BLUE WATERS

SUSTAINED PETASCALE COMPUTING

May 6, 2010

Exascale Computing: The Last Rehearsal Before the Post-Moore Era

Marc Snir / Bill Gropp / Bill Kramer

NCSA NSF IBM GREAT LAKES CONSORTIUM FOR PETASCALE COMPUTATION

THE WORLD IS ENDING

THE (CMOS) WORLD IS ENDING

End of CMOS?

IN THE LONG TERM (~2017 THROUGH 2024) ***While power consumption is an urgent challenge, its leakage or static component will become a major industry crisis in the long term, threatening the survival of CMOS technology itself, just as bipolar technology was threatened and eventually disposed of decades ago.*** [ITRS 2009]

- Unlike the situation at the end of the bipolar era, no technology is waiting in the wings.

"POST-CONVENTIONAL CMOS"

- New materials
  - .. such as III-V or germanium thin channels on silicon, or even semiconductor nanowires, carbon nanotubes, graphene or others may be needed.
- New structures
  - three-dimensional architecture, such as vertically stackable cell arrays in monolithic integration, with acceptable yield and performance.
- ROI challenges
  - … achieving constant/improved ratio of … cost to throughput might be an insoluble dilemma.
- …These are huge industry challenges to simply imagine and define

- Note: feature size in 2021 (13 nm) = ~55 silicon atoms (Si-Si lattice distance is 0.235 nm)

The Post-Moore Era

- **Scaling is ending**
  - Voltage scaling ended in 2004 (leakage current)
  - Feature scaling will end in 202x (not enough atoms)
  - Scaling rate will slow down in the next few years
  - Continued scaling in the next decade will need a sequence of (small) miracles (new materials, new structures, new manufacturing technologies)
- ☞ ***Compute Efficiency*** **becomes a paramount concern**
  - More computations per joule
  - More computations per transistor

HPC – The Canary in the Mine

- **HPC is already heavily constrained by low compute efficiency**
  - High power consumption is at the limit of current machine rooms
  - Low thread performance entails high levels of parallelism
- **Higher compute efficiency is essential for exascale computing (Kogge's report)**
- **Current Petascale systems are ideal test-bed for research to increase compute efficiency**

