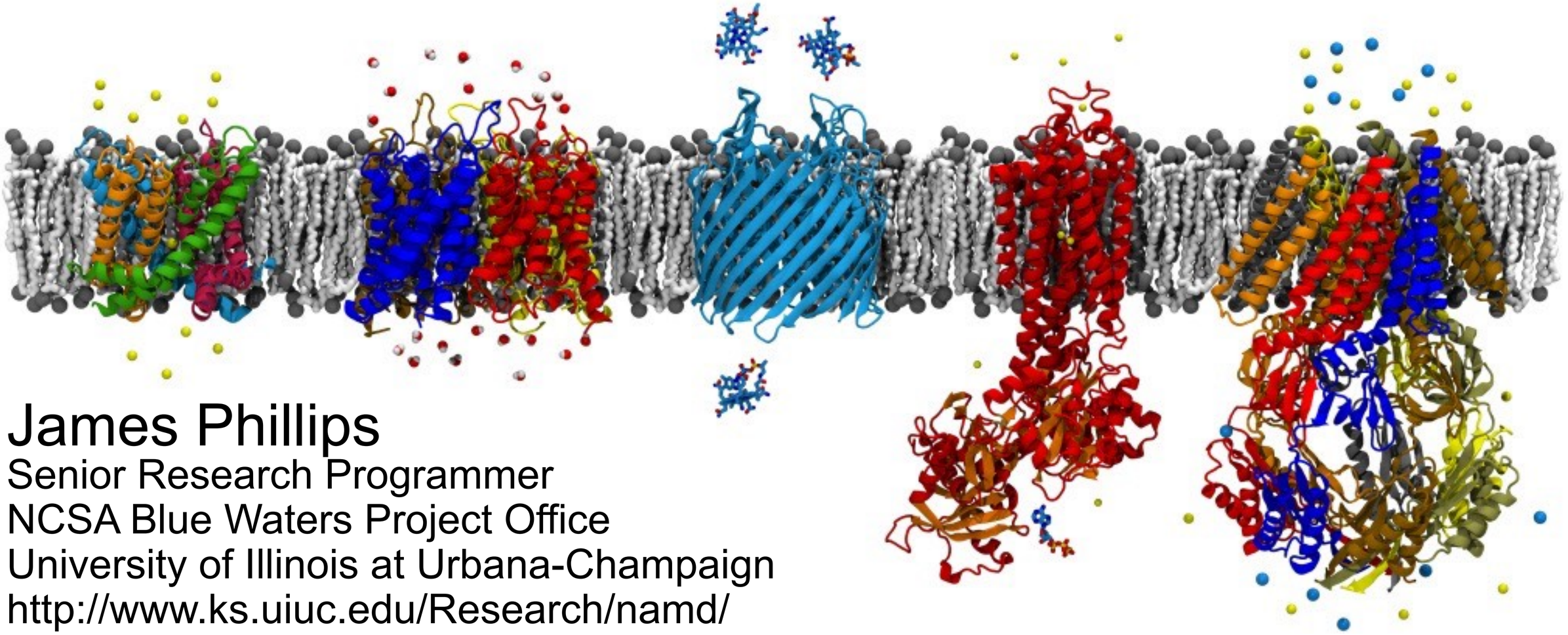
Experiences with Charm++ and NAMD on the Summit POWER9/Volta Supercomputer



James Phillips
Senior Research Programmer
NCSA Blue Waters Project Office
University of Illinois at Urbana-Champaign
http://www.ks.uiuc.edu/Research/namd/

NAMD: Practical Supercomputing for Biomedical Research

"**widest-used application**" on NCSA Blue Waters, NSF-specified benchmark for successor machine

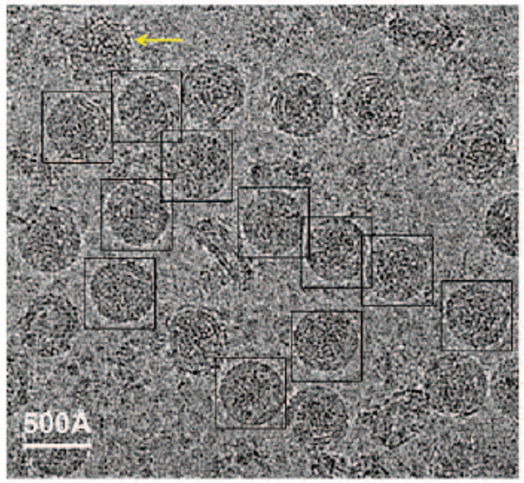
"**by a very large margin the most used code**" at Texas Advanced Computing Center (2nd largest)

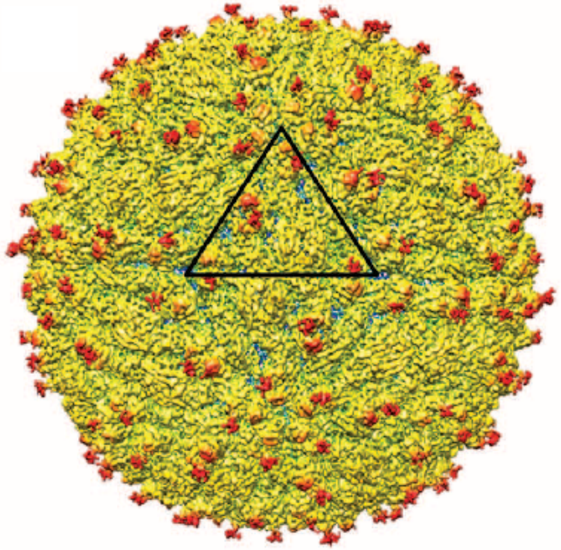
Early adopters of workstation clusters (1993), Linux clusters (1998), and CUDA (**2007**).

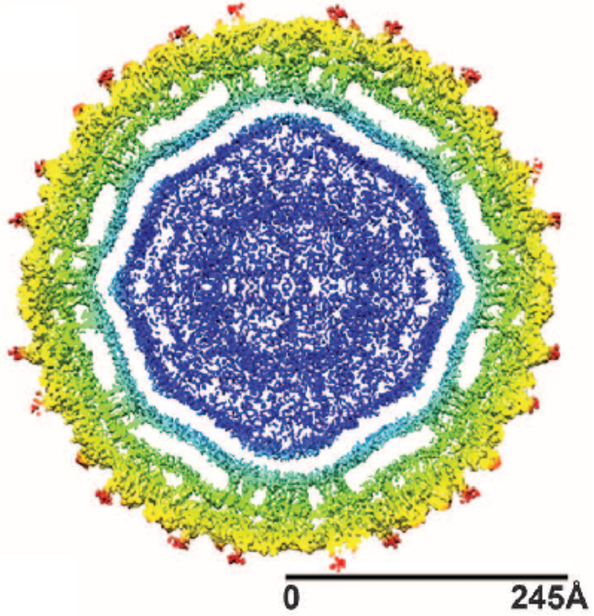
Application readiness/early science projects on
- Argonne Theta (10 PF Cray KNL, completed)
- Oak Ridge Summit (200 PF Power9/Volta, 2018)
- ~~Argonne Aurora (200 PF Cray KNH, 2019)~~
- Argonne Aurora (1 EF Intel Xeon + Xe, 2021)

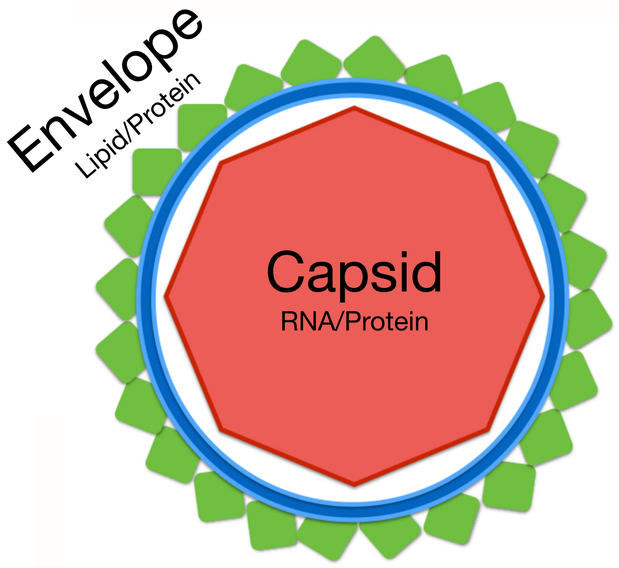
"For outstanding contributions to the development of widely used parallel software for large biomolecular systems simulation"

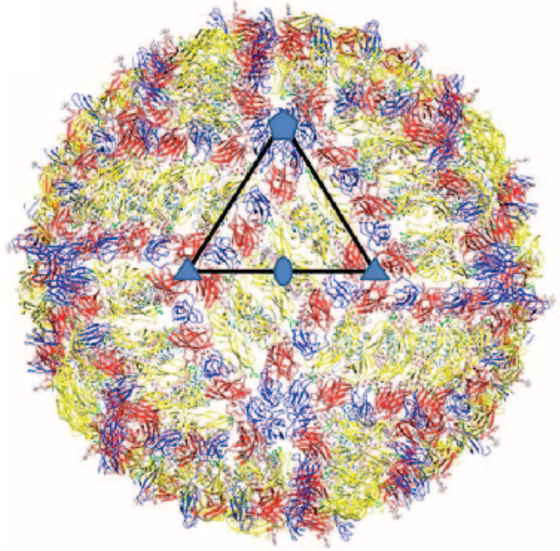
Ultimate Goal of Structural Biology
Construction of High-Resolution Structural Models

