Berkeley Lab

NERSC
NATIONAL ENERGY RESEARCH SCIENTIFIC COMPUTING CENTER

National Energy Research Scientific Computing Center (NERSC)

Meeting the Challenge of Massive Parallelism

John Shalf
NERSC Center Division, LBNL

Office of Science
U.S. DEPARTMENT OF ENERGY

GSPS
April 18, 2007

NERSC

Traditional Sources of Performance Improvement are Flat-Lining

- **New Constraints**
  - 15 years of *exponential* clock rate growth has ended
- **But Moore's Law continues!**
  - How do we use all of those transistors to keep performance increasing at historical rates?
  - Industry Response: #cores per chip doubles every 18 months *instead* of clock frequency!
- **Is multicore the correct response?**

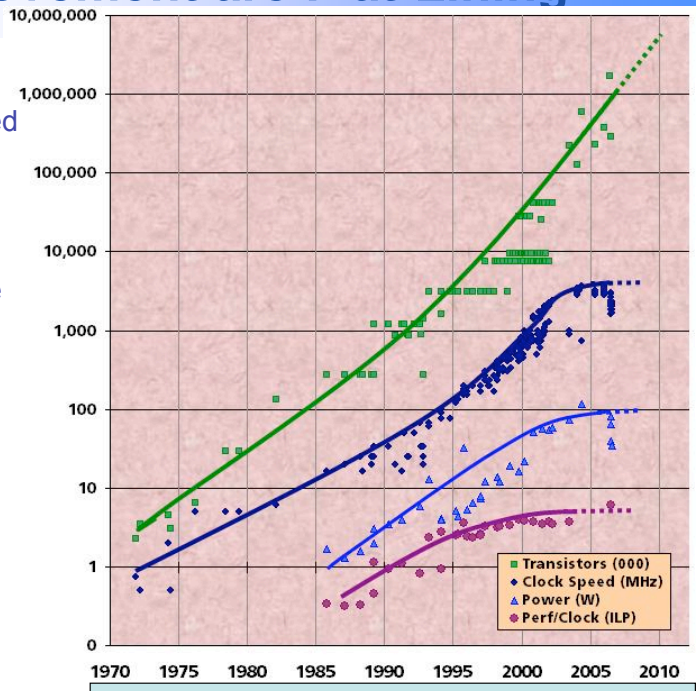


Figure courtesy of Kunle Olukotun, Lance Hammond, Herb Sutter, and Burton Smith

Office of Science
U.S. DEPARTMENT OF ENERGY

Is Multicore the Correct Response?

- Kurt Keutzer: **"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."**
- David Patterson: **"Industry has already thrown the hail-mary pass. . . But nobody is running yet."**
- Kathy Yelick: **"They are more confused then we are."**

Office of Science
U.S. DEPARTMENT OF ENERGY

Tension Between Commodity and Specialized Architecture

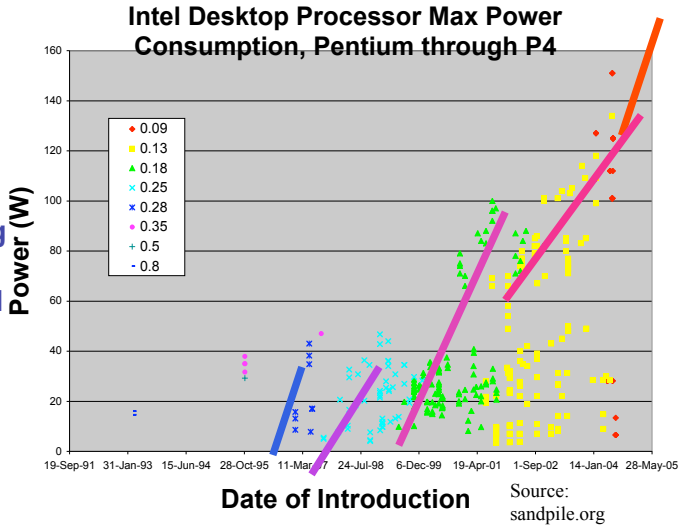
- **Commodity Components**
  - Amortize high development costs by sharing costs with high volume market
  - Accept lower computational efficiency for much lower capital equipment costs!
- **Specialization**
  - Specialize to application to improve computational efficiency.
  - Specialization used very successfully by embedded processor community
  - Not cost effective if volume is too low.
- **When cost of power or software development exceeds capital equipment costs**
  - Commodity clusters are optimizing wrong part of the cost model
  - Will need for higher computational efficiency drive more specialization?