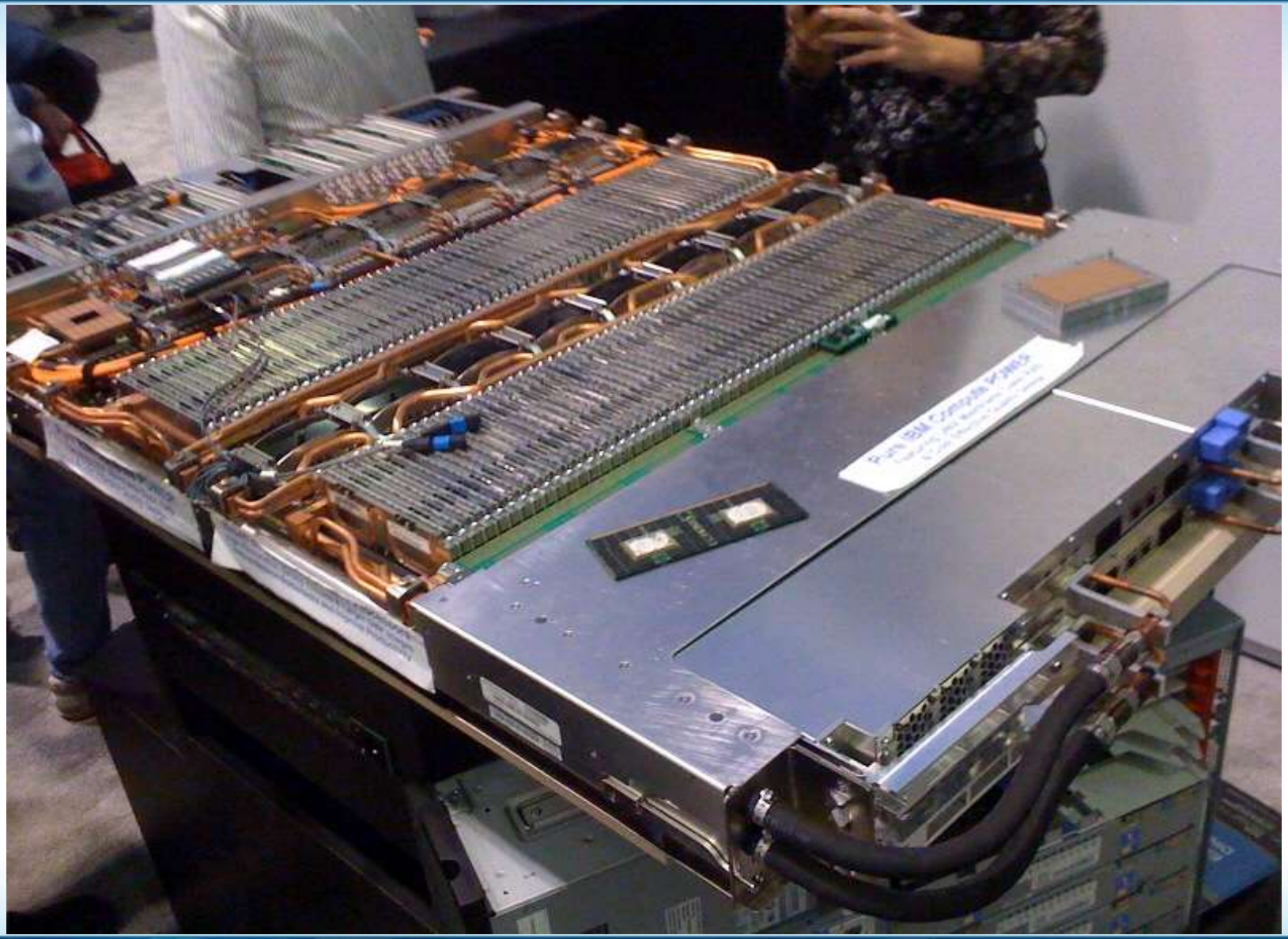
PETASCALE TODAY

Blue Waters

PETASCALE IN A YEAR

(Soon to be) Current Petascale Platforms: Blue Waters

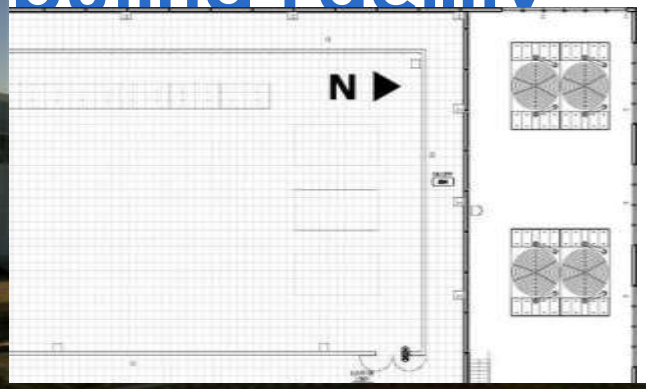
System Attribute	Blue Waters
Vendor	IBM
Processor	IBM Power7
Peak Performance (PF)	~10
Sustained Performance (PF)	~1
Number of Cores/Chip	8
Number of Cores	>300,000
Amount of Memory (PB)	>1
Amount of Disk Storage (PB)	18
Amount of Archival Storage (PB)	>500
External Bandwidth (Gbps)	100-400
Water Cooled	



Blue Waters is not "Out-of-the-Box" Product

- Work on power-efficient compute center
- Work on integration of Blue Waters in production environment
  - Storage, Networking, Workflow, Training…
- Collaborations with IBM and other partners on enhancements to essentially all software layers

putina Facility



Cooling Towers

PCF is Near University Power and

ture

Resource manager: Batch and interactive access

Performance tuning: HPC and HPCS toolkits, open source tools

Parallel debugging at full scale

Environment: Traditional (command line), Eclipse IDE (application development, debugging, performance tuning, job and workflow management)

