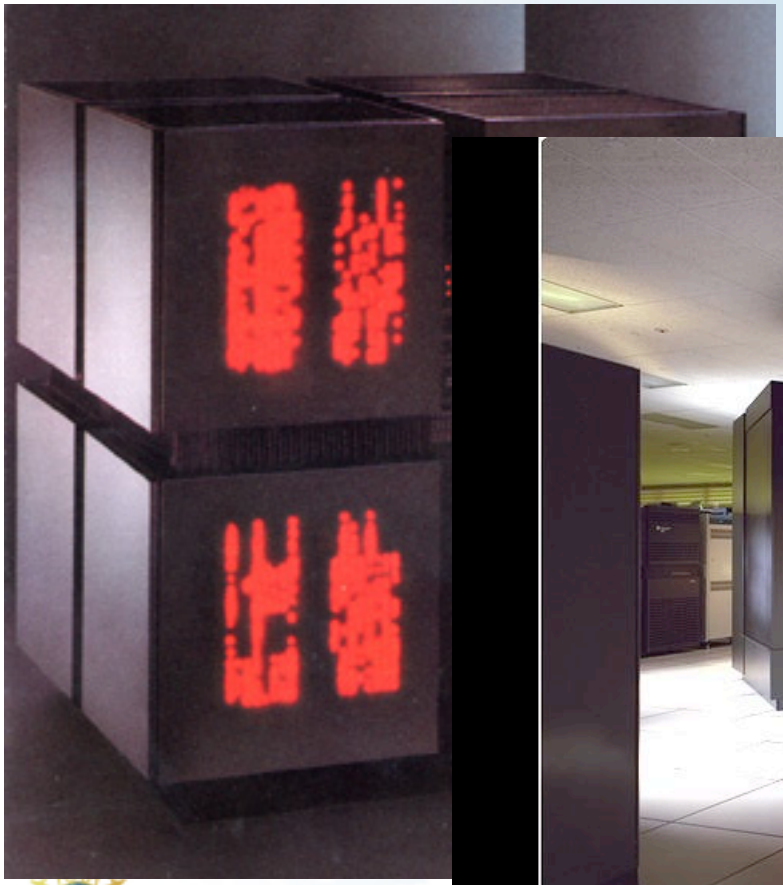
Some additional thoughts about accelerators…

- These GPU results are interesting and encouraging, but not yet compelling.
- Increasing importance of multiscale modeling may map well to the accelerator trend.

What we need to facilitate migration to accelerators…

- Got CI? => accelerate, but…
- Need robust hardware
  - Error trapping, IEEE compliance
  - Performance counters
  - Circuitry support for synchronization
- Need a programming model for these things
  - CUDA? Brook+?
  - Pragmas? Language extensions?
    - Begin/end define region
    - Data management: local allocation, data transfer support
- Need Robust Compilers
  - Automate computer intensity/profitability analysis.
  - Provide feedback about it to user.

This Harkens back to the First Era of Massively Parallel Computing (1986-1994)



TMC CM-2

TMC CM-5

The Difference: This Time, the Accelerators are Commodity Hardware

- First 1 TFLOPS GPU is out (February, 2008)
- 11 million PS3 units shipped in 2007
- Attract teens to supercomputing?
- Leverage new sources of talent and new techniques?

Maybe this sounds crazy…


Thanks!
Any Questions?