Office of Science
U.S. DEPARTMENT OF ENERGY

Microprocessors: Up Against the Wall(s)

From Joe Gebis

- Microprocessors are hitting a power wall
  - Higher clock rates and greater leakage increasing power consumption
- Reaching the limits of what non-heroic heat solutions can handle
- Newer technology becoming more difficult to produce, removing the previous trend of "free" power improvement

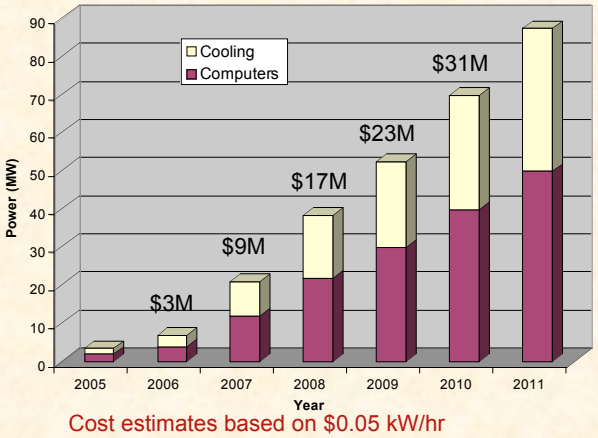


Source: sandpile.org

Office of Science
U.S. DEPARTMENT OF ENERGY

ORNL Computing Power and Cooling 2006 - 2011

- Immediate need to add 8 MW to prepare for 2007 installs of new systems
- NLCF petascale system could require an additional 10 MW by 2008
- Need total of 40-50 MW for projected systems by 2011
- Numbers just for computers: add 75% for cooling
- Cooling will require 12,000 – 15,000 tons of chiller capacity



Cost estimates based on $0.05 kW/hr

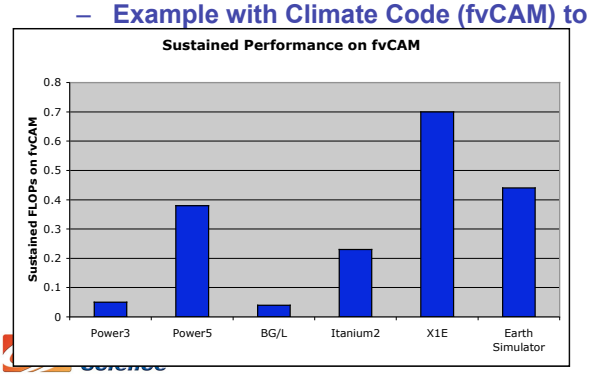
Annual Average Electrical Power Rates $/MWh

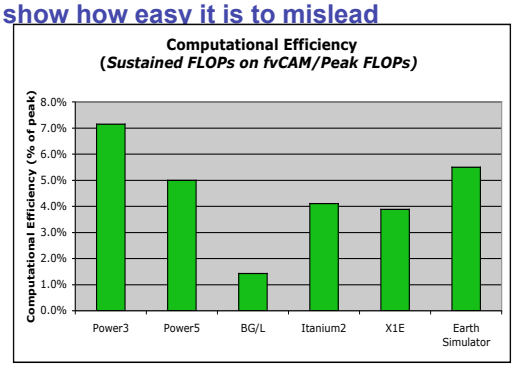
Site	FY 2005	FY 2006	FY 2007	FY 2008	FY 2009	FY 2010
LBNL	43.70	50.23	53.43	57.51	58.20	56.40 *
ANL	44.92	53.01				
ORNL	46.34	51.33				
PNNL	49.82	N/A				

Data taken from Energy Management System-4 (EMS4). EMS4 is the DOE corporate system for collecting energy information from the sites. EMS4 is a web-based system that collects energy consumption and cost information for all energy sources used at each DOE site. Information is entered into EMS4 by the site and reviewed at Headquarters for accuracy.