Languages: C/C++, Fortran (77-2008 including CAF), UPC

Libraries: MASS, ESSL, PESSL, PETSc, visualization…

Programming Models: MPI/MP2, OpenMP, PGAS, Charm++, Cactus

Low-level communications API supporting active messages (LAPI)

IO Model: Global, Parallel shared file system (>10 PB) and archival storage (GPFS/HPSS) MPI I/O

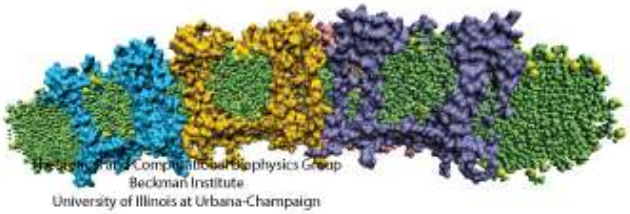
Full – featured OS Sockets, threads, shared memory, checkpoint/restart

Hardware
Multicore POWER7 processor with Simultaneous MultiThreading (SMT) and Vector MultiMedia Extensions
Private L1, L2 cache per core, shared L3 cache per chip
High-Performance, low-latency interconnect supporting RDMA

Illinois-IBM Collaborations: No Software Layer Left Untouched

- Integrated System Monitoring
  - Capture (in DB) extensive “system health” information
  - Develop automatic action triggers, failure root-cause diagnostics and datamining for predictive maintenance
- Novel Storage infrastructure
  - Disk as cache for tape storage; automatic data staging
- PGAS programming models with interoperability
- Eclipse based Integrated development environment (IDE) to support compile/debug/tune/exploit processes for heterogeneous codes
- Performance engineering
- Compiler enhancements, libraries, workflow…