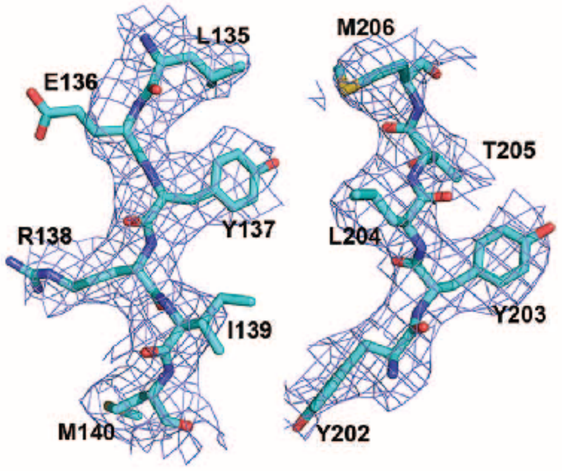










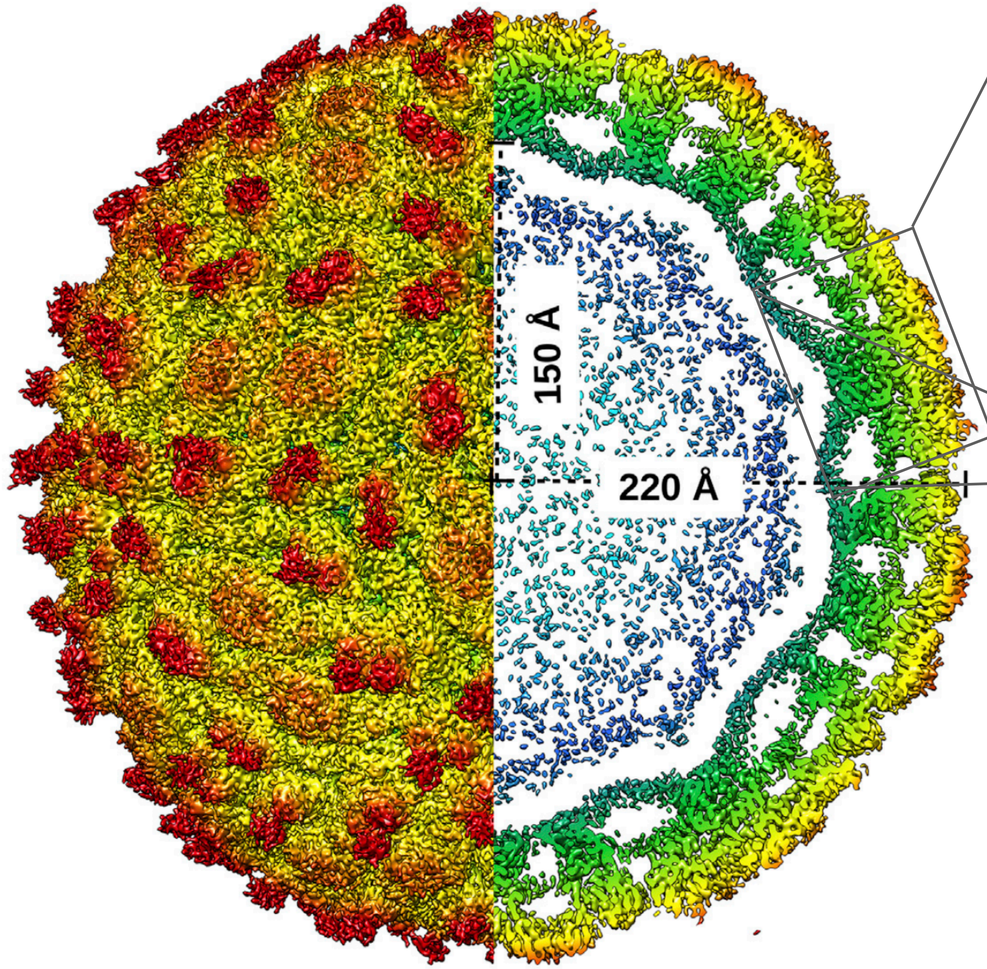


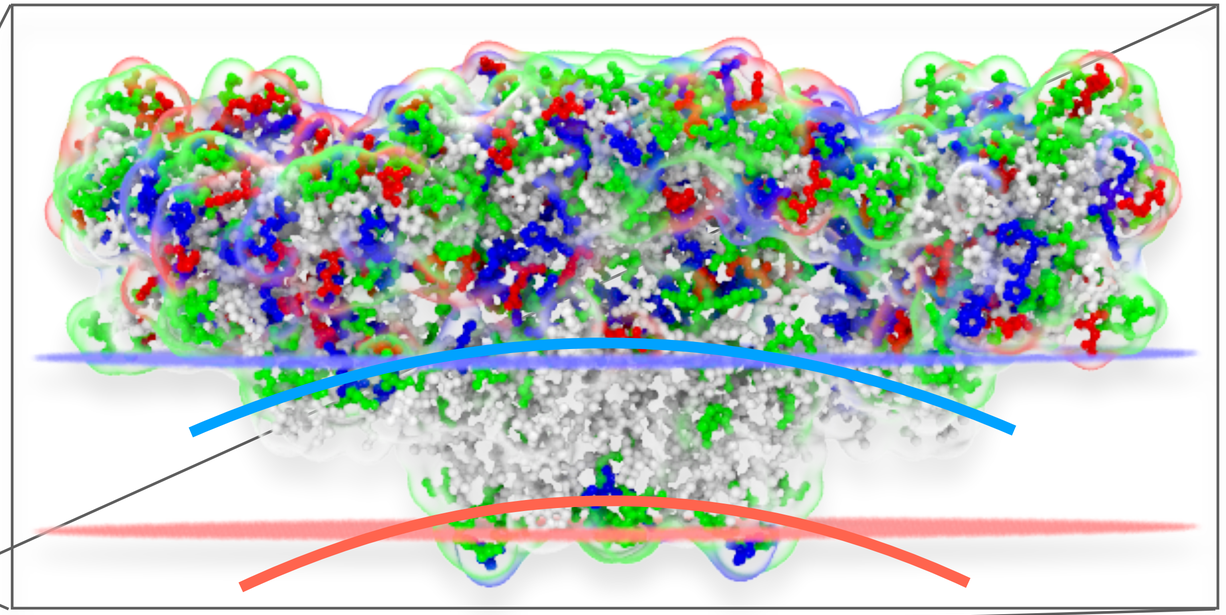
Zika Virus

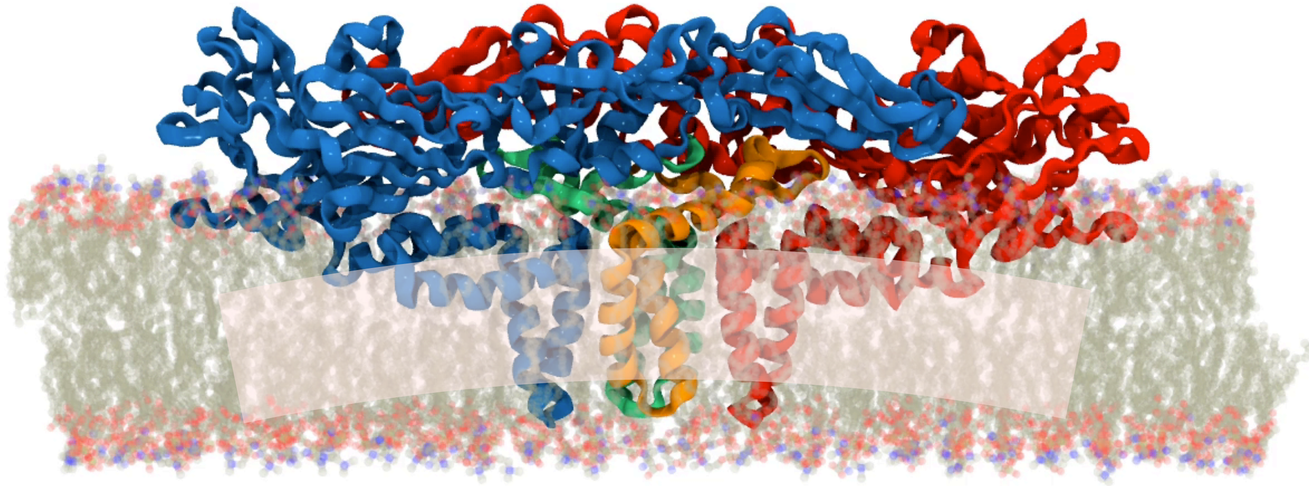
The 3.8 Å resolution cryo-EM structure of Zika virus.
Sirohi, et al., *Science* 352: 467, 2016

Emad Tajkhorshid
Illinois

Highly Localized Membrane Curvature Induced by Deeply Inserted Envelope Proteins

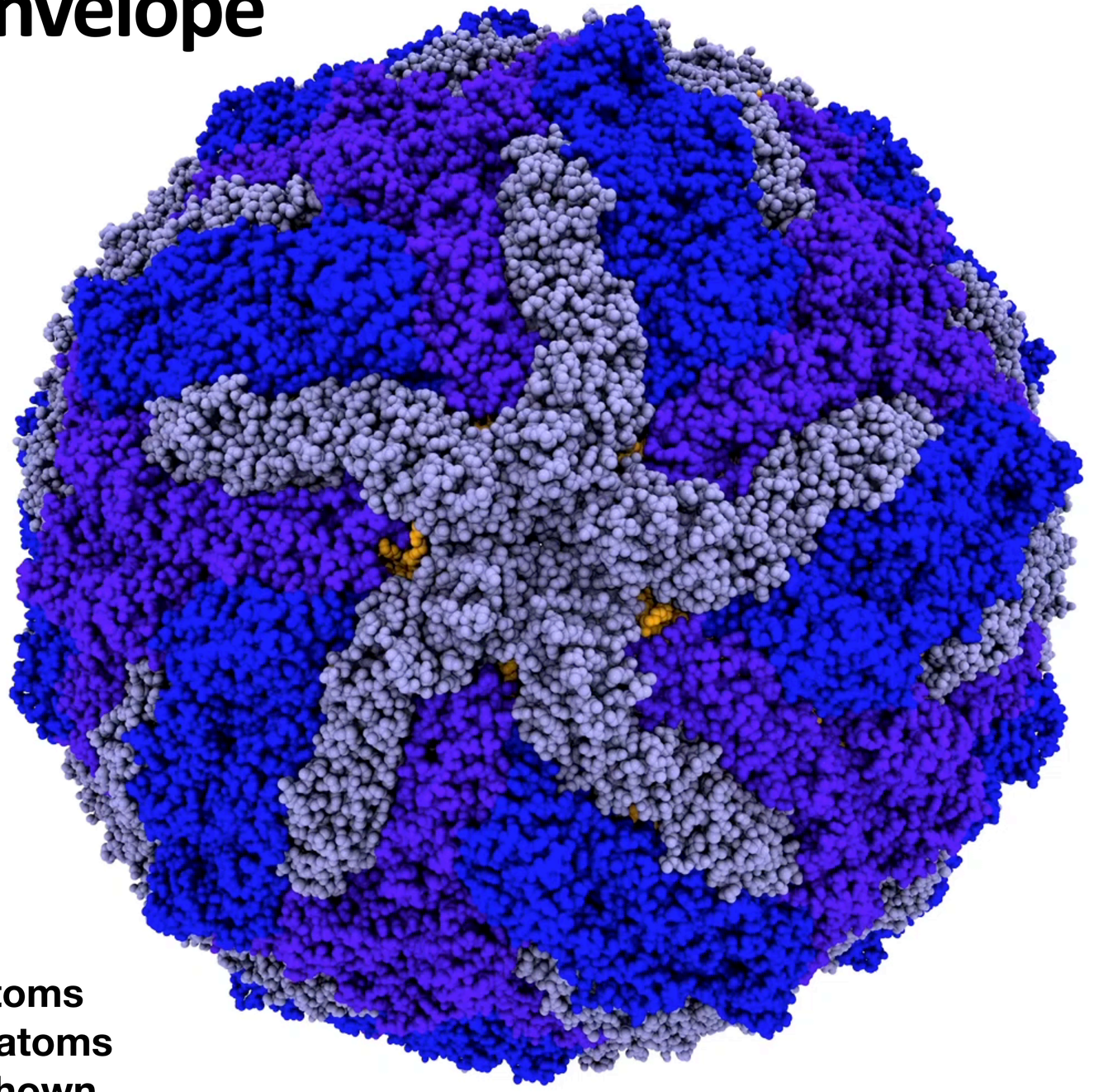






M. Sevvana et al., Refinement and Analysis of the Mature Zika Virus Cryo-EM Structure at 3.1 Å Resolution, *Structure*, Vol. 26, Issue 9, (2018).

