



IBM Research Division

Capability Machines: BlueGene/L

Ruud A. Haring

March 2, 2006

© 2006 IBM Corporation

Acknowledgements



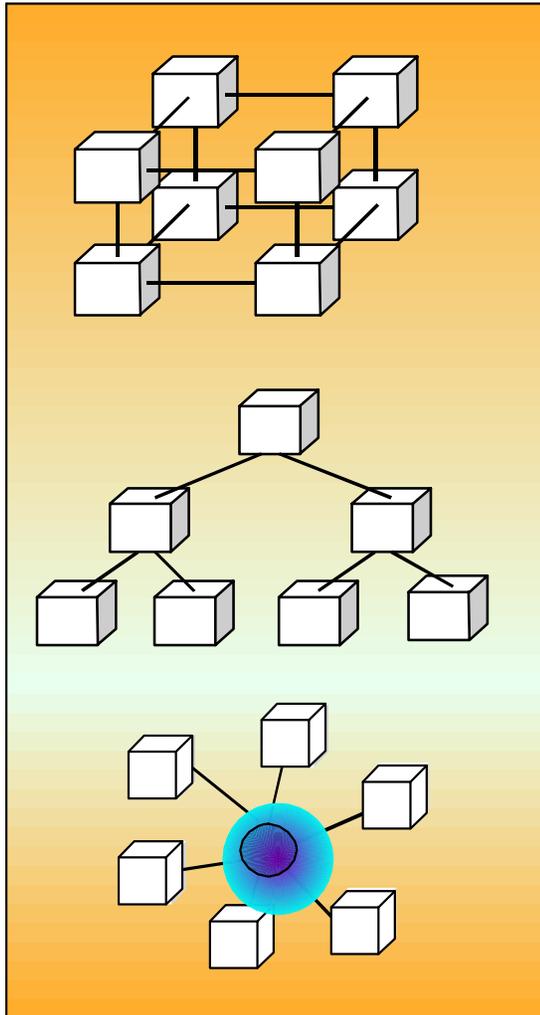
BGW: The **BlueGene/L** installation at the IBM Thomas J Watson Research Center, Yorktown Heights, NY.

- **20 rack BlueGene/L machine:**
 - each compute node:
 - dual PPC440 core ASIC,
 - 5.6 GFlop/s at 700 MHz,
 - 512 MB DRAM, 17W
 - 20 racks x 1024 Compute Nodes :
 - 114 TFlop/s peak;
 - 91.3 TFlop/s sustained (Linpack)
 - **Ranks #2 on Top500,**
June and November 2005
(#1 is 64-rack BlueGene/L system at LLNL)
 - 320 IO nodes, each 1 Gbps Ethernet
 - ~ 500 kW



BlueGene/L Interconnection Networks

3 Dimensional Torus



- Interconnects all compute nodes
- Virtual cut-through hardware routing
- 1.4Gb/s on all 12 node links (2.1 GB/s per node)
- 1 μ s latency between nearest neighbors, 5 μ s to the farthest
- MPI: 3.3 μ s latency for one hop, 10 μ s to the farthest
- Communications backbone for computations

Collective Network

- One-to-all broadcast functionality
- Reduction operations functionality
- 2.8 Gb/s bandwidth per link
- Latency of one way tree traversal 2.5 μ s, MPI 6 μ s
- Interconnects all compute and I/O nodes