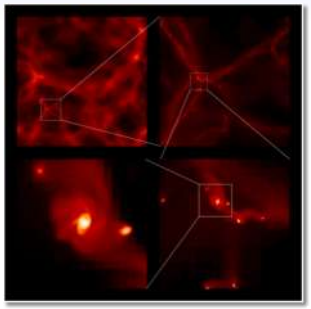
Molecular Dynamics



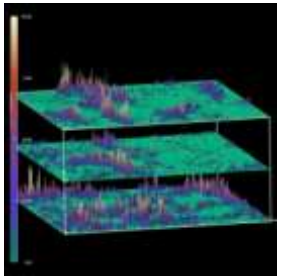
Climate

Star Evolution

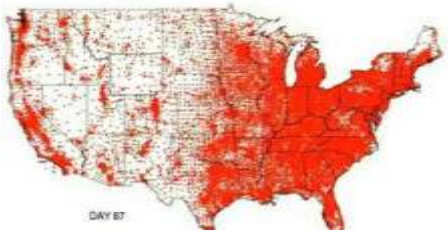
Galaxy Formation



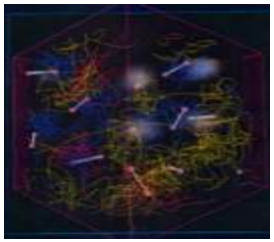
Turbulence



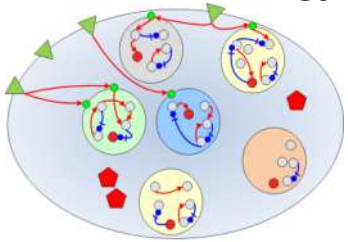
Contagion simulation



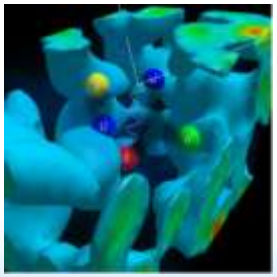
Quantum Monte Carlo



Multiscale Biology

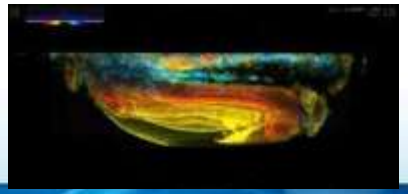


Lattice QCD

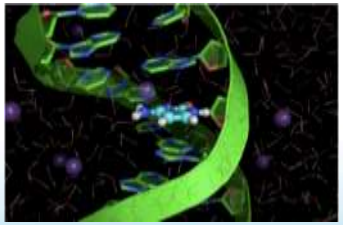


Tornados

Earthquakes



quantum chemistry



EXASCALE TOMORROW

Exascale in 2018 at 20 MWatts (?)

- We need it ASAP
  - ***Science, not nuclear weapons***: Climate, energy…
- It's hard [Kogge]
  - Conventional designs plateau at 100 PF (peak) – all energy is used to move data
  - Aggressive design is at 70 MW and is very hard to use
    - 600M instruction/cycle
    - 0.0036 Byte/flop
    - No ECC, no redundancy
    - No caching (addressable workpad)
    - HW failure every 35 minutes
    - …
- Waiting 3-4 years does not solve the problem
  - ***Exascale in CMOS requires revolutionary advances in compute efficiency***
  - ***The magnitude of the task requires long-term, science-focused international collaboration – i.e. NSF mission***
    - (much beyond G8, IESP)

Increasing Compute Efficiency (Software)

- Resiliency
- Communication-optimal computations
- Low entropy computations
- Steady-state computations
- Friction-less software layering
- Self-organizing computations

Resiliency

- HW for fault correction (and possibly fault detection) is too expensive
  - and is source of jitter
- Current global checkpoint/restart algorithms cannot cope with MTBF of few hours or less
- Research community has limited sources of information on types of failure and failure rates
  - Industry keeps such information confidential
  - Transient HW errors usually cause SW failures – root cause analysis is hard

