Argonne
NATIONAL
LABORATORY

... for a brighter future

ALCF

Argonne Leadership Computing Facility

U.S. Department of Energy

UChicago ▸ Argonne LLC

Office of Science
U.S. DEPARTMENT OF ENERGY

A U.S. Department of Energy laboratory managed by UChicago Argonne, LLC

Trends in Architecture

William Gropp
(for Ray Bair, Director of ALCF)

What is the ALCF?

- The Argonne Leadership Computing Facility is a new division at Argonne
  - A peer of the Mathematics and Computer Science Division
  - A home for Petascale computing at ANL
  - ACLF is currently home to a 1k node (5.6 TF) BG/L; deploying next generation BlueGene this fall
  - DOE INCITE Program awards time to open science projects; 9 currently awarded 10M CPU Hours (on BG/L at ANL and IBM)
- Announced Plans

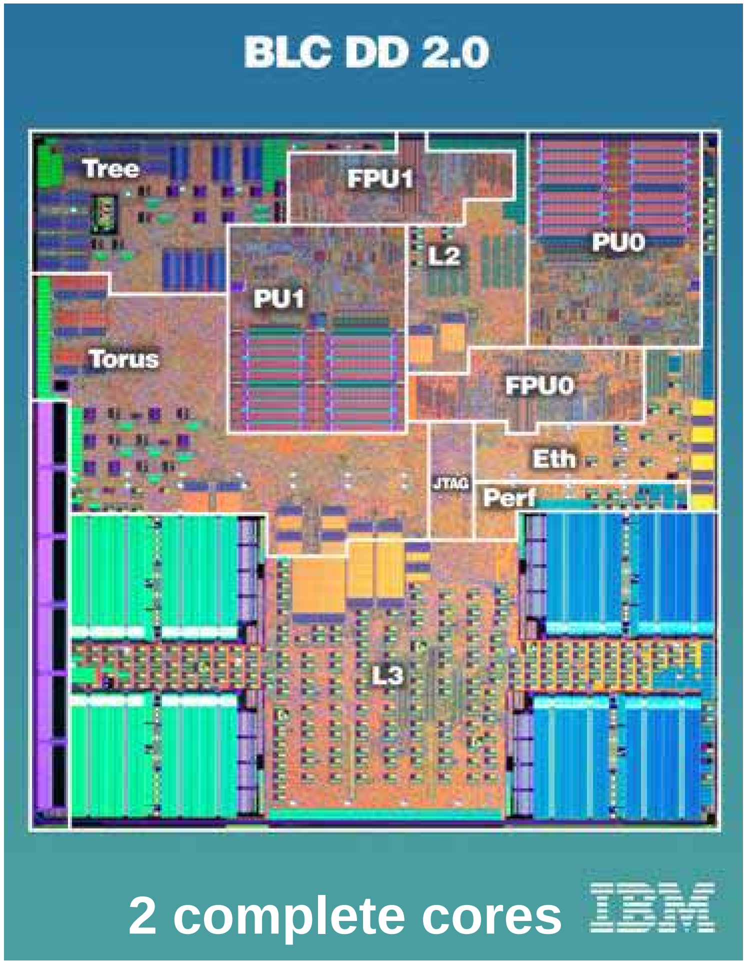
2007

- Increased to 9 INCITE projects; continue development projects
- Install 100 teraflops next gen Blue Gene system (late 2007)

2008

- Begin support of INCITE projects on next generation Blue Gene
- Add 250-500 Teraflops Blue Gene system

BlueGene/L Chip



Just add DRAM

Processor

- PPC440x5 Processor Core – 700 MHz
  - *Superscalar: 2 instructions per cycle*
  - *Out of order issue and execution*
  - *Dynamic branch prediction, etc.*
- Two 64-bit floating point units
  - *SIMD instruct. over both register files*
  - *Parallel (quadword) loads/stores*
  - *2.8 GFLOPS/processor*