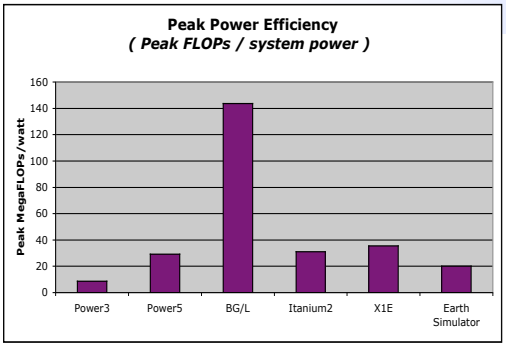
Power Efficiency vs. Power Consumption

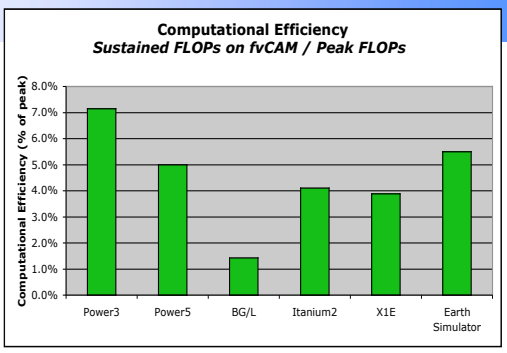
- Vendor Focus has been driven by Peak FLOPs/watt or reducing idle-power consumption using Dynamic Frequency/Voltage Scaling
  - Good for Consumer electronics which are idle most of the time
  - Marginal Benefit for HPC
    - Run ~100% loads
    - Time to solution is important
    - Effective/sustained performance is more important than peak
- Need a good metric for computational efficiency in order to influence industry
  - Example with Climate Code (fvCAM) to show how easy it is to mislead

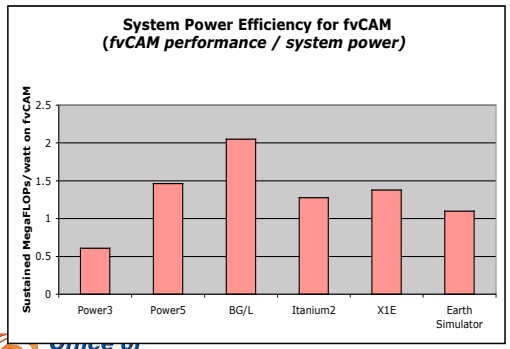


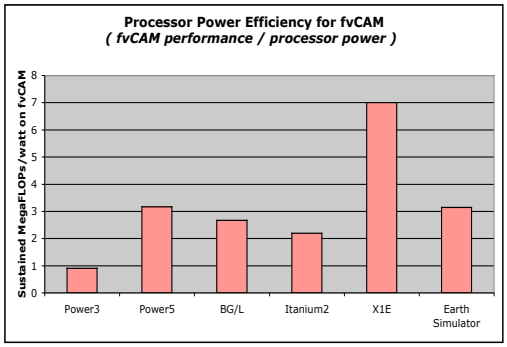


Power Efficiency running fvCAM



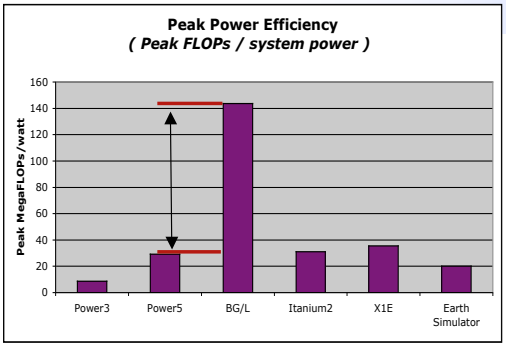


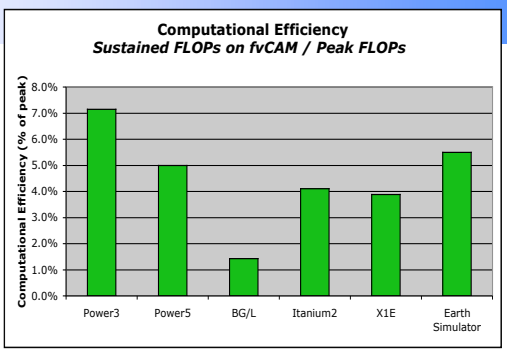


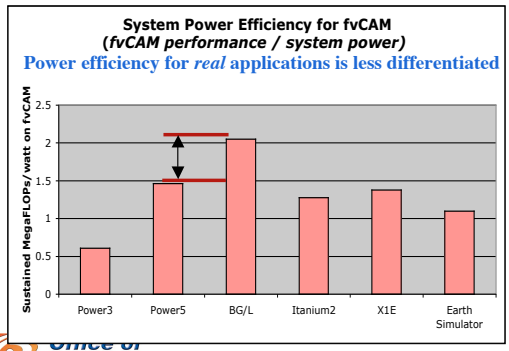


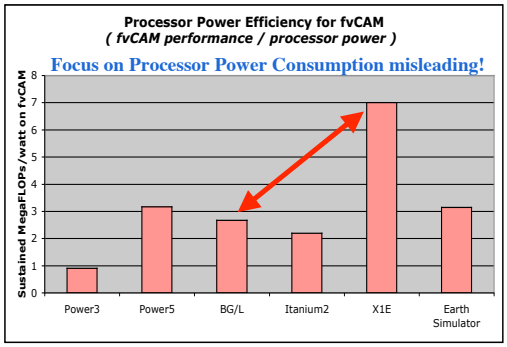
Benchmark results from Michael Wehner, Art Mirin, Patrick Worley, Leonid Oliker