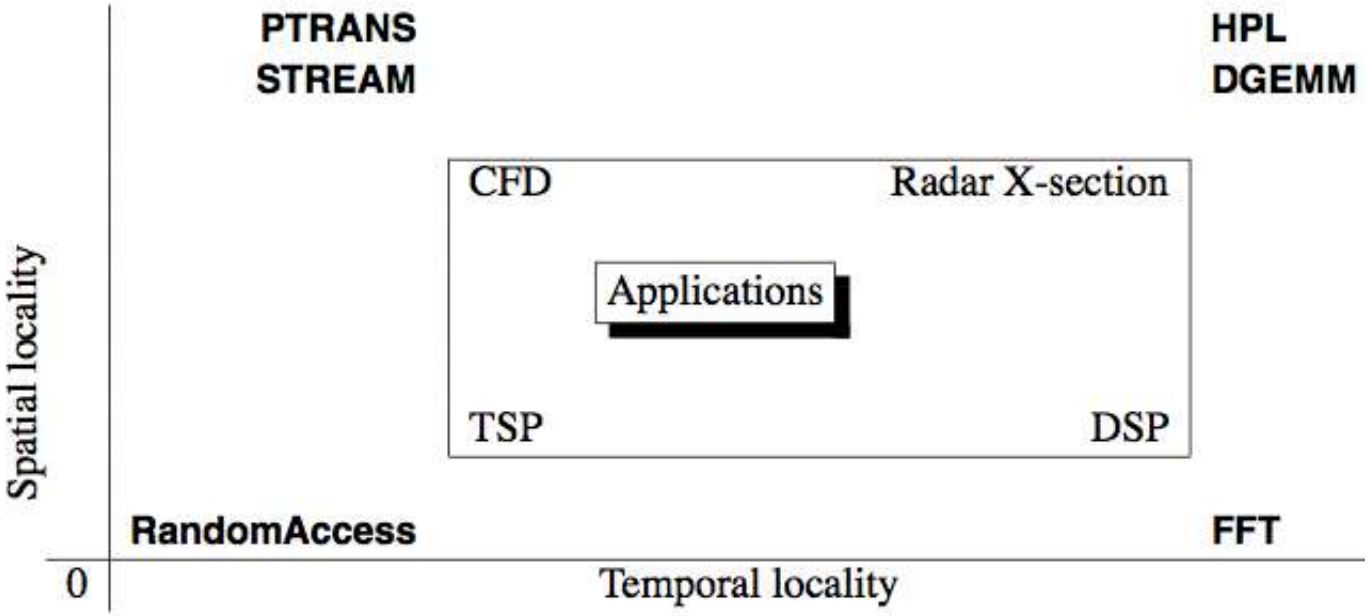
Resiliency 2

- Blue Waters is ideal research instrument to study failure types and their frequencies
  - Large
  - Heavily instrumented
  - Expect there will be an observable “shielding effect” (and can add shielding to accentuate effect)
- Illinois has significant expertise in this area
- *Propose to create at NCSA a center of excellence for large system resilience*
  - Collect and make available observation data on failure types and rates (from BW and other systems)
  - Develop new recovery algorithms
  - Develop SW based error detection protocols
  - Develop resilient algorithms (***need new theory***)