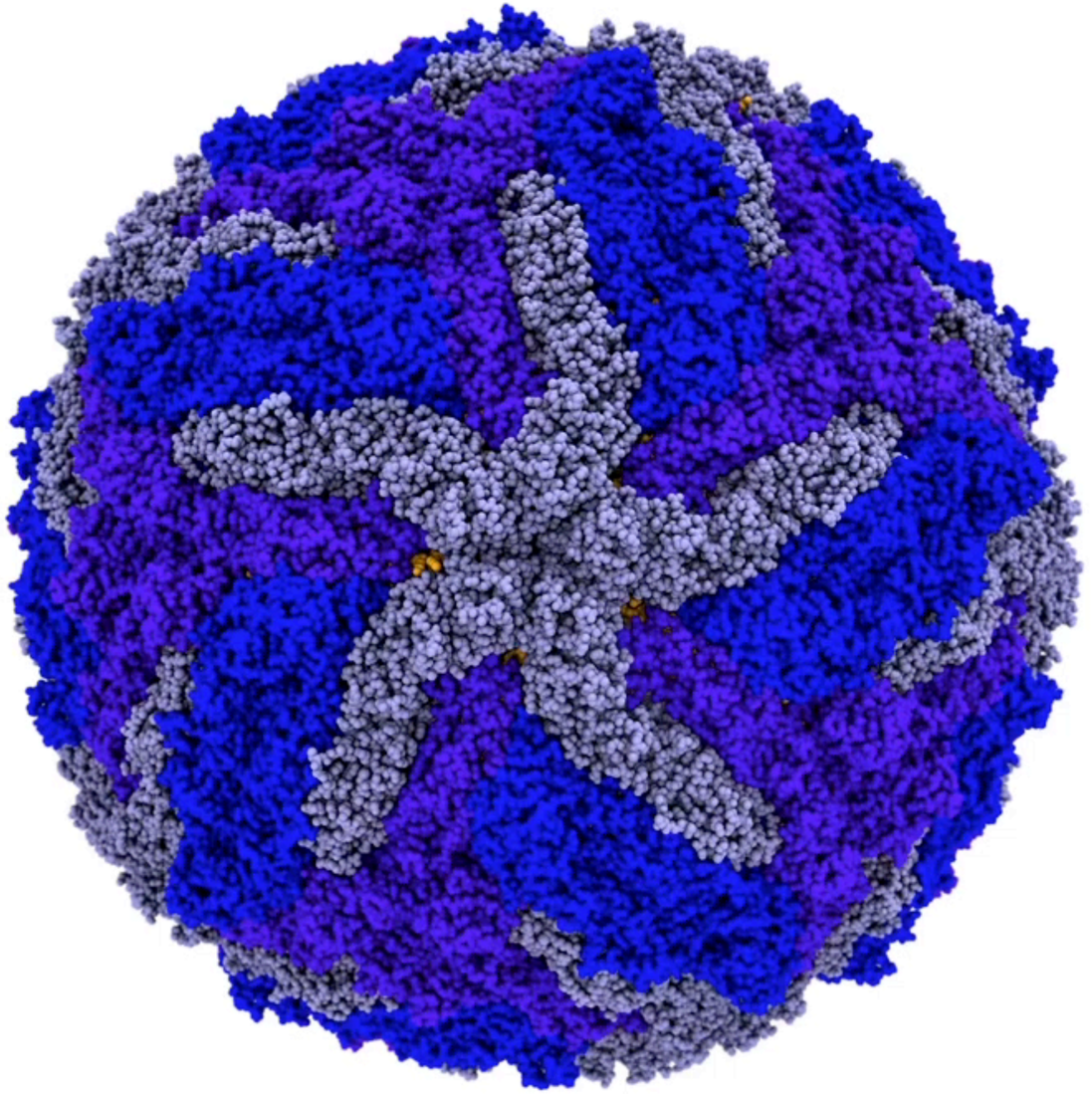
Full Zika Envelope



Envelope: 2.5M atoms
Full System ~ 20M atoms
Solvent/ions not shown

Emad
Tajkhorshid
Illinois

Full Zika Envelope

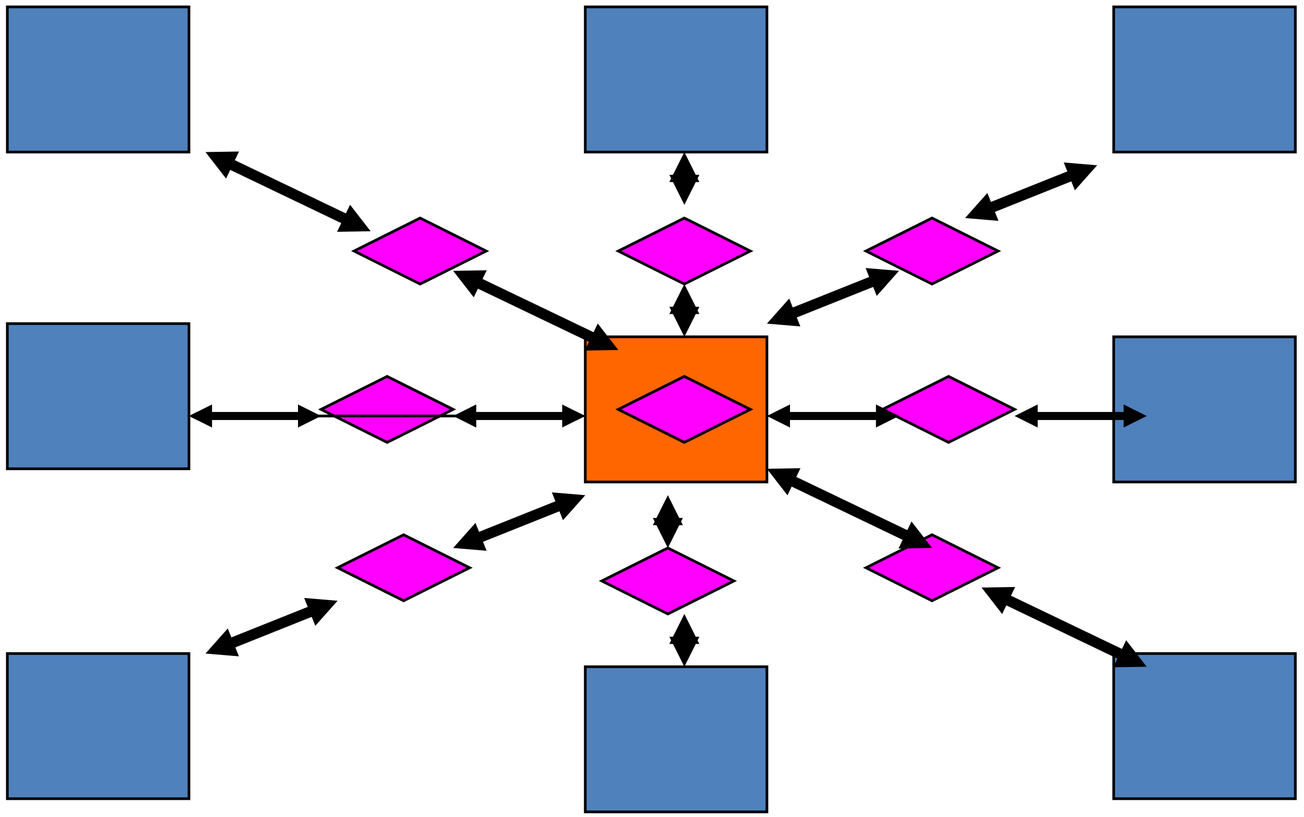


Emad Tajkhorshid
Illinois

Bad setup causes unstable simulation!

NAMD Hybrid Decomposition

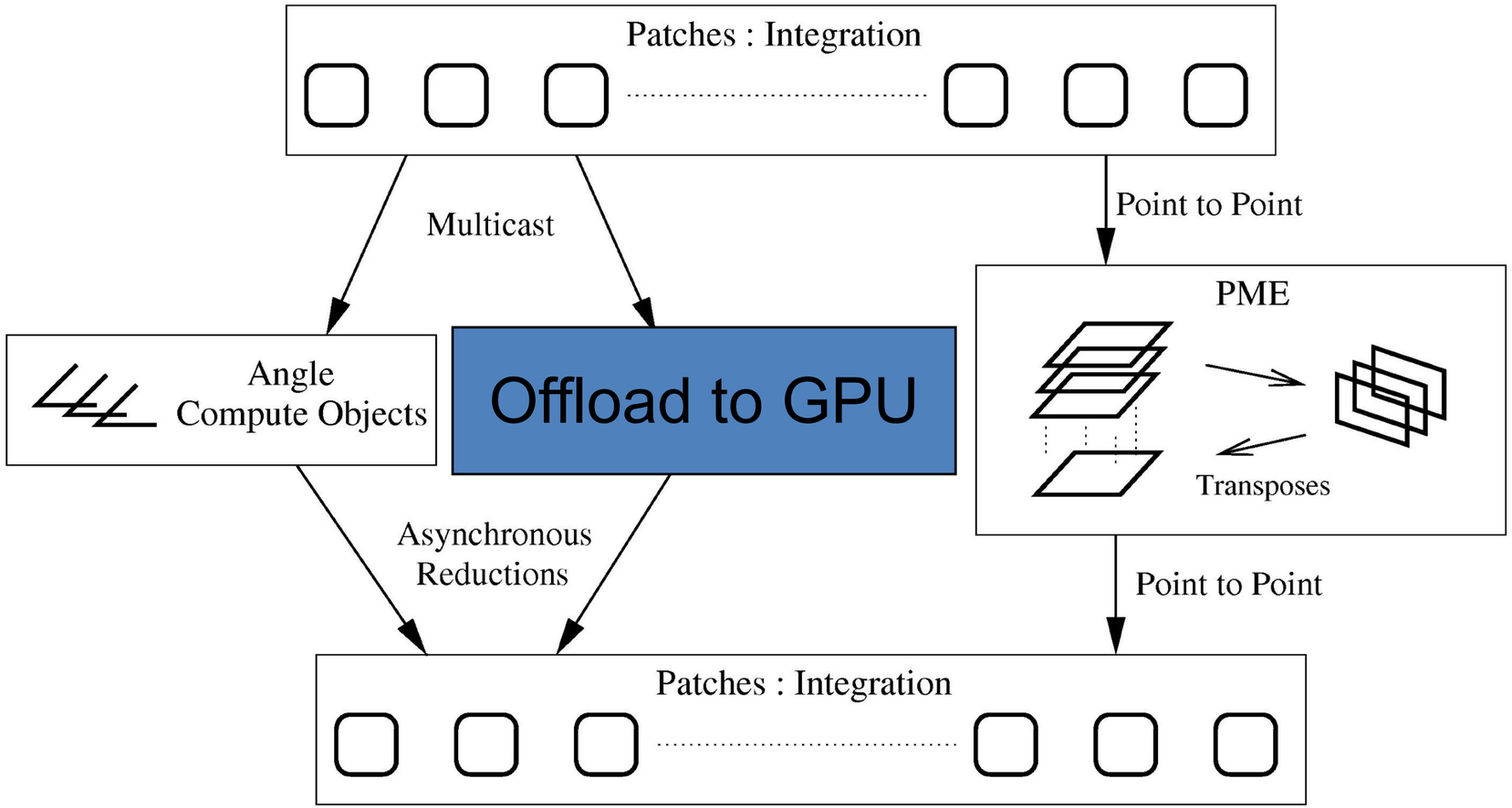
Kale *et al., J. Comp. Phys.* 151:283-312, 1999.



- Spatially decompose data and communication.
- Separate but related work decomposition.
- “Compute objects” facilitate iterative, measurement-based load balancing system.

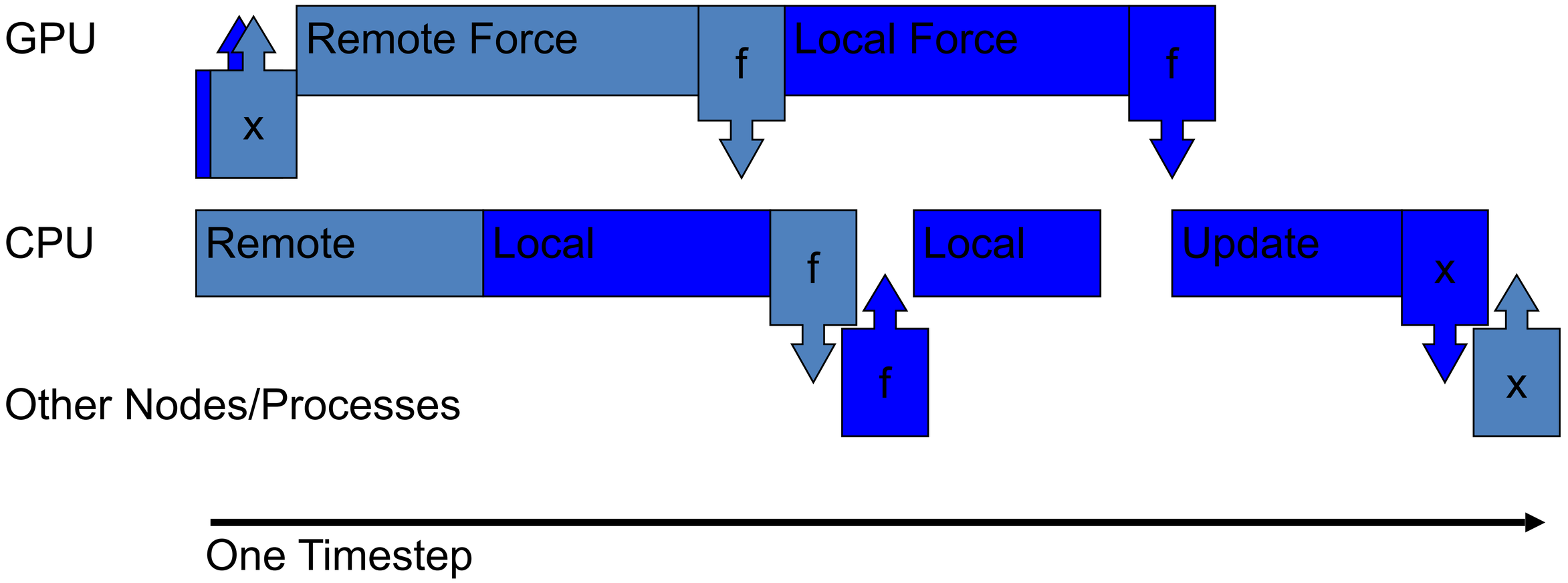
NAMD Overlapping Execution

Phillips *et al.*, *SC2002*.



Objects are assigned to processors and queued as data arrives.