Interconnect

- 3 Dimensional Torus
  - *Virtual cut-through hardware routing*
  - *1.4Gb/s on all 12 node links*
  - *1 µs latency bet. neighbors, 5 µs to farthest*
- Global Tree
  - *One-to-all broadcast, reduction functionality*
  - *2.8 Gb/s of bandwidth per link*
  - *Latency of one way tree traversal 2.5 µs*
- Low Latency Global Barrier and Interrupt
  - *Latency of round trip 1.3 µs*
- Ethernet
  - *All external comm. (file I/O, control, etc.)*

Old and New Conventional Wisdom

- Memory wall *is* old conventional wisdom - observation dates from 1995 (Wulf and McKee)!
- Cache coherency is required
  - Long history of oscillations (at the high end)
  - Hard to support at scale
    - *In practice, hard even for 2-4 cores; many examples of either correctness or performance bugs*
  - Relaxed consistency models may work, particularly with programming model support. Algorithms can help.
- General purpose machines exist
  - All machines optimized for some workload which is not yours
- Heterogeneity is coming
  - Its already here!
    - *PCs contain multiple processors*
    - *Game engines (the PS2 was heterogeneous)*

Crystal Ball Gazing

- Power remains a constraint on everything (CPU speed, memory, etc.)
  - Algorithms need higher memory density (more accuracy/word)
- Compute Notes
  - Massive parallelism (10^7), modest memory per node.
    - *May overprovision for faults during manufacture and operation*
  - Increasing number of functional units/CPU
    - *But see power*
  - Increasing numbers of CPUs/node
    - *May not be cache coherent over entire node*
  - Heterogeneous processing elements
  - Multicore is (already!) commodity
    - *Just about the only practical route to continued performance increases without radical (though already prototyped) alternatives*

Crystal Ball Gazing con’t

- Interconnect
  - 1/2-2 usec message latency (comparable to main memory latency so unlikely to be much faster); good shared interconnect bandwidth at the cost of faster individual links
    - *Algorithms need to support concurrency in communication per node*
  - Support for remote memory operations
  - Support for some form of remote atomic operation
    - *More than compare-and-swap; perhaps remote thread or split operation*
    - *What is the right operation? The world wonders*
  - Support for relatively fast subsets of collective operations (e.g., Allreduce on COMM_WORLD)
    - *But still not fast enough at the largest scales*
  - Needs higher degree at massive scale
    - *Work needed on hierarchical algorithms*
    - *Quiz: Is the time complexity of MPI_Bcast O(nlog p) for long messages?*

Crystal Ball Gazing con't

- I/O
  - I/O for parallel jobs is collective. File systems will come recognize and exploit that … or die
  - Effective parallel file systems exploiting precise, non-POSIX semantics (related to memory consistency rules, already known to be unscalable)
- Commodity processors are already multicore
  - Two phases for multicore:
    - *Small scale, where simple, task parallelism works*
    - *Large scale (O(1000) cores), where fine grain parallelism is required*
    - *Because of memory, each core may need to support dozens to hundred of threads*
- Increasing challenge for the software in supporting scaling and per-node performance. Longer term will push all of the above; may exploit/integrate/switch to techniques used in graphics processors.
- The machine of 2016 is probably more of the same, but with more concurrency and bandwidth. Beyond that…

Disruptive Technologies

- Compound interest
  - 10^6 x improvement in CPU performance was evolutionary. This is bad?
- Integrated CPU and memory
  - 10-100X bandwidth to local memory (e.g., PIM, IRAM)
  - Basic op is not a word, rather a line (e.g., 128 words or more)
- Commodity lightweight threads
  - Practical mechanism for hiding latency
  - One form already in use in Graphics processors
- Reversible logic
  - Theoretical advantages in power
  - Many practical problems (achieved clock rate is one)
- Quantum Computing
  - Someone had to say it
  - Not a panacea
    - *Does some things much better, doesn't help with others*
    - *Google "limits of quantum computing"*
- Automatic Software