Power Efficiency running fvCAM









Benchmark results from Michael Wehner, Art Mirin, Patrick Worley, Leonid Oliker

Office of Science
U.S. DEPARTMENT OF ENERGY

Power Efficiency

- **State of the Art**
  - Coarse Grained DVFS (slow down entire chip or core)
  - Clock gating
  - Ad-hoc environmental monitoring
- **Need innovations in**
  - Power efficiency metrics
  - Tight coupling of instrumentation, system HW, & software response/instrumentation (sensors and actuators)
  - Power aware algorithms
  - Joule Counters
  - PAPI-like analogue for collecting/unifying power/environmental monitoring data
  - Fine grained DVFS
    - must always slow down for something
    - New notion of system balance (unbalanced if slow down to wait for the same resource)

Office of Science
U.S. DEPARTMENT OF ENERGY

Ultimate Destination is Manycore

(what building blocks should we be leveraging from industry)

- **Convergence between HPC and Embedded Computing**
  - Technology from embedded market is now trickling up into server design rather than traditional trickle down flow of innovation. *(BlueGene and SiCortex)*
- **Convergence towards manycore**
  - hundreds of cores per chip (Cisco Metro, Intel TFLOPs, NVidia CUDA)
- **Effects on Computer architecture**
  - More/simpler cores per chip!
  - Lower degree interconnects
  - Constrained memory sizes (no longer 1 byte/flop)
  - Doubling of concurrency every 18 months
- **Effect on users**
  - How to ride a wave of exponentially increasing concurrency
  - As significant as migration from vector to MPP (early 90's)
  - Widespread panic regarding programming model

Office of Science U.S. DEPARTMENT OF ENERGY

Tension between concurrency and power efficiency

- **Highly concurrent systems can be more power efficient**
  - *Dynamic power is proportional to V^2fC*
  - *Build systems with even higher concurrency?*
- **However, many algorithms are unable to exploit massive concurrency yet**
  - *If higher concurrency cannot deliver faster time to solution, then power efficiency benefit wasted*
  - *So we should build fewer/faster processors?*
- **With Massive Concurrency, Assumptions our current software infrastructure is built upon are no longer valid**
  - Programming model will break
  - System software will break
  - Applications will break
  - Hardware will be unbalanced
- ***Some of these fears are unfounded***
  Some require fundamental SW/HW innovation

Office of Science U.S. DEPARTMENT OF ENERGY

Looking Forward

- **15 years of relentless clock frequency scaling has killed research into innovative architectures**
  - Why innovate when brute force will win?
  - Need to minimize capital costs favors commoditization (press consumer electronics into service of science)
  - DOE went from fundamental architecture research to building wacky prototype systems from existing hardware
- **Clock frequency stall and power density issues strongly favor innovation**
  - Performance "ceiling" now limited by computational efficiency rather than peak perf.
  - New economic model will favor specialization to task (from embedded)
  - This affects everything from cell phones to *BOGO*flops HPC.

Office of Science
U.S. DEPARTMENT OF ENERGY

Other Stuff

- **Need deeper analysis of scientific application resource requirements at the microarchitectural level (resolve long simmering debates about balance)**
- **Locality important**
  - Cannot make user responsible for all specification of parallelism and locality
  - But current languages do not offer sufficient semantic guarantees of locality (makes autoparallelization difficult & unbounded search/analysis… intractable job for a compiler)
  - Control of side-effects is well understood in functional languages and years of research from the 80's (dig it back up)

Office of Science
U.S. DEPARTMENT OF ENERGY

Multicore is NOT an SMP-on-a-Chip

- **What about Message Passing on a chip?**
  - MPI buffers & datastructures growing O(N) or O(N^2) a problem for constrained memory
  - Redundant use of memory for shared variables and program image
- **What about SMP on a chip?**
  - Hybrid Model: *Long and mostly unsuccessful history due to loop startup/shutdown*
  - But it is NOT an SMP on a chip
    - 10-100x higher bandwidth on chip
    - 10-100x lower latency on chip
  - SMP model ignores potential for much tighter coupling of cores
  - *Same deal for stream programming model!*
- **Looking beyond SMP**
  - Cache Coherency: *necessary but not sufficient*
  - Fine-grained language elements difficult to build on top of CC protocol
  - Hardware Support for Fine-grained hardware synchronization
  - Message Queues
  - Transactions: Protect against incorrect reasoning about concurrency