Overlapping GPU and CPU with Communication

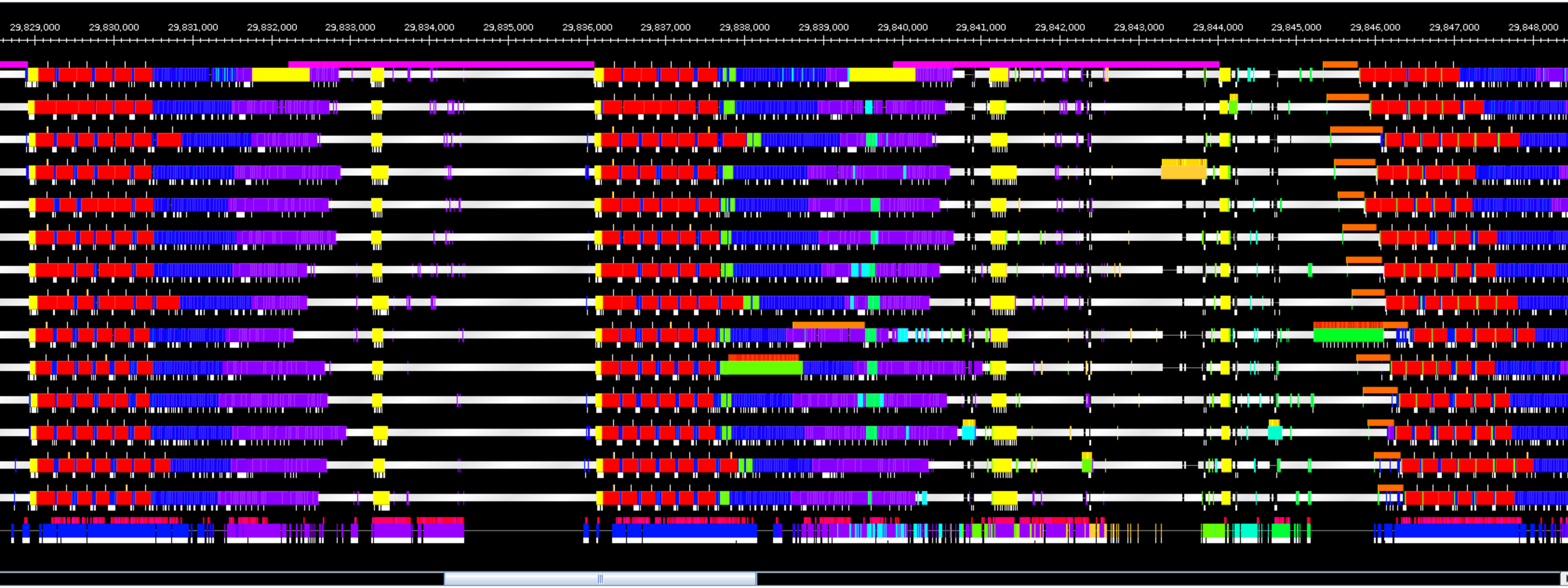


Phillips *et al.*, SC2008

Streaming GPU Results to CPU

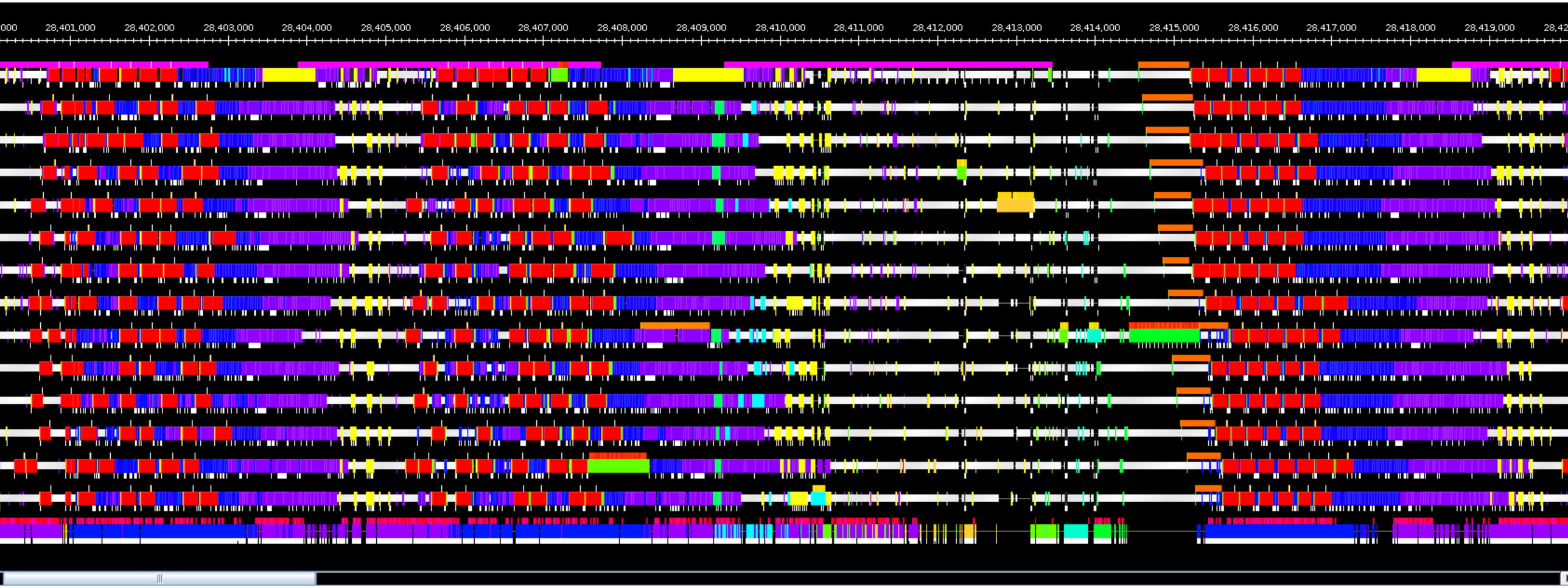
- Allows incremental results from a single grid to be processed on CPU before grid finishes on GPU
- Allows merging and prioritizing of remote and local work
- GPU side:
  - Write results to host-mapped memory (also without streaming)
  - __threadfence_system() and __syncthreads()
  - Atomic increment for next output queue location
  - Write result index to output queue
- CPU side:
  - Poll end of output queue (int array) in host memory

Non-Streaming Kernel



Charm++ *Projections* performance-analysis tool

Streaming Kernel



Charm++ *Projections* performance-analysis tool

Summit will replace Titan as the OLCF's leadership supercomputer

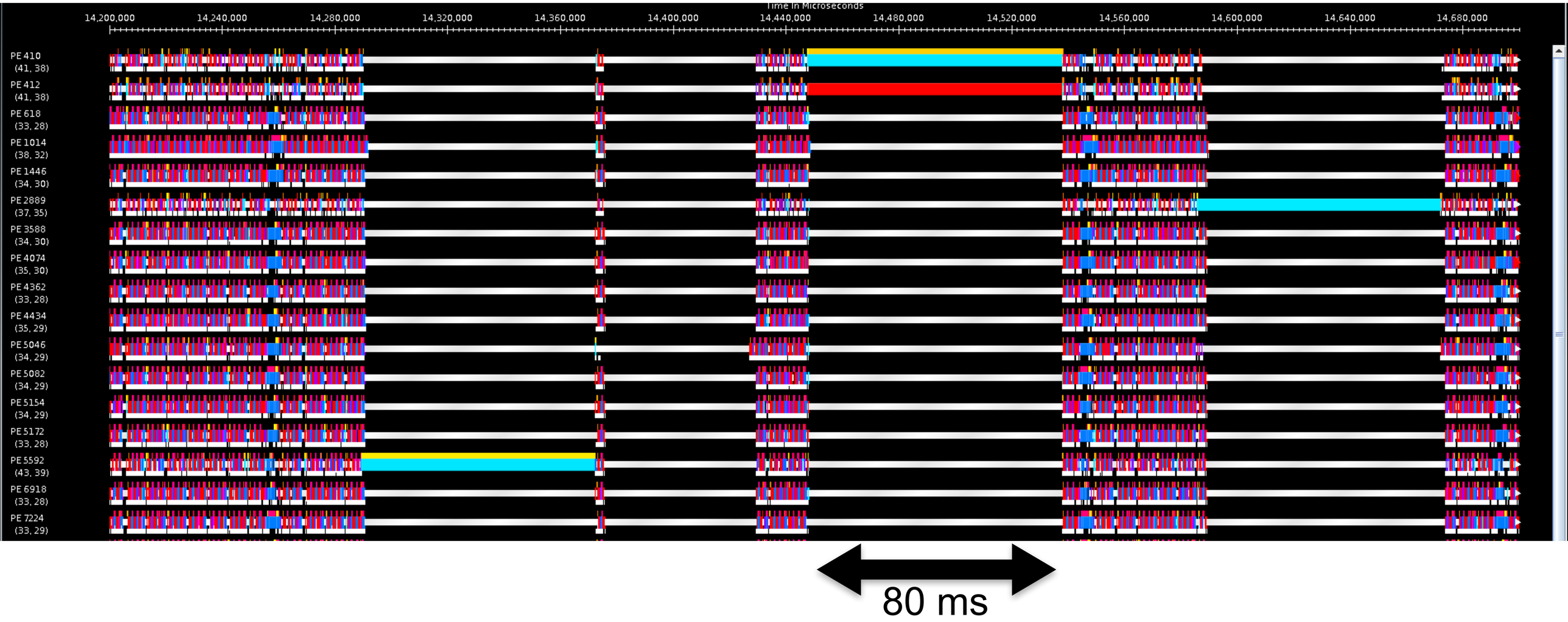
- Many fewer nodes
- Much more powerful nodes
- Much more memory per node and total system memory
- Faster interconnect
- Much higher bandwidth between CPUs and GPUs
- Much larger and faster file system

Feature	Titan	Summit
Application Performance	Baseline	5-10x Titan
Number of Nodes	18,688	~4,600
Node performance	1.4 TF	> 40 TF
Memory per Node	32 GB DDR3 + 6 GB GDDR5	512 GB DDR4 + HBM
NV memory per Node	0	1600 GB
Total System Memory	710 TB	>10 PB DDR4 + HBM + Non-volatile
System Interconnect (node injection bandwidth)	Gemini (6.4 GB/s)	Dual Rail EDR-IB (23 GB/s)
Interconnect Topology	3D Torus	Non-blocking Fat Tree
Processors	1 AMD Opteron™ 1 NVIDIA Kepler™	2 IBM POWER9™ 6 NVIDIA Volta™
File System	32 PB, 1 TB/s, Lustre®	250 PB, 2.5 TB/s, GPFS™
Peak power consumption	9 MW	15 MW

NAMD 2.13 released Nov 9

- First release since December 2016, many improvements
- All force calculation now done on GPU
- CUDA 9 and Volta compatibility
- IBM PAMI SMP machine layer
- Support for two-billion-atom simulations
- New constant pH, improved QM-MM
- Improved core binding of CUDA CPU threads
- Improved CUDA error reporting, **print hostname on Cray**

2018: Summit has a noise problem - now fixed!