Low Latency Global Barrier and Interrupt

- Latency of one way to reach 64K nodes 0.65 μ s, MPI 1.6 μ s

BGW machine

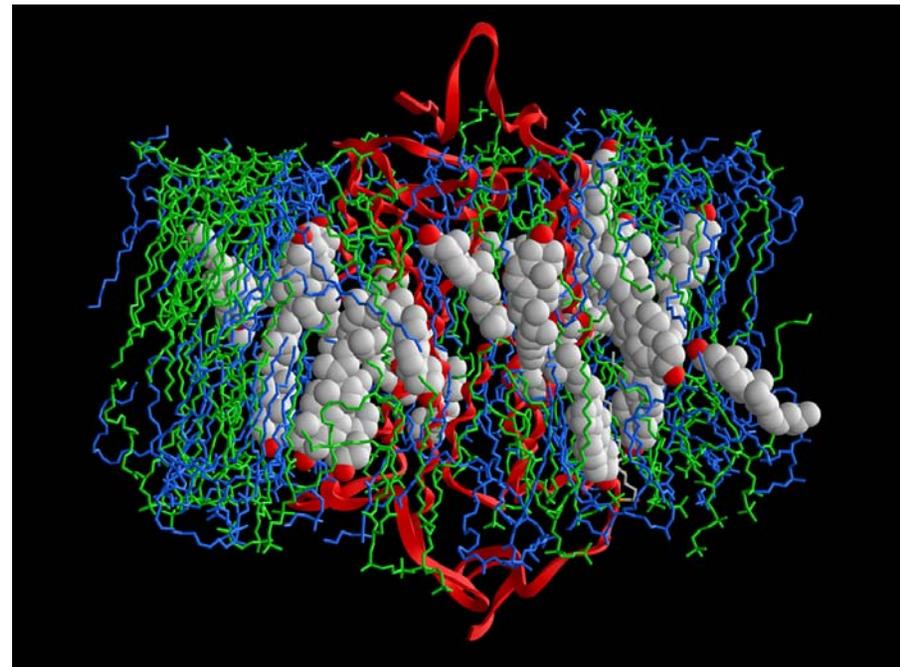
–Supported by:

- Service Node (configuration, boot-up, RAS): 8-way IBM pSeries 655
- Front-End Nodes (users, application hosts): 3 x 4-way IBM pSeries 655
- SN and FEN are PowerPC (POWER4+),
running Linux (SUSE™ SLES9).

- 60 TByte GPFS file system
- 500 TByte tape archive

Biomolecular Simulation on Blue Gene/L at Watson (BGW)

- **Rhodopsin: Light sensitive protein in “rods and cones” cells of retina.**
 - Rhodopsin spans across cell membrane (lipid bilayer with cholesterol)
 - Model for G-protein coupled receptor (GPCR).
 - Diseases associated with malfunction of GPCRs are:
 - Congestive Heart Failure, Hypertension,
 - Stroke, Cancer, Ulcers, Allergies, Asthma, Anxiety, Psychosis, Migraines,
 - Parkinson’s Disease.
 - GPCRs are therefore targets for many drugs



Biomolecular Simulation on BGW – ct.

Rhodopsin + lipids + cholesterol + water > 43,000 atoms total

➤ Single runs:

- 118ns *simulated time* “dark-adapted”,
- ~1μsec “light-adapted” – used 512→1024→2048 →4096 nodes
-- took about a month – **lead to new, experimentally verifiable conjectures**

➤ “Dark ensemble” runs:

- 26 simulations of rhodopsin, each > 100ns,
- used 1024→2048 nodes/trajectory

➤ Thus in aggregate > 3.7 us simulated time on BGW, ~90x previous record at 40 ns.
“Breakthrough in capability for the membrane protein community.”

➤ Currently BGW supports ~1 us of protein folding *simulated time* (single or aggregate runs) in about 2 weeks *elapsed time*.

➤ **BGW is used in a combination of capability** (longer simulated time runs than ever before) **and capacity** (multiple runs on smaller partitions)

BlueGene/L: capability or capacity?

- BlueGene/L architecture optimizes Flop/s /Watt
 - hence Flop/s per rack, per m², per dollar
 - makes a Top-20 machine affordable ...

- BlueGene/L is very modular, and shows excellent scaling (both weak scaling and strong scaling).

- Therefore BlueGene/L machines can be used very well **both in capability mode** (large user partitions) **and in capacity mode** (smaller user partitions).

- **Usage policy issue** – we favor using BGW for its strength: as a capability system.
 - Where we run “ensembles”, the ensembles are not very large
 - each individual run is a capability run in itself!