Office of Science
U.S. DEPARTMENT OF ENERGY

Conclusion

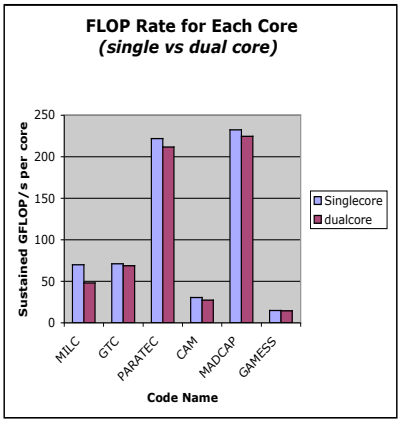
- **There is no practical path to an "Exaflop" vision of advanced computing without fundamental advances in computer architecture research**
  - Unfamiliar territory that offers more questions than available answers (classic underconstrained problem in applied math)
  - Need devices like RAMP to give software and applications people practical experience with innovative architectures before we invest $200M in fielding full production machine
  - This is the same role that simulation science plays in multi-billion dollar investments in terrestrial experiments like ITER and LHC.

Office of Science
U.S. DEPARTMENT OF ENERGY

Extra Material

Will Multicore Slam Against the Memory Wall?

- **Memory Bandwidth Starvation**
  - *"Multicore puts us on the wrong side of the memory wall. Will CMP ultimately be asphyxiated by the memory wall?"* Thomas Sterling
  - Memory wall is NOT a problem that is caused by multicore (*term coined in 1994*).



- **What about latency (other part of memory wall)**
  - Effective use of bandwidth is progressively inhibited by poor latency tolerance of modern microprocessor cores *(memory mud rather than memory wall)*
  - Stalled clock rates actually halt growing gap of memory latency / operation

- **We can fix bandwidth (but not latency)**
  - With current technology, we could put 8x more bandwidth onto chips then we currently do! *. . . GPUs and Cicso Metro already do this!*
  - So why don't we do it? . . . b*ecause it is ineffective for current processor cores*
  - *Cell/Software controlled memory can use bandwidth more effectively*

More Exotic Solutions Unlikely In the Near Term

- **FPGAs**
  - Inefficient use of chip area
  - More efficient than multicore
  - Not as efficient as manycore *(see Chris Rowen/Tensilica slide)*
  - Wire routing heuristics still troublesome
- **GPUs (prior to NVidia CUDA)**
  - Render texture maps to framebuffer (repeat)
  - Gather but no scatter
  - No inter-core communication
  - Must go to main memory for each iteration
- **Dataflow and tiled processor architectures**
  - Have considerable experience with dataflow from 1980's
  - Are we ready to return to functional programming languages?
  - Many are "rediscovering" dataflow, but call it something else
- **Cell**
  - Software controlled memory uses bandwidth efficiently
  - Programming model not yet mature

Office of Science U.S. DEPARTMENT OF ENERGY

More Exotic Solutions?

- **More exotic solutions may be our ultimate destination**
  - **But need practical experience with exotic HW to find their limits**
  - **But research pipeline is pretty empty (killed by 15 yrs of relentless clock frequency scaling)**
- **Locality is key**
  - **Must be able to expose/manage through language constructs**
  - **Slim hope of full automation of locality management *(existing serial programming languages do not offer sufficient guarantees about locality of effect. Too little information for compiler to make sane decisions)***
- **Rediscovering dataflow *(although we aren't calling it that)***
  - **Hardware implementation of transactional memory look just like dataflow activation frames from Monsoon**
  - **Similar observation on programming models for Cell and G80 GPU.**

Office of Science U.S. DEPARTMENT OF ENERGY

Need for Arch Research

- We have a lot of questions with fewer answers to match them
  - In math terms, its an under-constrained system of equations
- Need to a platform (a workbench) that enables us to set up experiments to answer some of these questions
  - Early Prototype Hardware does not provide a fast enough feedback loop
  - Science driven system architecture not fast enough feedback loop
  - Software Simulators?
  - RAMP?

Office of Science
U.S. DEPARTMENT OF ENERGY

Learning from Embedded Computing

- **Surprising Convergence of Interests between Embedded and HPC**
  - Power consumption has only recently become a top-level/performance-limiting concern for desktop and server markets
  - Power efficiency and cost have always been top-level concerns for embedded computing
    - Hotbed of innovation as they specialize to improve computational efficiency and cost!
    - Want cell phone that cost nothing and last forever on each battery charge
  - Technology from embedded market is now trickling up into server designs
    - Rather than traditional trickle down flow of innovations
    - Look at BlueGene and SiCortex
- **What will HPC learn from the embedded market? *(familiar set of lessons)***
  - Simpler, smaller cores
  - Many cores on chip (100's of cores, not 2,4,8)
  - Lower clock rates
  - More specialization to applications