2018 Charm++/NAMD configuration

- IBM PAMI SMP machine layer
  - Initially developed for Blue Gene series
  - No dedicated communication thread
- Single GPU per process (6 processes per node, 6 threads per process)
  - Leaving one core free per resource set seems to reduce noise
  - One core per socket is reserved by jsrun, so 8 unused cores per node
- With thread to core affinity:
  - jsrun -r6 -g1 -c7 namd2 +ignoresharing +ppn 6 +pemap 4-27:4,32-55:4,60-83:4,92-115:4,120-143:4,148-171:4
- Or without (expected to run slower, but sometimes faster):
  - jsrun --bind rs -r6 -g1 -c7 namd2 +ignoresharing +ppn 6

2019 Charm++/NAMD configuration

- IBM PAMI SMP machine layer
  - Initially developed for Blue Gene series
  - No dedicated communication thread
- Single GPU per process (6 processes per node, ~~6~~ **7 threads per process**)
  - ~~Leaving one core free per resource set seems to reduce noise~~
  - One core per socket is reserved by jsrun, so ~~8~~ **2 unused cores per node**
- With thread to core affinity (plus resource-set binding for CUDA thread):
  - jsrun **--bind rs -a1** -r6 -g1 -c7 namd2 +ignoresharing **+ppn 7 +pemap 0-83:4,88-171:4** ~~4-27:4,32-55:4,60-83:4,92-115:4,120-143:4,148-171:4~~
- ~~Or without (expected to run slower, but sometimes faster):~~
  - ~~jsrun --bind rs -r6 -g1 -c7 namd2 +ignoresharing +ppn 6~~

“Words of wisdom and comfort on the loss of 90% of your supercomputer performance”

or

“When bad OS updates happen to good scientific applications”

Helpful Activities

- DON’T PANIC
- Recompile
- Try MPI instead of PAMI communication layer
- Report issue to user support
- Periodically ask for updates
- Escalate at every opportunity
- Allow unaffected multi-copy early science to run

Neutral Activities

- Blame <vendor>
- Curse <vendor>
- Wonder if this is related to your contact leaving
- Hope she wasn't the only one who knows the code
- "Not my circus, not my monkeys."
- "No, I will not fix your supercomputer."
- Update Charm++ to bleeding edge…

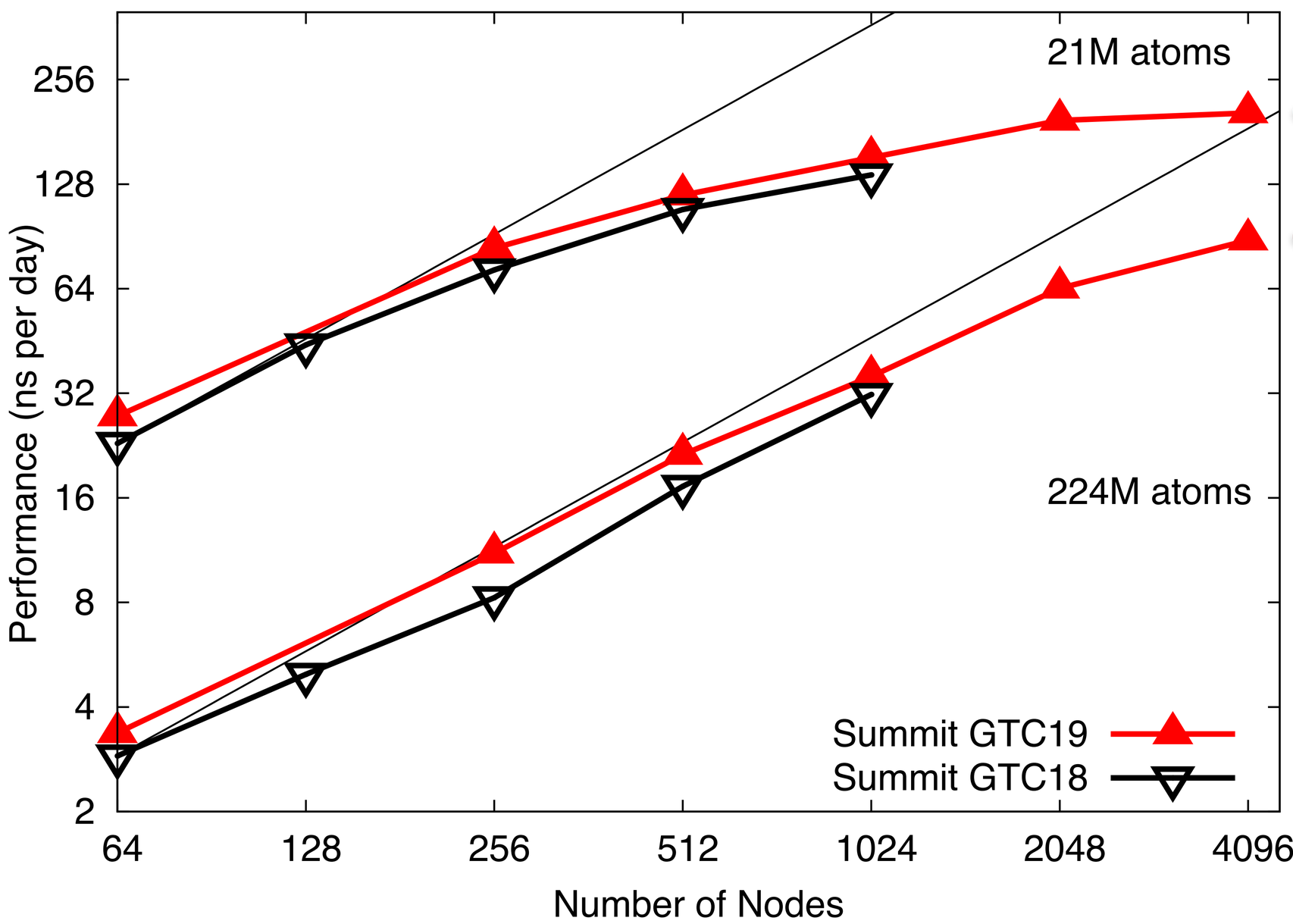
Unhelpful Activities

- Forget you updated Charm++
- Blame instability with new Charm++ on compiler
- Change integrator build flag to -O0 as workaround
- Forget you changed build flag to -O0
- When <vendor> fixes PAMI library, don’t check performance until Friday before GTC
- Fantasize about throwing <vendor> under bus

Helpful Activities (2)

- Remember -O0 change to integrator
- Realize binary from November works fine now
- Notice compiler from November is still available
- Notice compiler from November doesn’t work now
- Realize that Charm++ from November works
- “git log src/archpami-linux-ppc64le”
- “git revert …”