BGW usage

- We devote 90% of BGW's cycles to 4 production applications, with the following estimated computational requirements:

– A physical simulation	1176 rack-days	= 3.22 BG rack-years = 1.156 billion processor-hours
– Protein folding # 1	900 rack-days	= 2.46 BG rack-years = 0.885 billion processor hours
– Biological simulation	4220 rack-days	= 11.56 BG rack-years = 4.148 billion processor hours
– Protein folding # 2	3440 rack-days	= 9.42 BG rack-years = 3.381 billion processor hours

- Overall CPU cycle usage rate ~ 85% sustained

- **Allocation criteria:**

IBM internal production science proposals

- a. interest to the scientific community
- b. interest to the general population
- c. **whether this problem truly requires BGW or could run on other architectures**

For external proposals (such as DOE Incite program) in addition:

- a. the scientific merits of an application
- b. the merits of the work for improved understanding of computer science
(is this a new class of applications to be run on highly scalable architectures, such as BG ?)

And of course...

projects that support IBM BlueGene business opportunities

Capability metric: time to solution?

- **Well... yes... but what is the solution? What is the problem?**

- Who is asking? (user vs sysadmin vs lab director)
- Fixed size problem ?
 - reduce time to solution with
 - higher frequency -- will be difficult to sustain in future technology
 - Increased parallelism
 - implies that the machine should behave well under **strong scaling**.

.....

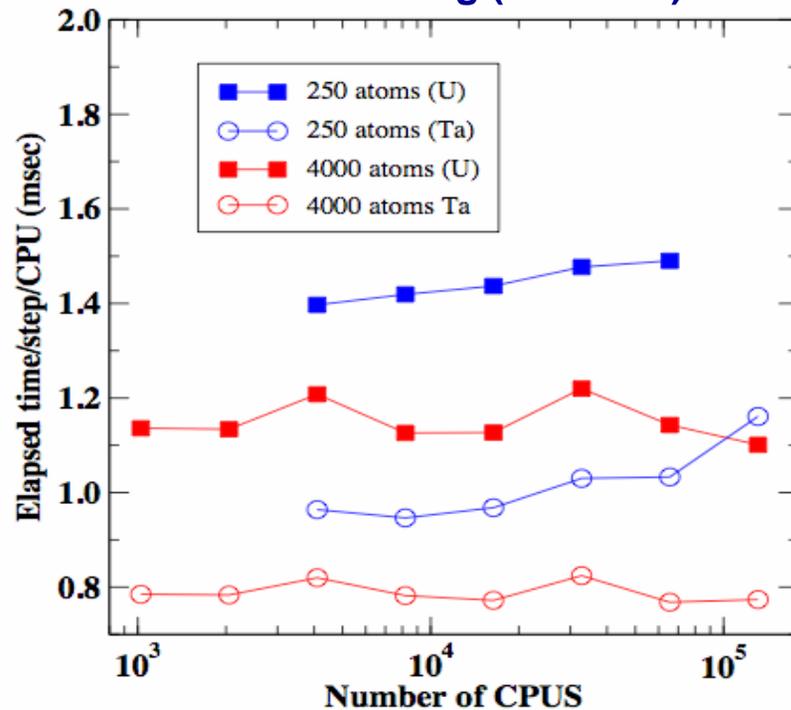
Classical MD – ddcMD (LLNL) -- 2005 Gordon Bell Prize Winner

524 million atom simulations on 64K nodes achieved 101.5 TF/s sustained.