Office of Science
U.S. DEPARTMENT OF ENERGY

tensilica

Basic Processor Efficiency
The Usual List of Suspects

	IBM/ Sony/ Toshiba Cell	IBM BlueGene /L (PowerPC 440 ASIC)	AMD Opteron K8L	Intel Xeon 5100 Woodcrest	Intel Itanium2	Xtensa-based SIMD/LIW Scientific Engine
DP Operations per Cycle per Processor	0.6 per SPE	4	4	4	4	4
Cycles per second (GHz)	3.2	0.7	3.0	3.0	1.4	0.65
Processors per IC	8	2	2	2	1	32
Aggregate DP GLFOPS per IC	15	5.6	24	24	5.6	83
Approx IC Power (Watts)	30	10	80	65	130	12
IC GFLOPS/Watt	0.5	0.6	0.3	0.4	0.06	7

DP FP pipelines in FPGA: 15.9 GFLOPs @ 25W (Xilinx Virtex-4 LX200): **0.63 GFLOPS/W**

16

Source: Vendor websites www.geek.com,www.answers.com

NERSC

Materials Science
Mass Migration to New Algorithms

- **Materials Science**
  - Predict bulk material properties from first principles (ab-initio)
  - One algorithm, Planewave DFT, accounts for 75% of the materials science workload
  - Codes: QBox, PARATEC, VASP
  - QBox won Gordon Bell award for scalability!

- **However, this is *not* the correct algorithm to use for petaflop scale calculations!**
  - FLOP requirements grow O(N^3)
  - Increasingly dominated by BLAS3 *(good for FLOPs)*
  - But only get to simulate marginally larger system
  - Fails to exploit locality of quantum wave component!

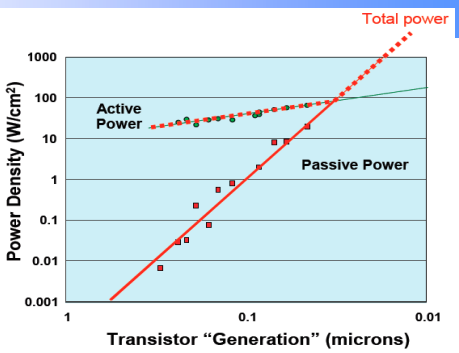
- **Classical DFT approach cannot continue!**
  - O(N) algorithms will eventually replace them
  - O(N) methods are not yet fully developed because the attention is going to classical DFT because it generates impressive FLOP rates
  - 75% of the NERSC MatSci workload is going to have to migrate to O(N) methods, but little support that migration

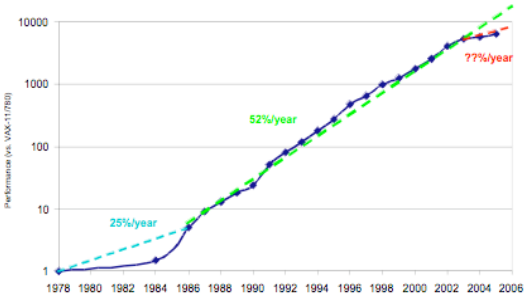
Office of Science
U.S. DEPARTMENT OF ENERGY

Why are Clock Frequencies Stalling?

- Moore's Law
  - Silicon lithography will improve by 2x every 18 months
  - Double the number of transistors per chip every 18mo.
- CMOS Power

  Total Power = $V^2 * f * C$ (active power) + $V * I_{leakage}$ (passive power)
  - As we reduce feature size Capacitance (C) decreases proportionally to transistor size
  - Enables increase of clock frequency (f) proportionally to Moore's law lithography improvements, with same power use
  - This is called "Fixed Voltage Clock Frequency Scaling" *(Borkar `99)*
- Since ~90nm
  - $V^2 * f * C$ ~= $V * I_{leakage}$
  - Can no longer take advantage of frequency scaling because passive power ($V * I_{leakage}$) dominates
  - Result is recent clock-frequency stall reflected in Patterson Graph at right





SPEC_Int benchmark performance since 1978 from Patterson & Hennessy Vol 4.