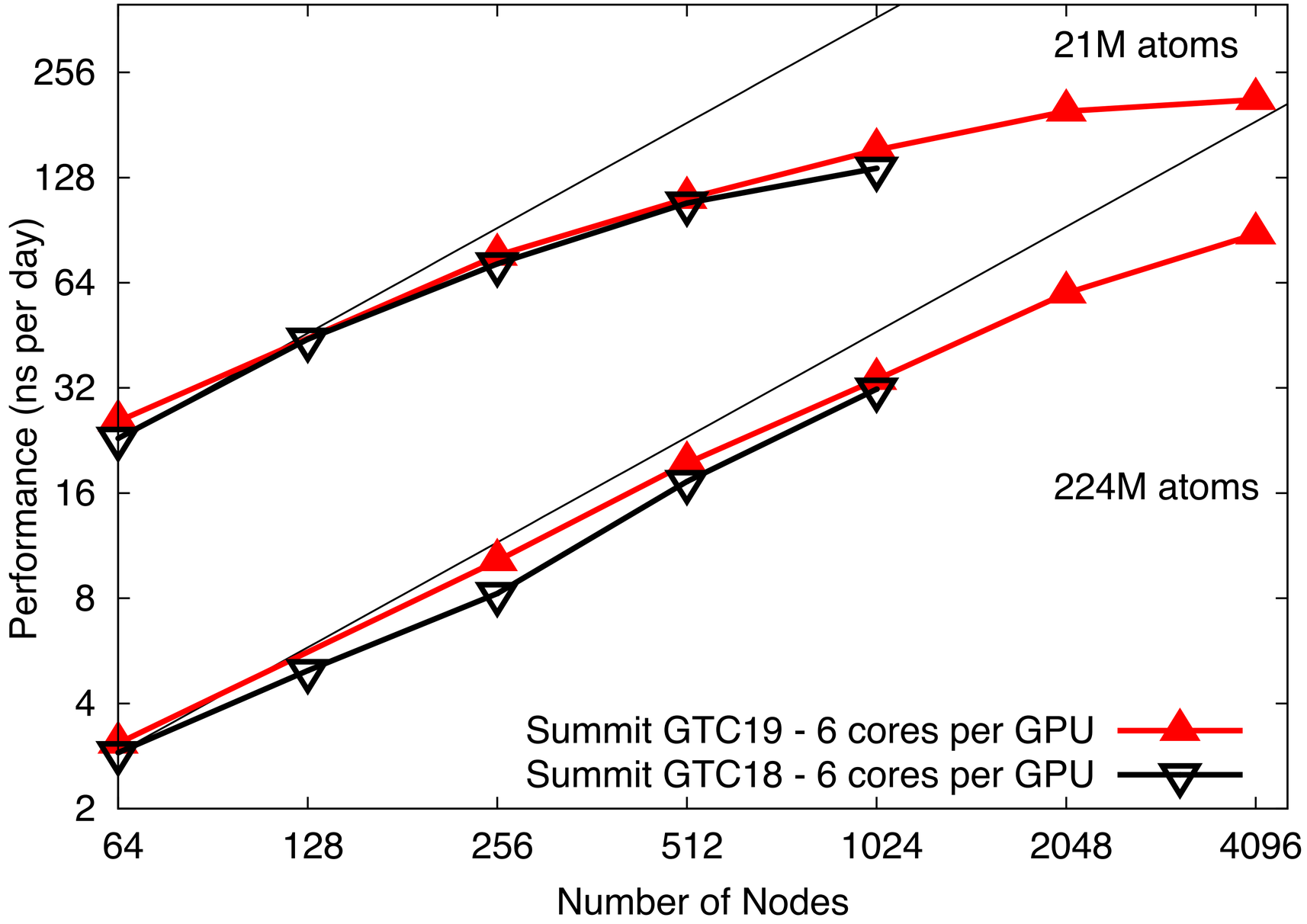
Comparison vs 2018



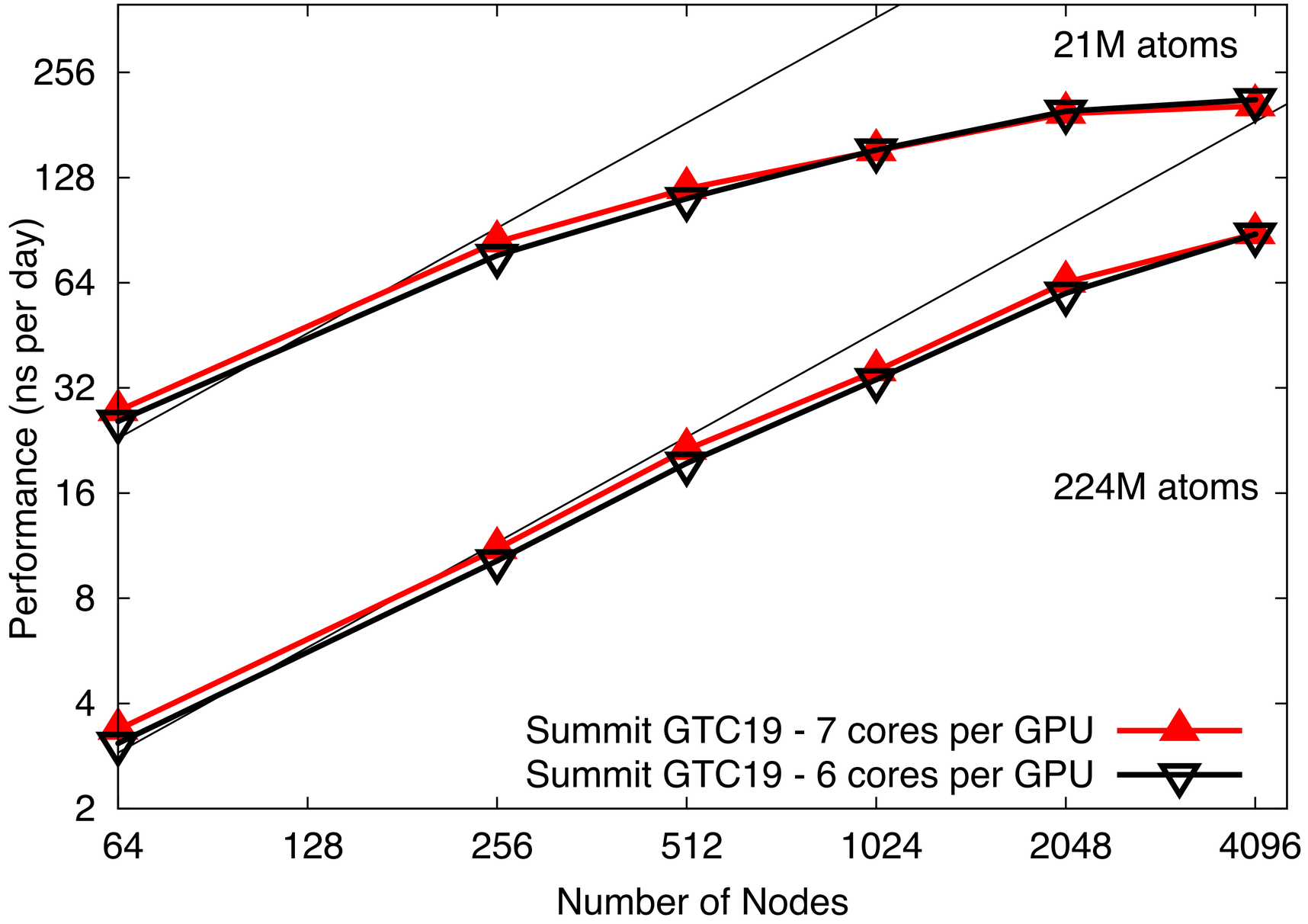
0.8 ms/step
210 ns/day

2.0 ms/step
90 ns/day

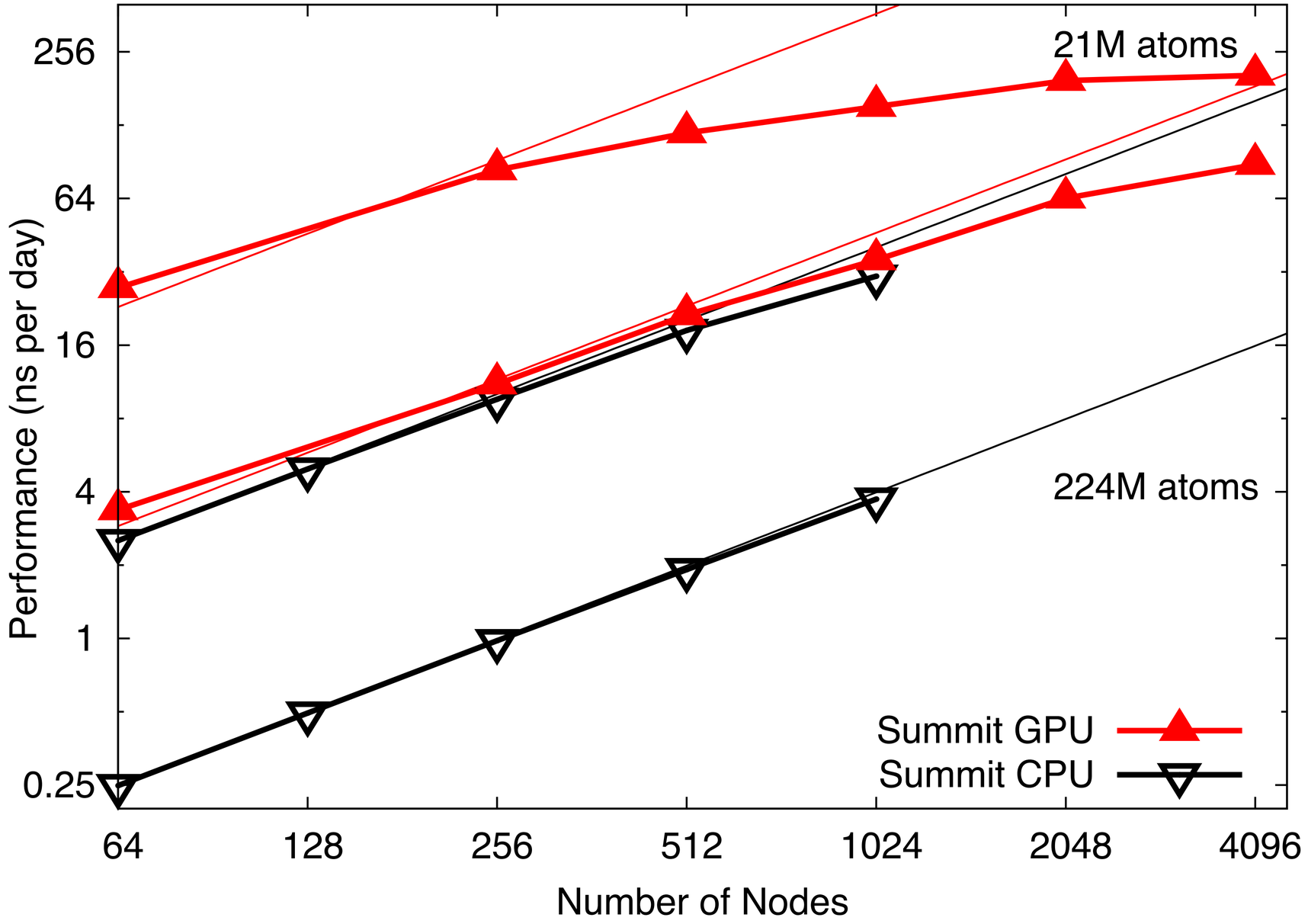
Fairer Comparison vs 2018



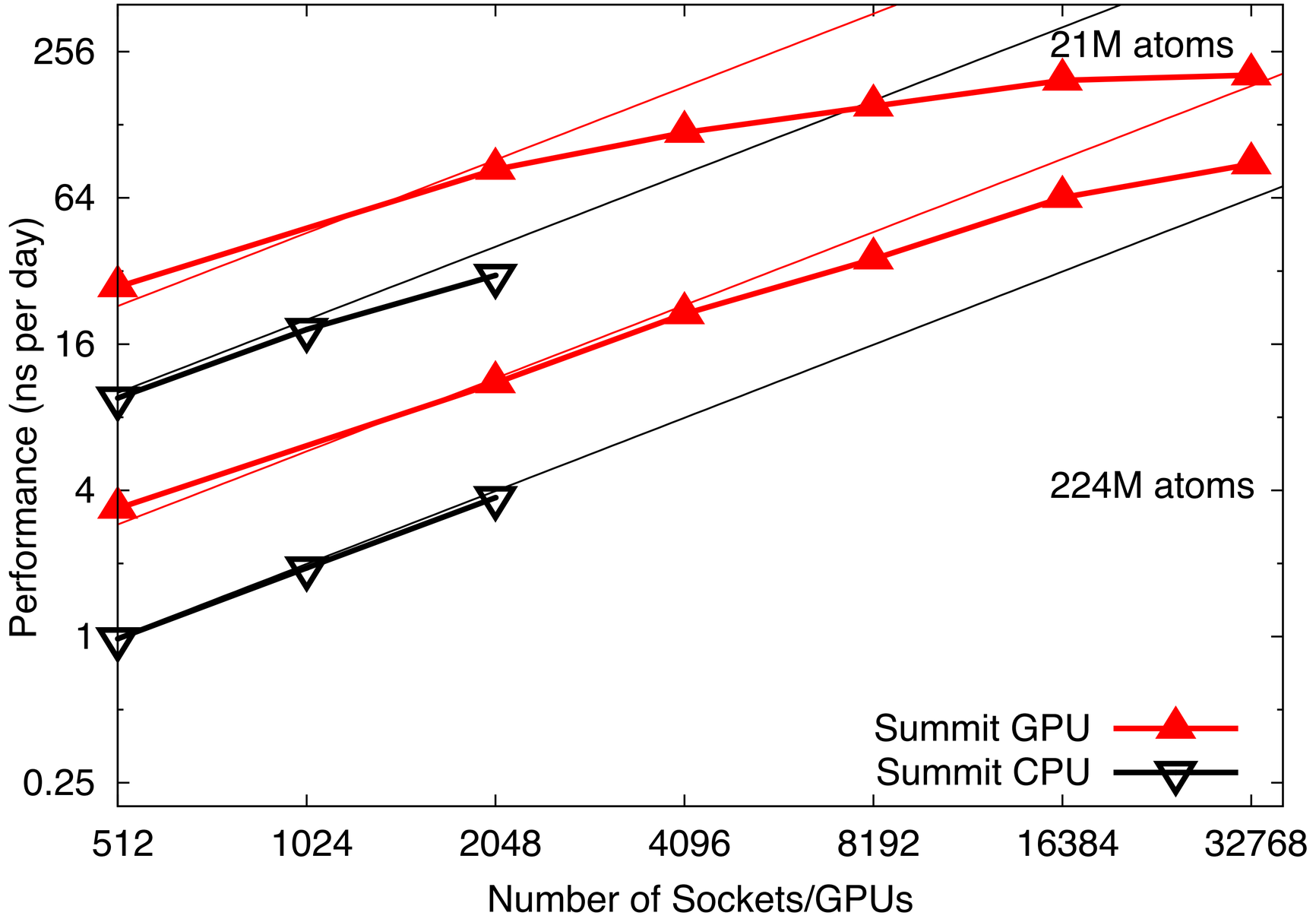