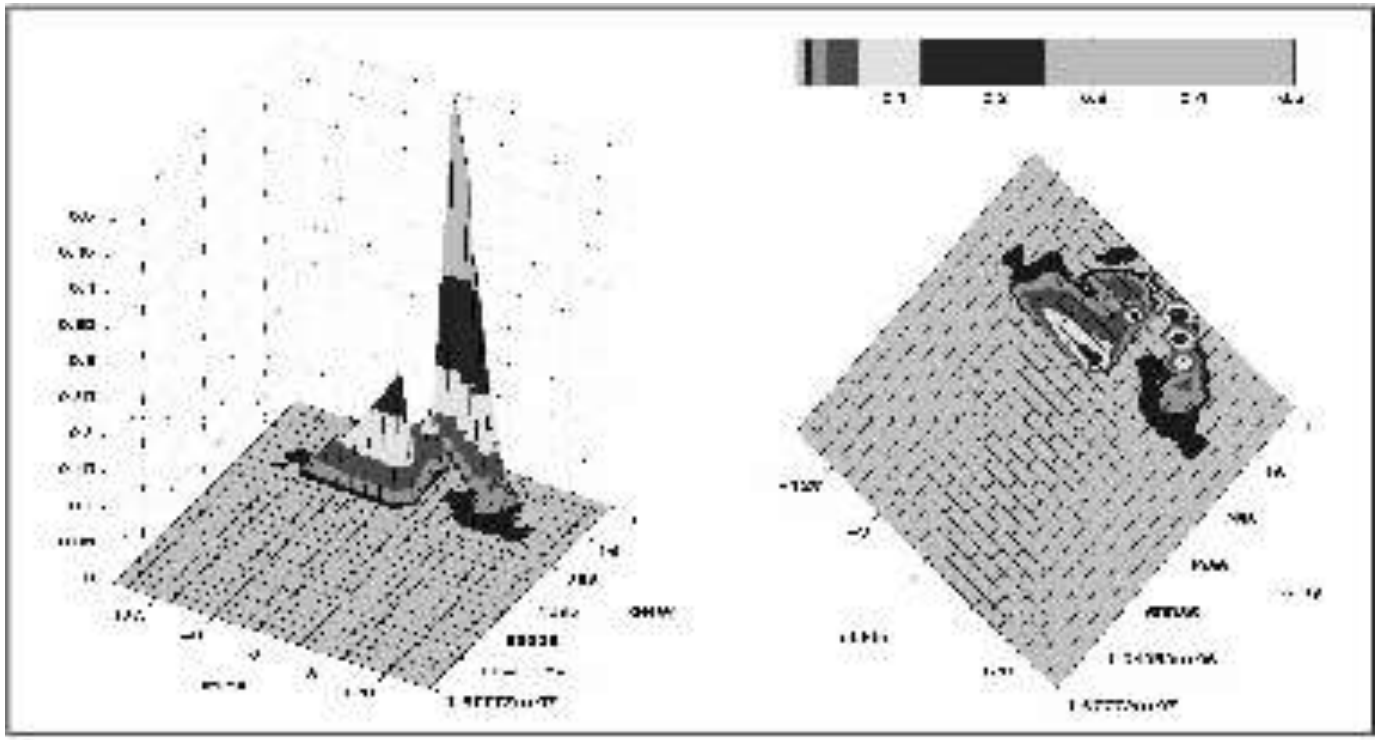
Communication-Efficient Algorithms

- Communication in time (memory) and space is, by far, the major source of energy consumption
- Our understanding of inherent communication needs of algorithms and communication-efficient algorithm design is very limited (FFT, dense linear algebra)
- Current characterization of communication patterns is deficient (dimensionality of space is not understood)
- Need:
  - Better theory of communication complexity
  - Better benchmarks
  - Better metrics
  - Communication-focused language and perf. analysis

Naïve 2D View



Spatio-Temporal Locality Surface



Gzip

Sorenson-Flanagan

Parallelism introduces (at least) one extra dimension

Low-Entropy Communication

- Communication can be much cheaper if "known in advance"
  - Latency hiding, reduced arbitration cost, bulk transfers… bulk mail vs. express mail
- Current HW/SW architectures take little advantage of such knowledge
- CS is lacking a good algorithmic theory of entropy
- Need theory, benchmarks, metrics

A Possible Benchmarking Center

- Theory-based micro-benchmarks and analytical models for characterizing parallel system performance and "encouraging" efficient implementations
  - E.g. MPI: Linear search structures, sub-optimal collective communication, computation interference
- Theory based metrics and synthetic benchmarks for application characterization (locality, entropy)
- Performance models for NSF applications

Steady-State Computation

- Each subsystem of a large system (CPU, memory, interconnect, disk) has low average utilization during a long computation
- Each subsystem is the performance bottleneck during part of the computation.
- Utilization is not steady-state – hence need to over-provision each subsystem.
- Proposed solution A: power management, to reduce subsystem consumption when not on critical path.
  - Hard (in theory and in practice)
- Proposed solution B: Techniques for steady-state computation
  - E.g., communication/computation overlap
- Need research in Software (programming models, compilers, run-time), and architecture.

Friction-less Software Layering

- Current HW/SW architectures have developed multiple, rigid levels of abstraction (ISA, VM, APIs, languages…)
  - Facilitates SW development but energy is lost at layer matching
- Flexible matching enables to regain lost performance
  - Inlining, on-line compilation, code morphing…
  - Similar techniques are needed for OS layers

Self-Organizing Computations

- Hardware continuously changes (failures, power management)
- Algorithms have more dynamic behavior (multigrid, multiscale – adapt to evolution of simulated system)

☞ Mapping of computation to HW needs to be continuously adjusted

- Too hard to do in a centralized manner -> Need distributed, hill climbing algorithms