Office of Science U.S. DEPARTMENT OF ENERGY

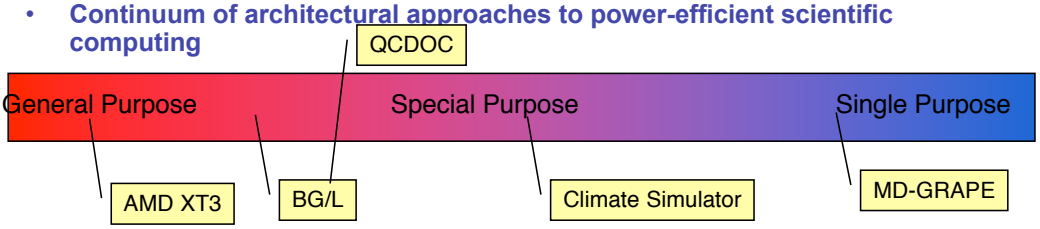
Tension between concurrency and power efficiency

- **Highly concurrent systems can be more power efficient**
  - *Dynamic power is proportional to V^2fC*
  - *Build systems with even higher concurrency*
- **However, many algorithms are unable to exploit massive concurrency yet**
  - *If higher concurrency cannot deliver faster time to solution, then power efficiency benefit wasted*
  - *so we should build fewer/faster processors*

Office of Science U.S. DEPARTMENT OF ENERGY

Special-Purpose Architecture for 1km Climate Simulation

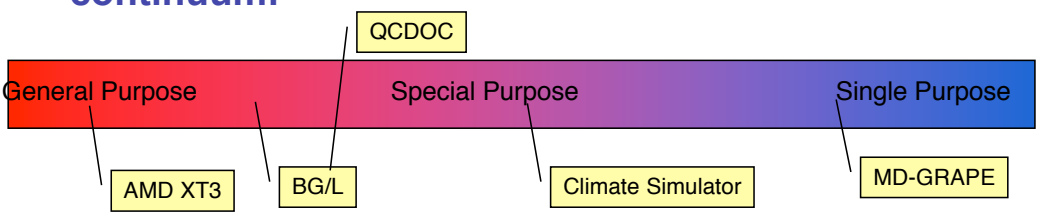
- We design system around the requirements of the km-scale climate code.
- Examined 3 different approaches
  - AMD Opteron: Commodity Approach - Lower efficiency for scientific applications offset by cost efficiencies of mass market
    - Popular building block for HPC, from commodity to tightly-coupled XT3.
    - Our AMD pricing is based on servers only without interconnect
  - BlueGene/L: Use generic embedded processor core and customize System on Chip (SoC) services around it to improve power efficiency for scientific applications
    - Power efficient approach, with high concurrency implementation
    - BG/L SOC includes logic for interconnect network
  - Tensilica: In addition to customizing the SOC, also customizes the CPU core for further power efficiency benefits but maintains programmability
    - Design includes custom chip, fabrication, raw hardware, and interconnect
- Continuum of architectural approaches to power-efficient scientific computing



Office of Science U.S. DEPARTMENT OF ENERGY

Tension Between Specialized and General Purpose

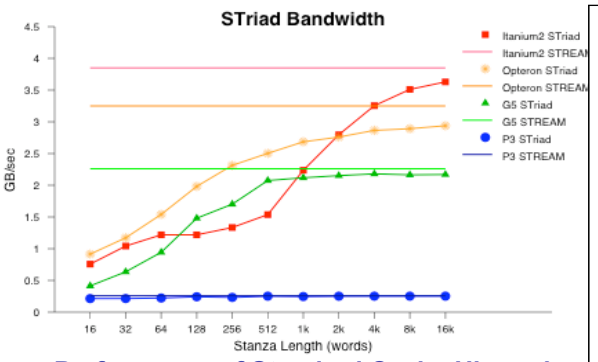
- Specialized architecture
  - more power efficient
  - lower total design cost due to narrower design target
  - Lower volume means higher component cost
- General purpose architecture
  - Less power efficient for some applications
  - Higher total design cost due to broader design target
  - High volume means lower component costs
- The choices for degree of specialization lie on a continuum!

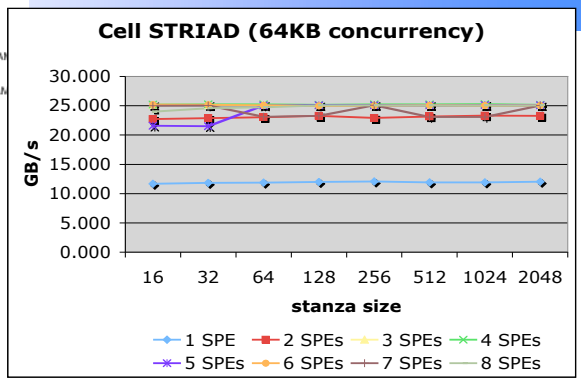


Office of Science U.S. DEPARTMENT OF ENERGY

Why is the STI Cell So Efficient?