Comparison 7 vs 6 Cores per GPU



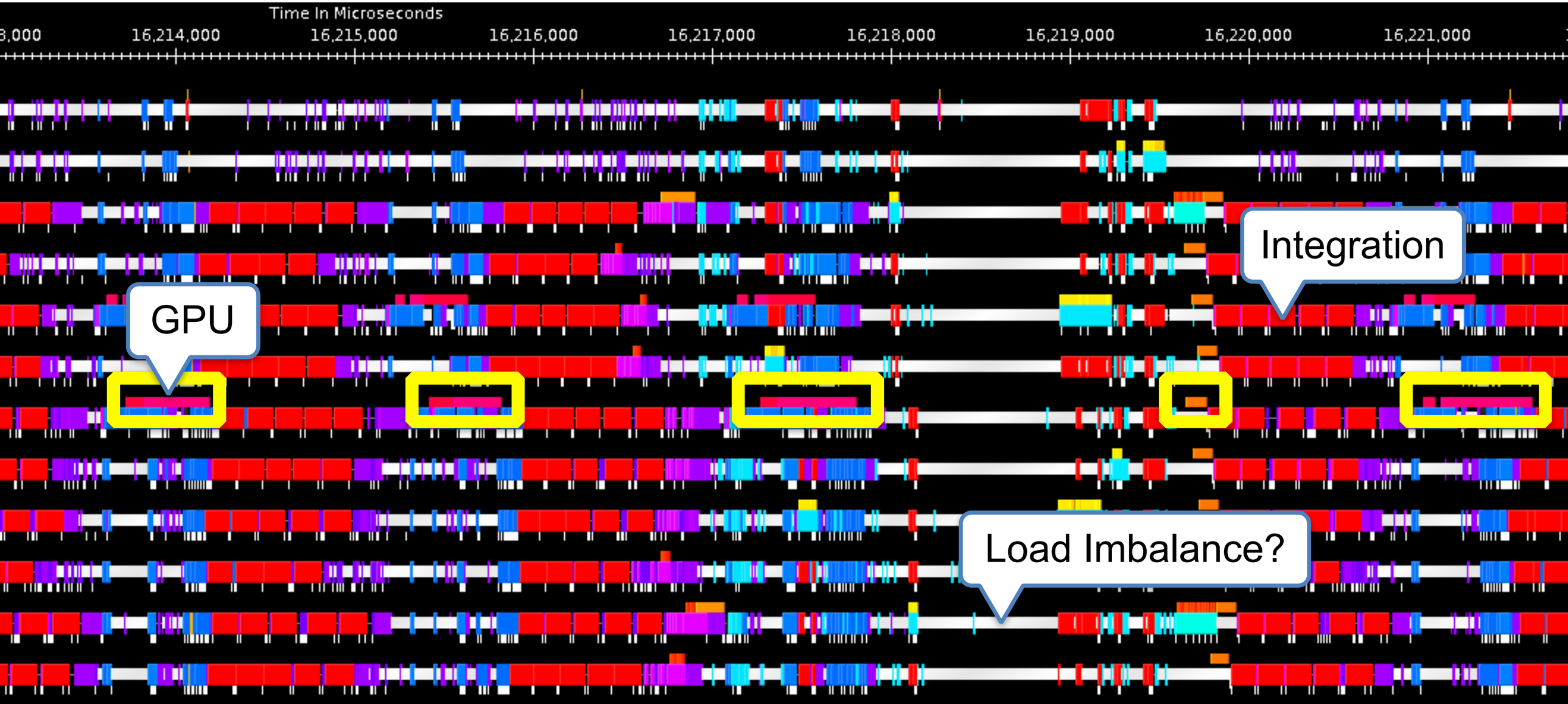
CPU-GPU comparison for large benchmarks



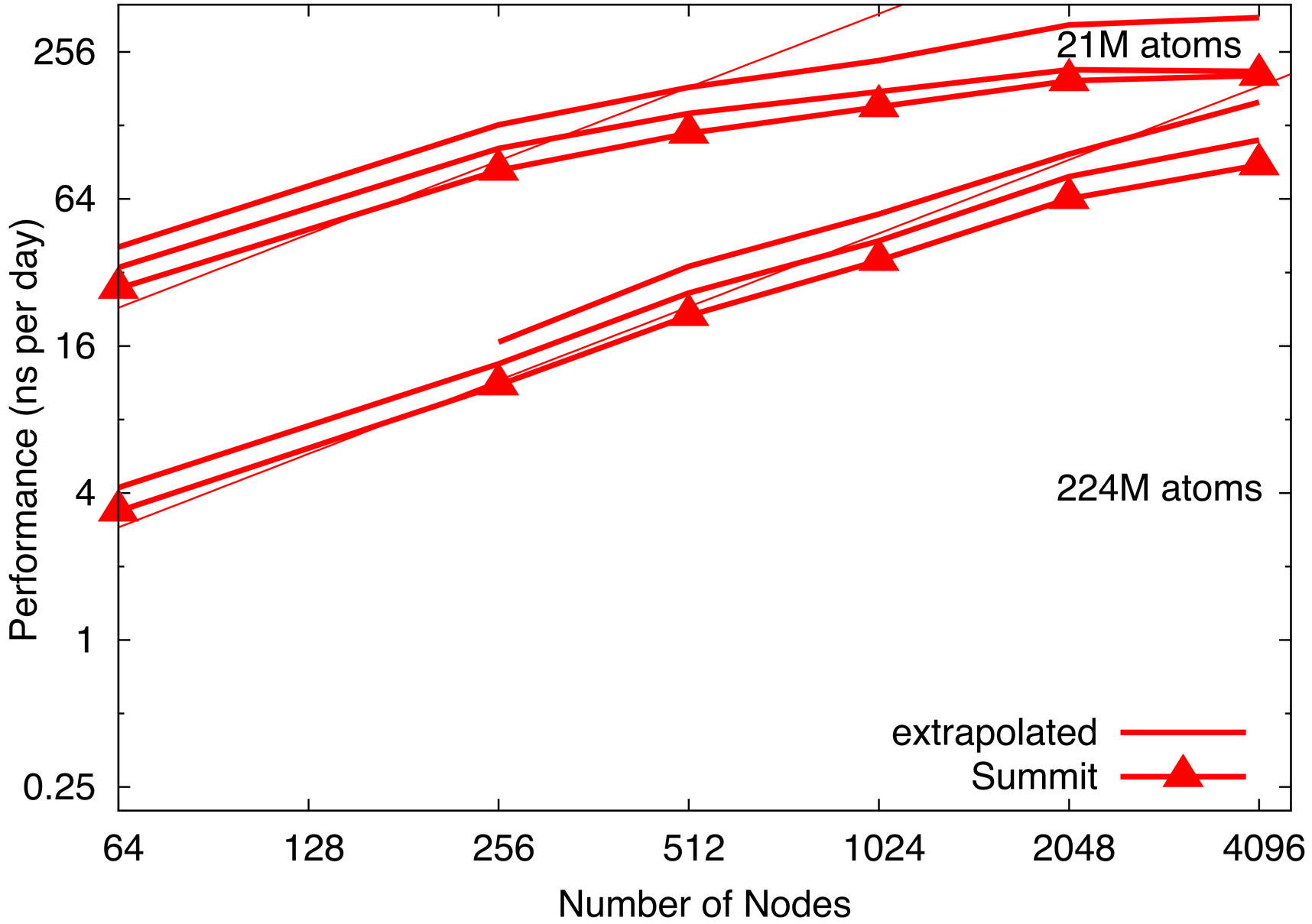
“Fair” comparison for large benchmarks



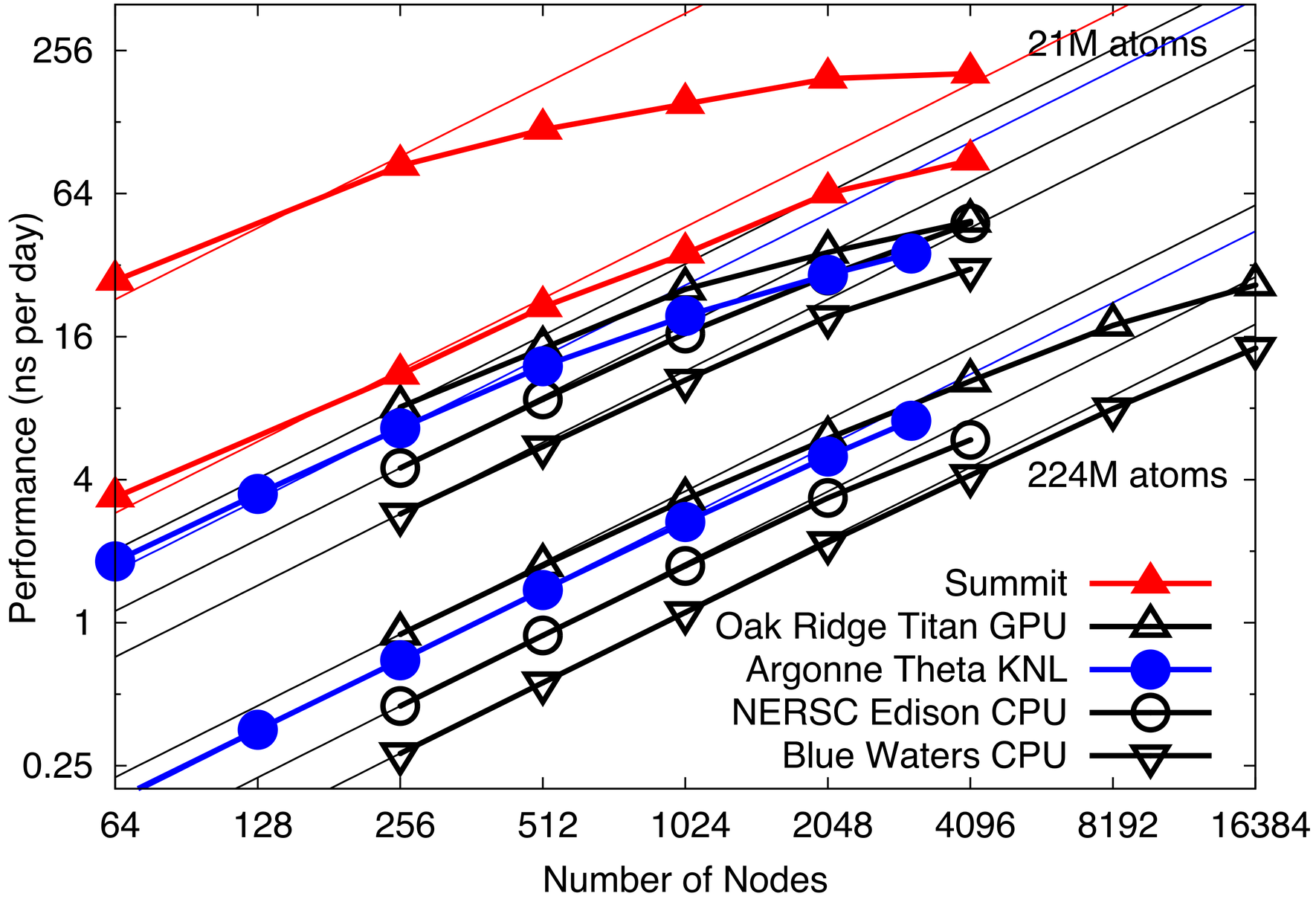
Charm++ *Projections* tool shows bottlenecks