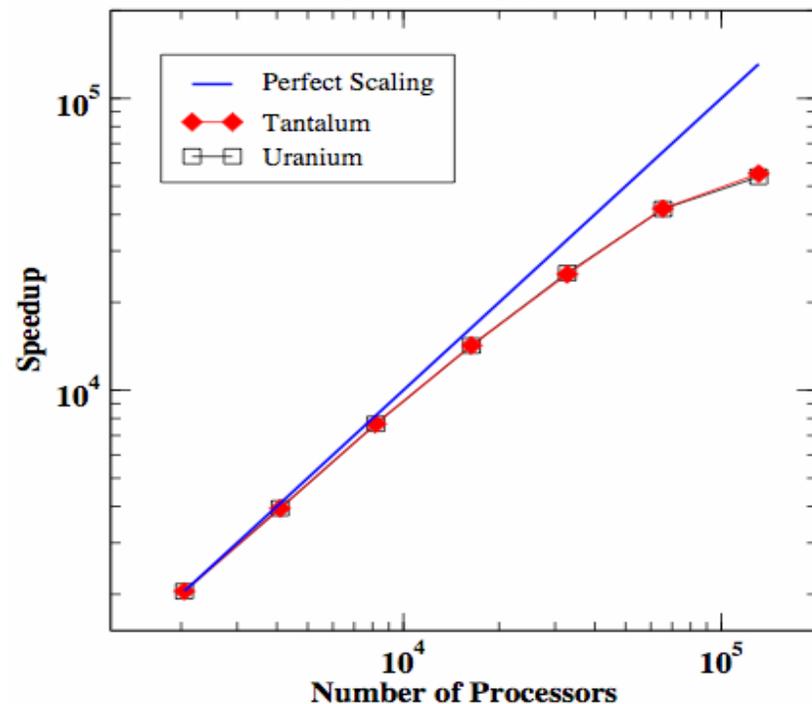
... unprecedented scaling of size or time

- Weak scaling is virtually flat across the entire machine - enables simulation of tens of billions of atoms (roughly a cubic micron of material)
- Strong scaling shows speedup down to 8 atoms/CPU - enables simulations involving millions of steps (typically ns of simulated time)

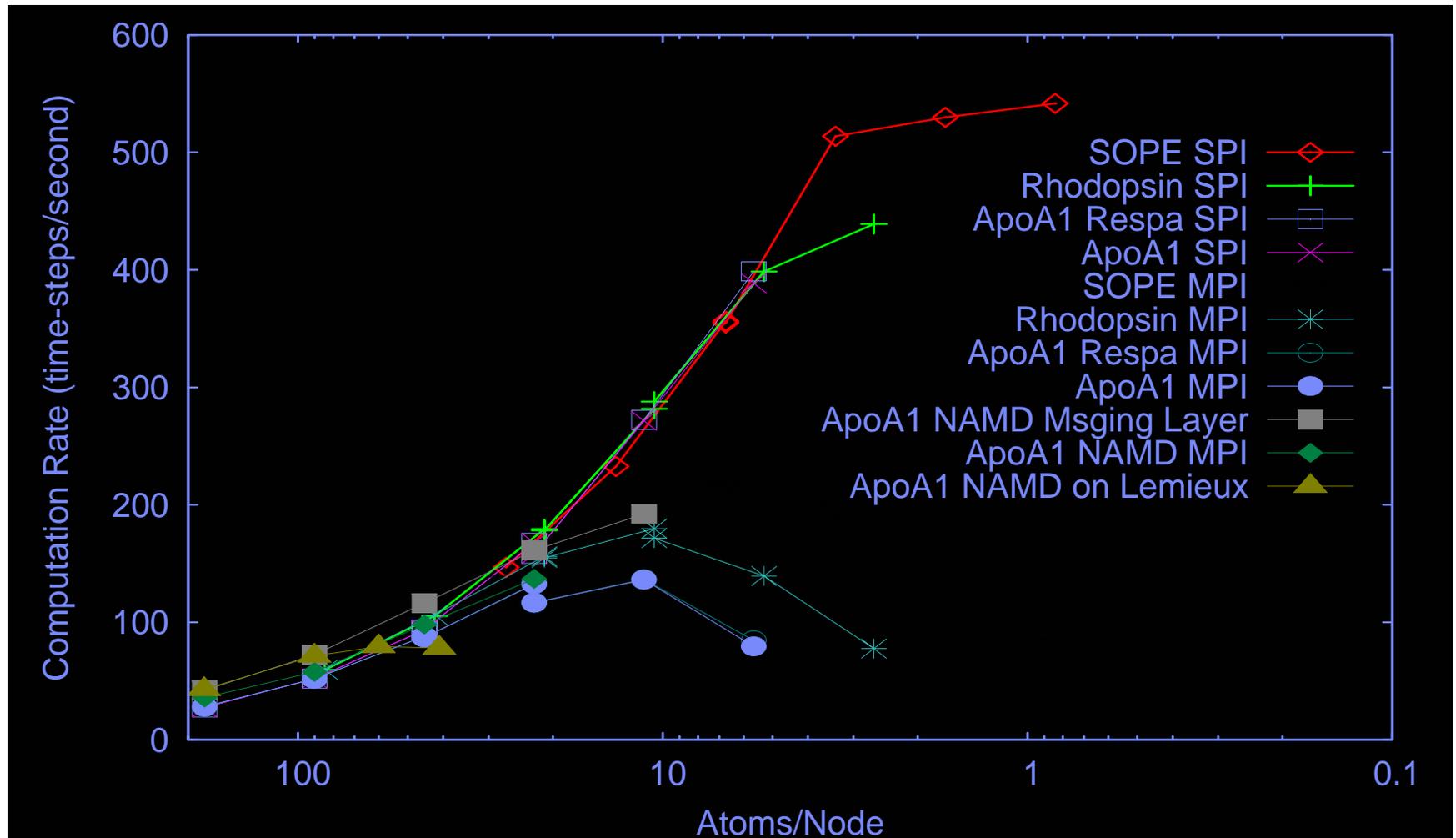
Weak Scaling (Ta and U)