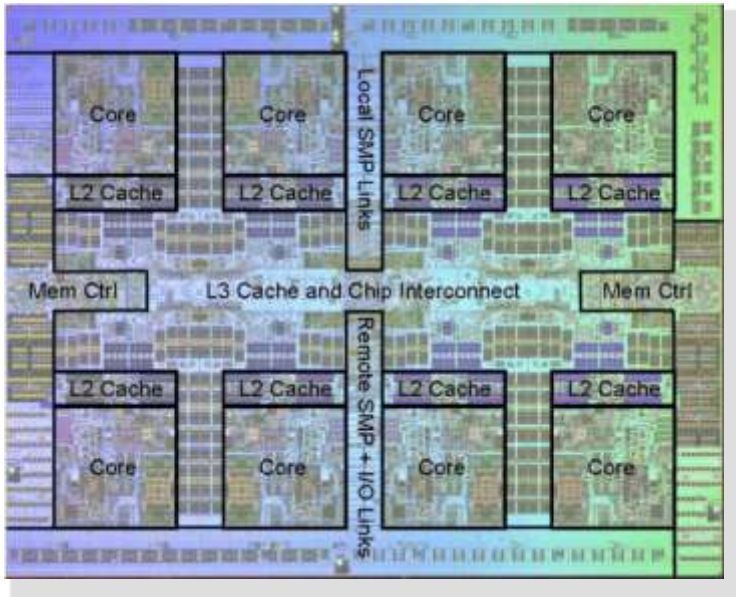
Summary

- Need to be in the right Pasteur's quadrant: deep, use motivated research
  - NCSA has the use (ahead of the broad community)
  - Illinois has the depth in parallel computing, resilience, architecture, compilers…
- We need a tight coupling of advanced CS/CE research with experimental research that leverages Blue Waters and other large platforms
- Supercomputing Centers should not only be tools for current computational scientists but tool builders for the next generation of computational scientists and pioneers in research on computing efficiency for the post-Moore era.

BACKUP

Power7 Chip: Computational Heart of Blue Waters

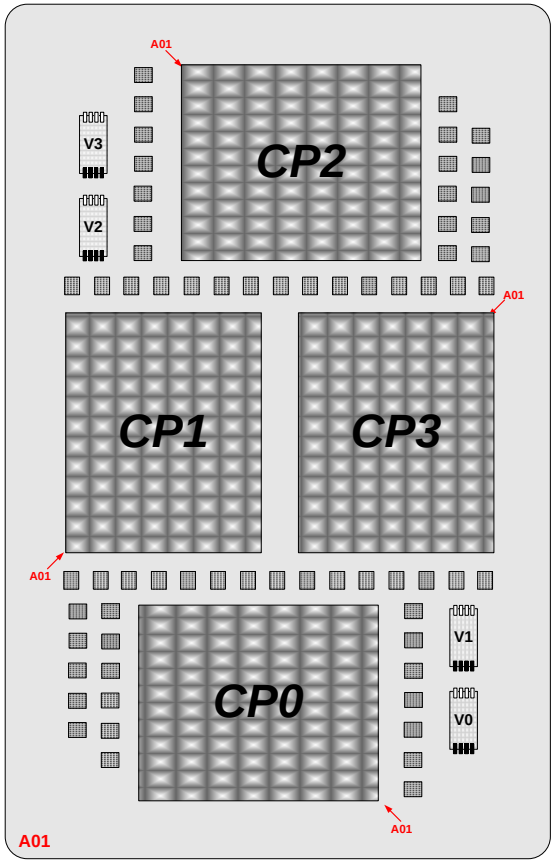
- Base Technology
  - 45 nm, 576 mm2
  - 1.2 B transistors
  - 3.5 – 4 GHz
- Chip
  - 8 cores
  - 12 execution units/core
  - 1, 2, 4 way SMT/core
  - Caches
    - 32 KB I, D-cache, 256 KB L2/core
    - 32 MB L3 (private/shared)
  - Dual DDR3 memory controllers
    - 100 GB/s sustained memory bandwidth



Power7 Chip

Quad-chip MCM

Node = Quad-Chip Module + Hub chip



Hub Chip Module

- Connects to QCM (192 GB/s)
- Connects to 2 PCI-e slots (40 GB/s)
- Connects to 7 other QCM's in same **drawer** (336 GB/s – copper fabric)
  - Enables a single hypervisor to run across 8 QCM's
  - Allows I/O slots attached to the 8 hubs to be shared
- Connects four drawers together into a **supernode** (240 GB/s per hub – optical bus)
- Connects up to 512 supernodes together (320 GB/s per hub – optical bus)

Rack

- 990.6w x 1828.8d x 2108.2
- 39”w x 72”d x 83”h
- ~2948kg (~6500lbs)