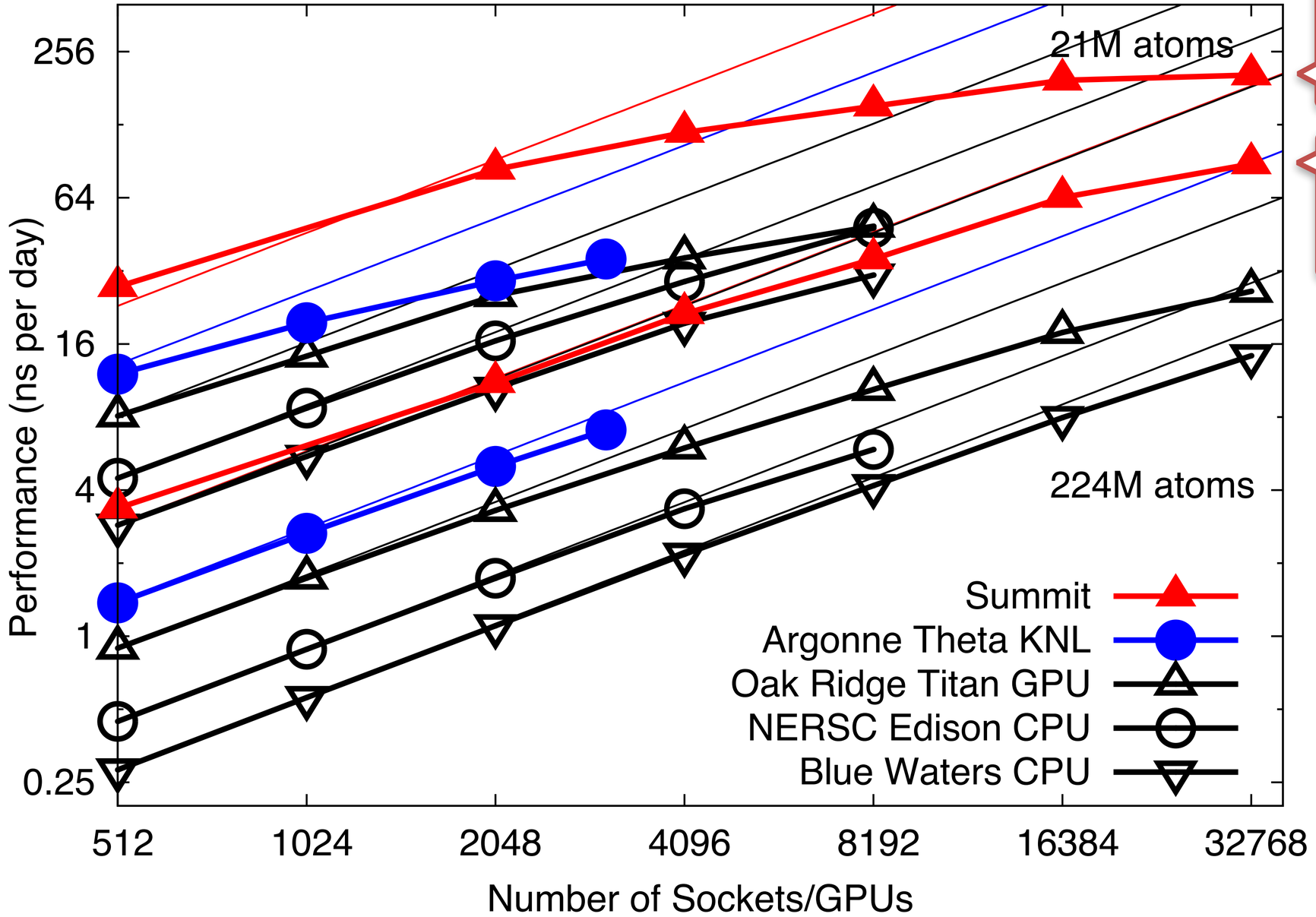
“Fix” problems with simpler integrator



Machine comparison for large benchmarks



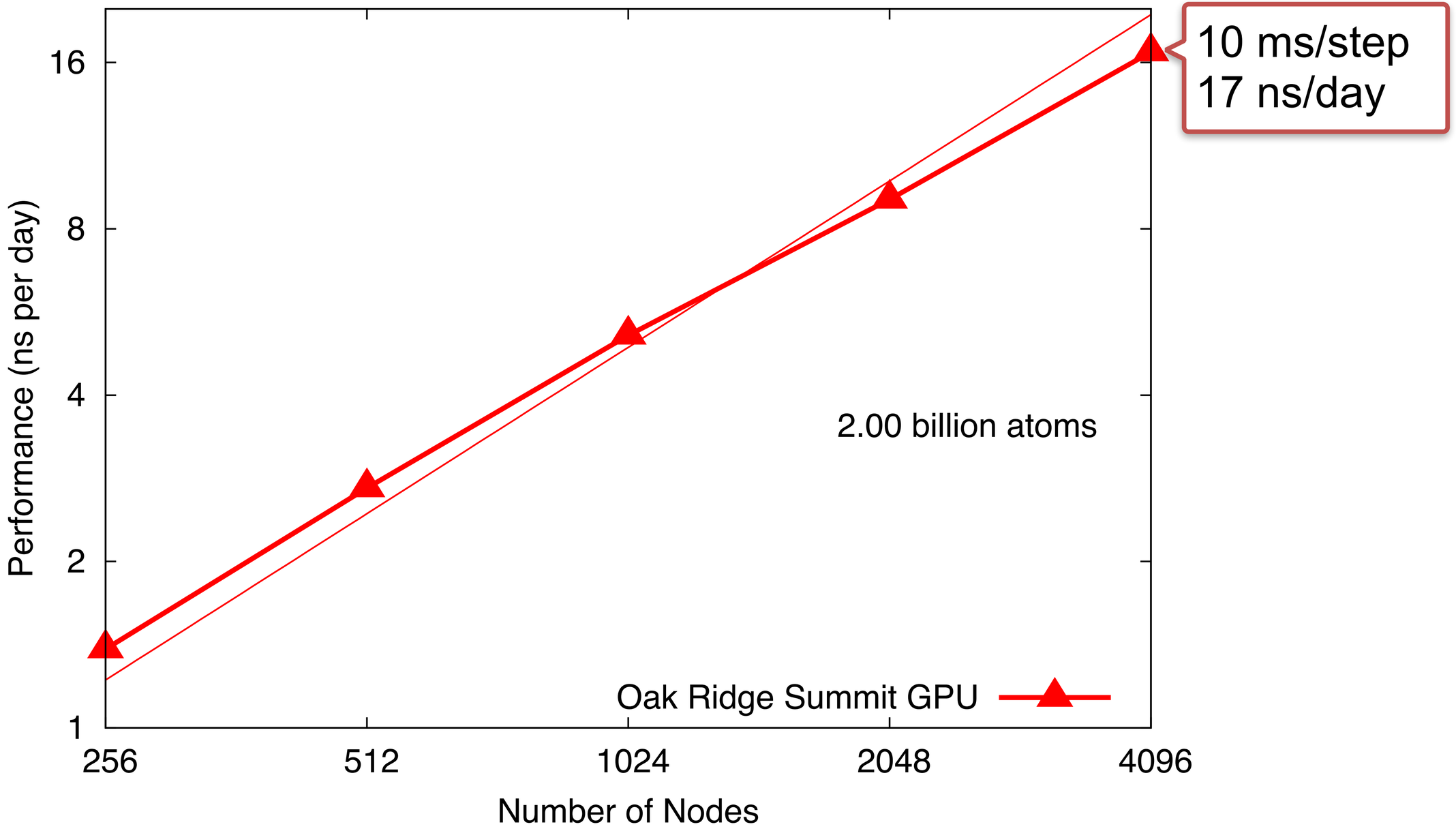
“Fair” comparison for large benchmarks



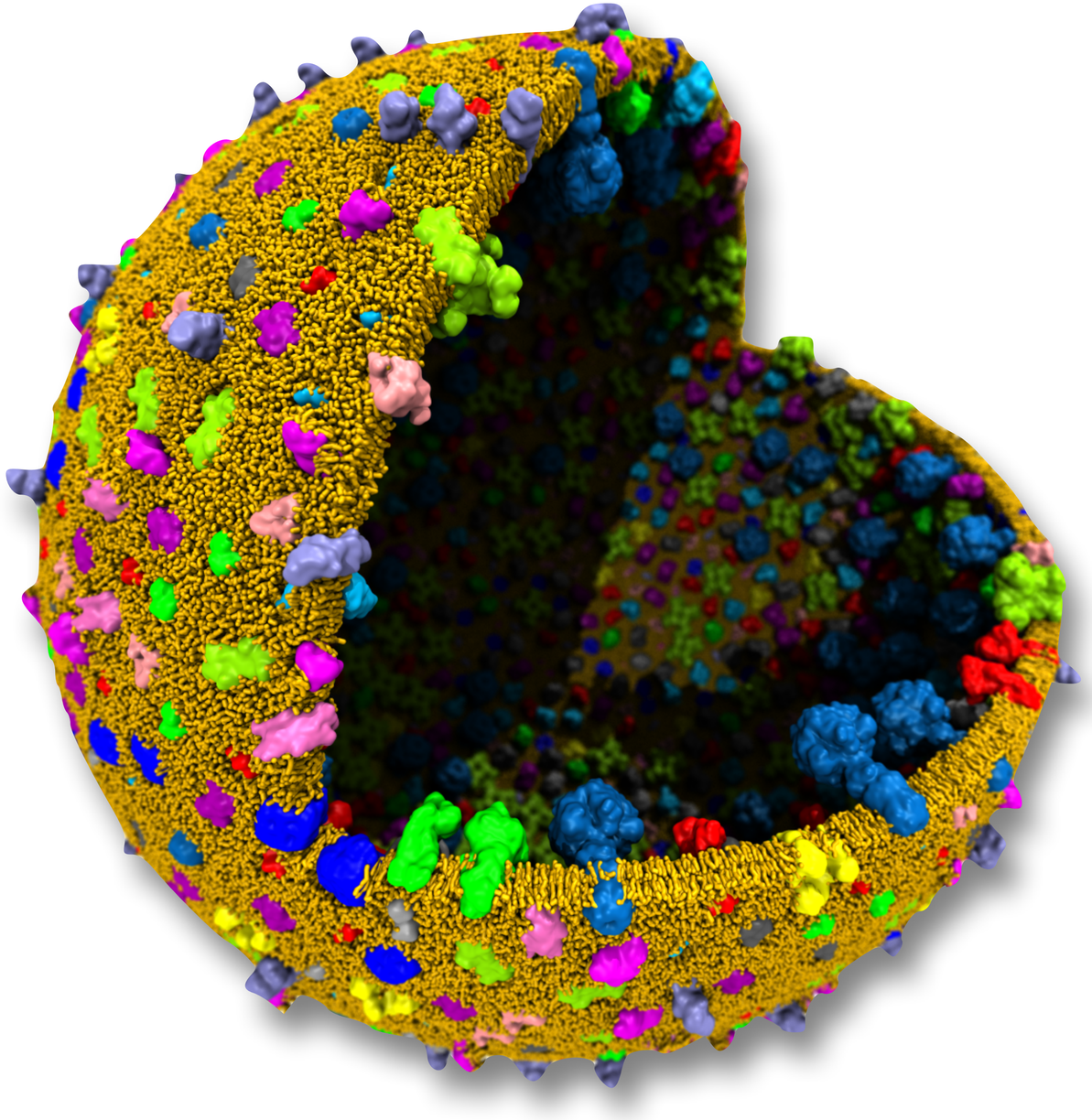
0.8 ms/step
210 ns/day

2.0 ms/step
90 ns/day

Two billion atoms



Summit Early Science: Modeling of a Minimal Cell Envelope



Protein Components	Copy #
Aquaporin Z	97
Copper Transporter (CopA)	166
F1 ATPase	63
Lipid Flipase (MsbA)	29
Molybdenum transporter (ModBC)	130
Translocon (SecY)	103
Methionine transporter (MetNI)	136
Membrane chaperon (YidC)	126
Energy coupling factor (ECF)	117
Potassium transporter (KtrAB)	148
Glutamate transporter (Glt_{Tk})	41
Cytidine-Diphosphate diacylglycerol (Cds)	50
Membrane-bound protease (PCAT)	57
Folate transporter (FolT)	134
	1,397

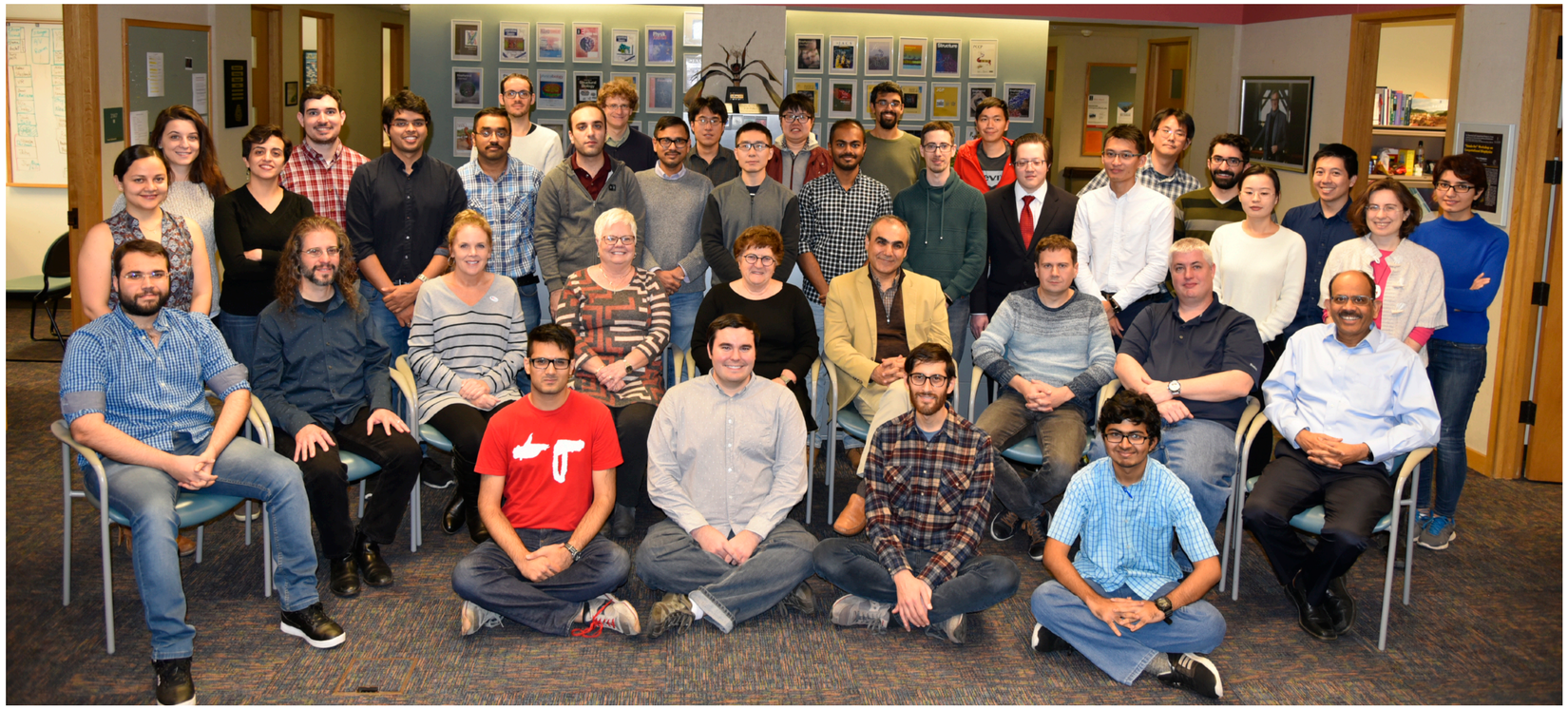
3.7 M lipids, 1,400 proteins,
416 M water molecules, 2.4 M ions

Conclusions and Future Work

- Summit represents a new era in GPU acceleration
  - The CPU will be the bottleneck for many codes
  - Optimizing/vectorizing/parallelizing on the CPU not enough
  - Offload everything practical to the GPUs
- Worry about optimizing the CUDA code last
  - Stage/stream data to reduce CPU/network bottlenecks
- A supercomputer is not just a large cluster
  - IBM knows this (Blue Gene series), Summit now scales well
  - Change is bad, performance regression tests are good
- New Cray machines on the horizon (Perlmutter and Aurora)

Acknowledgments

Antti-Pekka Hynninen,
Ke Li, & Peng Wang, NVIDIA
Sameer Kumar &
Bilge Acun, IBM
Tjerk Straatsma, OLCF
William Kramer, NCSA
Jodi Hadden, Delaware
Rommie Amaro, UCSD
Lorenzo Casalino, UCSD
Abhi Singharoy, ASU



NIH Center for Macromolecular Modeling and Bioinformatics
University of Illinois at Urbana-Champaign

Related Talks to Stream

- Available at on-demand-gtc.gputechconf.com:
  - S9302: Petascale Molecular Dynamics Simulations on the Summit POWER9/Volta Supercomputer
  - S9503 - Using Nsight Tools to Optimize the NAMD Molecular Dynamics Simulation Program
  - S9589 - Interactive High-Fidelity Biomolecular and Cellular Visualization with RTX Ray Tracing APIs
  - S9594 - Bringing State-of-the-Art GPU-Accelerated Molecular Modeling Tools to the Research Community