Strong Scaling (Ta and U)



BGW: Strong Scaling Results



Capability metric: time to solution?

.....

So if the machine scales well, users just increase the problem size!

- longer runs (or ensemble runs) for same # of atoms; or
- increase size of problem (more atoms, finer grids – weak scaling)

- Thus, more capability → more science

- Shorter time to solution... users do not go home earlier...
they do more!
And demand more! Demand for BGW is over 2x capacity ...

Capability metric: Flop/s ?

- Science (or bragging rights!) relates to problem size that can be solved in fixed elapsed time.
- Time to solution = time to next publication deadline
= Flop (problem size) / Flop/s (speed)
- Thus, FLOP (problem size) / fixed time
→ ... → **Flop/s (speed of machine)**
- Note that these are real sustained (averaged) Flop/s for the particular problem.
 - such as the real 100+ TFlop/s ddcMD calculation (LLNL 64 rack)
 - Note: “only” 28% FPU utilization (peak speed ~360 TFlop/s) Most applications score less...
 - Peak speed and Linpack speed are important ...
 - ... but they are **not** direct Capability indicators...
 - communication / software / optimization / ... play into the real numbers.
- Thus... HPC Challenge benchmarks (which each focus on different aspects, and are expressed in OPS/s) may provide a better Capability metric ?

HPC Challenge

- **HPL** - the Linpack TPP benchmark which measures the floating point rate of execution for solving a linear system of equations.
- **RandomAccess** - measures the rate of integer random updates of memory (GUPS).
- **FFTE** - measures the floating point rate of execution of double precision complex one-dimensional Discrete Fourier Transform (DFT).
- **STREAM** - a simple synthetic benchmark program that measures sustainable memory bandwidth (in GB/s) and the corresponding computation rate for simple vector kernel.

Benchmark	64-rack BG (optimized)	16-rack BG (optimized)
HPL (TFlop/s)	259.213	67.11
RandomAccess (GUP/s)	35.46	17.29
FFT (GFlop/s)	2311.09	988.18
STREAM Triad (GB/s)	160,064	39,991

← Only 1 TFlop/s, still best in class!

We find that for protein simulation -- mix of short range (direct space) & long range (k-space) interactions -- we get about twice the FFT rate (~ 1.85 TFlop/s on 16 racks).

So that is the appropriate figure of merit for that particular problem!