Data Center In a Rack

Compute

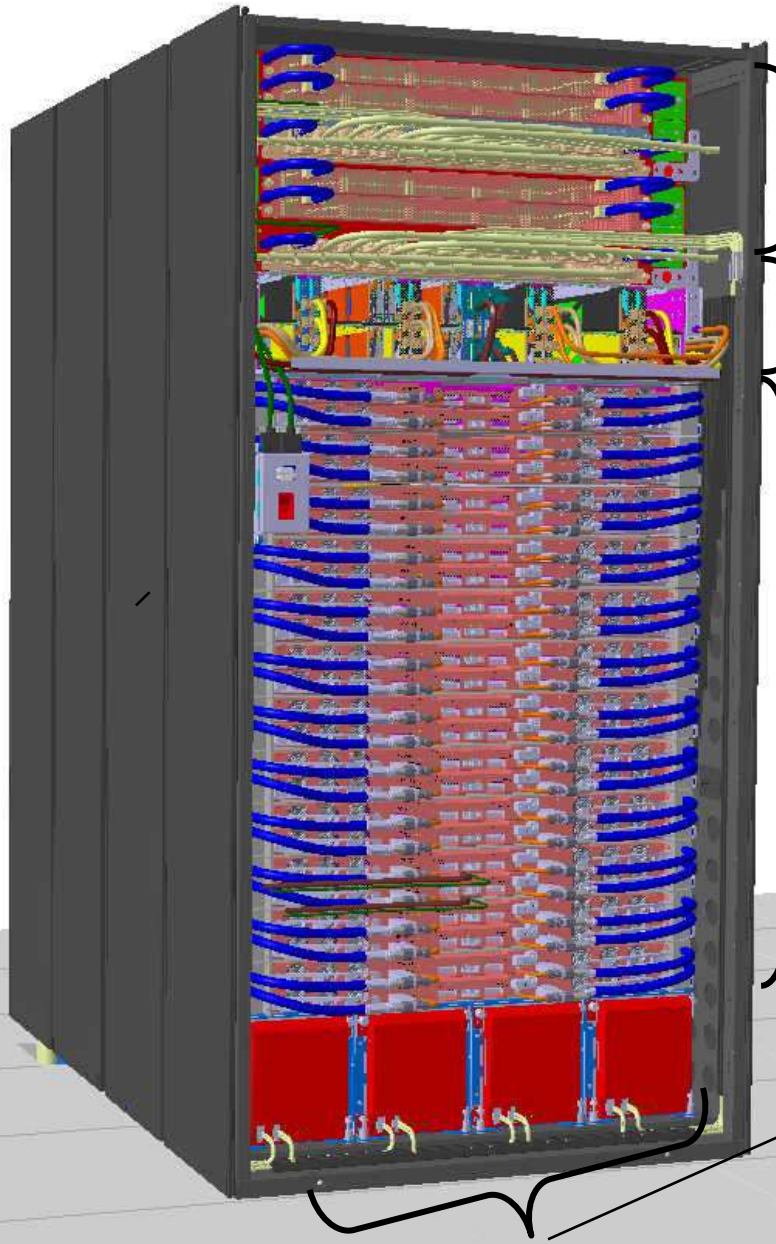
Storage

Switch

100% Cooling

PDU Eliminated

Input: 8 Water Lines, 4 Power Cords
Out: ~100TFLOPs / 24.6TB / 153.5TB
192 PCI-e 16x / 12 PCI-e 8x



BPA

- 200 to 480Vac
- 370 to 575Vdc
- Redundant Power
- Direct Site Power Feed
- PDU Elimination

Storage Unit

- 4U
- 0-6 / Rack
- Up To 384 SFF DASD / Unit
- File System

CECs

- 2U
- 1-12 CECs/Rack
- 256 Cores
- 128 SN DIMM Slots / CEC
- 8,16, (32) GB DIMMs
- 17 PCI-e Slots
- Imbedded Switch
- Redundant DCA
- NW Fabric
- Up to:3072 cores, 24.6TB (49.2TB)

WCU

- Facility Water Input
- 100% Heat to Water
- Redundant Cooling
- CRAH Eliminated