(understanding memory subsystem response)





- **Performance of Standard Cache Hierarchy**
  - Cache hierarchies underutilize memory bandwidth due to inability to tolerate latency
  - Hardware prefetch prefers long unit-stride access patterns (optimized for STREAM)
  - But in practice, access patterns are for shorter stanzas: so never reaches peak bandwidth (still latency limited)
- **Cell "explicit DMA"**
  - Cell software controlled DMA engines can provide nearly flat response for a variety of access patterns
  - Response is nearly full memory bandwidth can be utilized for all access patterns
  - Cell memory requests can be nearly completely hidden behind the computation due to asynchronous DMA engines
  - Performance model is simple and deterministic (much simpler than modeling a complex cache hierarchy), min{time_for_memory_ops, time_for_core_exec}

Office of Science
U.S. DEPARTMENT OF ENERGY

Much Ado about Dwarves

High-end simulation in the physical sciences = 7 numerical methods:

1. **Structured Grids (including locally structured grids, e.g. Adaptive Mesh Refinement)**
2. **Unstructured Grids**
3. **Fast Fourier Transform**
4. **Dense Linear Algebra**
5. **Sparse Linear Algebra**
6. **Particles**
7. **Monte Carlo**

Why are they interesting?

- *Benchmarks enable assessment of hardware performance improvements*
- *The problem with benchmarks is that they enshrine an implementation*
- *At this point in time, we need flexibility to innovate both implementation and the hardware they run on!*
- *Dwarves provide that necessary abstraction*

Slide from "Defining Software Requirements for Scientific Computing", Phillip Colella, 2004

Office of Science
U.S. DEPARTMENT OF ENERGY

Dwarf Popularity (Red Hot → Blue Cool)

	Embed	SPEC	DB	Games	ML	HPC
1 Finite State Mach.						
2 Combinational						
3 Graph Traversal						
4 Structured Grid						
5 Dense Matrix						
6 Sparse Matrix						
7 Spectral (FFT)						
8 Dynamic Prog						
9 N-Body						
10 MapReduce						
11 Backtrack/ B&B						
12 Graphical Models						
13 Unstructured Grid						

Claim: parallel architecture, language, compiler … must do at least these well to run future parallel apps well

Office of Science U.S. DEPARTMENT OF ENERGY

Architectural Exploration using RAMP

What is Berkeley RAMP: Research Accelerator for Multiple Processors

- Sea of FPGAs linked together via hypertransport
- Provides enough programmable *gates* to simulate large chip designs
- Building community of "open source" hardware components (GateWare)
  - PPC4xx cores, Sun Niagra-1 netlists, Tensilica netlists
- Assemble gateware components (CPU and interconnects) using RDL (RAMP Description Language)
- Enables emulation of large clusters (100's or 1000's of nodes) using $20K FPGA board.
  - Boots Linux - it looks like the *real* hardware to the software
  - Runs 100x slower than realtime, compared w/ million time slowdown of simulators
  - Can change HW parameters and explore new design on daily basis
- **NERSC intends to use RAMP to test out tomorrow's hardware *before* it is delivered**
  - Will to simulations of BG/Q chip design (multi-petaflop hardware in development for 2010 release)
  - Will use RAMP to test out theories about architectural features that may benefit scientific application performance

Office of Science U.S. DEPARTMENT OF ENERGY

IO Performance

- **Many users have not made the transition to parallel IO**
  - Even the climate code still has some serial-IO *(and they have a big SW development effort!)*
  - All of these deficiencies will become painfully obvious as we move from 6k processors to 20k, then 100k etc…
- **All HPC centers should expect a mass migration (or panic) to modern parallel IO methods**
  - Users who have thus far avoided thinking about IO will be forced to confront these issues or suffer greatly
  - This will be the *least* expert of our users
  - They will all have this revelation at nearly the same time (when they first run on a 20k processor system)
  - They will be knocking down the door of the consulting office

Office of Science
U.S. DEPARTMENT OF ENERGY

Disconnect Between Productive Science and Easy Scaling

- **Combustion/Adaptive Mesh Refinement**
  - Not limited by bisection bandwidth
  - Dominated by compute component for relevant problems
  - Scaling of Hyperbolic problems trivial, Elliptic problems challenging
- **Materials Science (PARATEC, LS3DF)**
  - Dominant algorithm: Planewave DFT dominates materials science workload at NERSC
  - Dominated by $O(N^3)$ localized BLAS3 at petaflop scale (*good for FLOP rates)*
  - Must move to O(N) methods beyond 1k atoms (*mass migration required)*
- **Accelerator Modeling**
  - Currently formulated as direct solve on sparse matrix and will not scale
  - Moving to petaflop scale requires innovation in the mathematical formulation of the problem!
- **NERSC is handling all of the most difficult-to-scale applications**
  - Leadership application selection process favors applications that already demonstrate scalability
  - Resulting distilling process concentrates hard-to-scale applications at NERSC!

Office of Science
U.S. DEPARTMENT OF ENERGY