Quantum Chemistry (QC) on GPUs

May 5, 2016

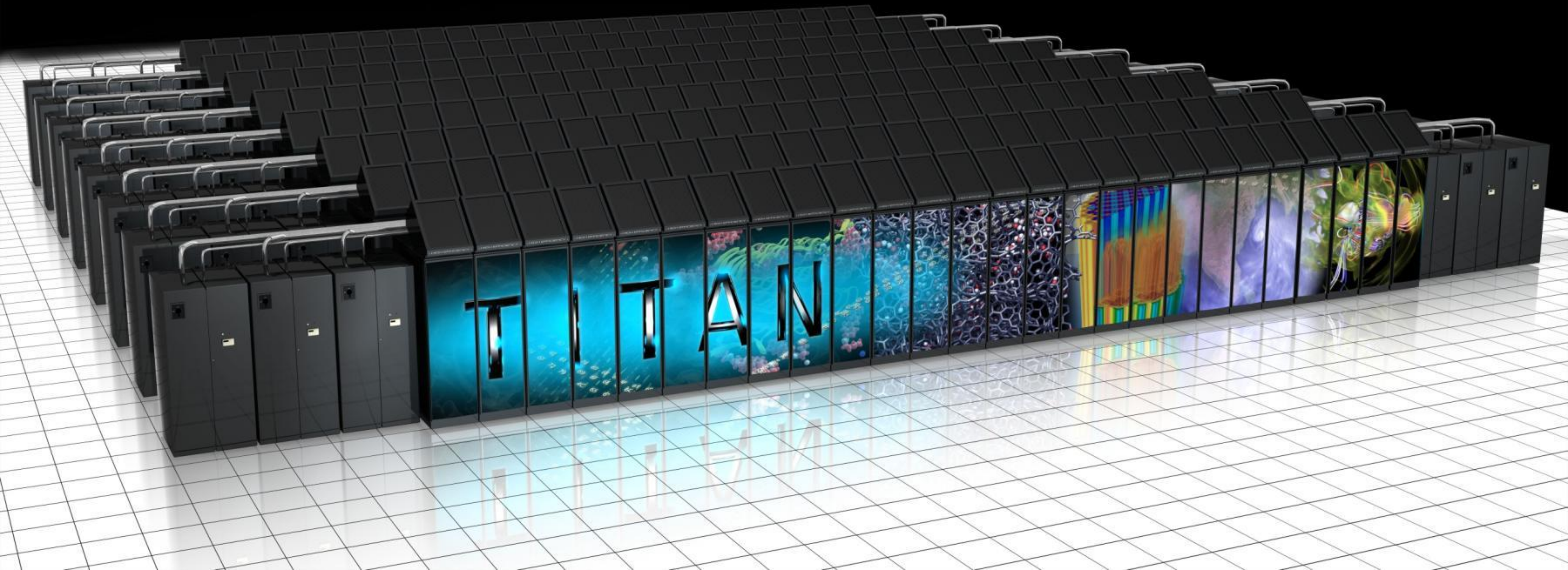
NVIDIA

TITAN: World's #2 Fastest Supercomputer

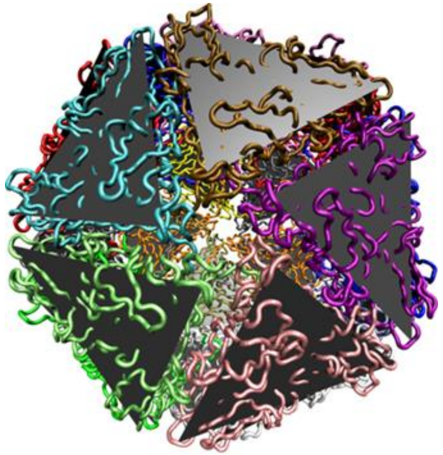
18,688 Tesla K20X GPUs

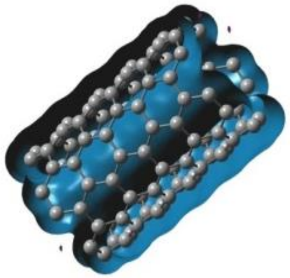
27 Petaflops Peak, 17.59 Petaflops on Linpack

90% of Performance from GPUs



Overview of Life & Material Accelerated Apps





MD: All key codes are GPU-accelerated

- Great multi-GPU performance
- Focus on dense (up to 16) GPU nodes &/or large # of GPU nodes
- ACEMD*, AMBER (PMEMD)*, BAND, CHARMM, DESMOND, ESPResso, Folding@Home, GPUgrid.net, GROMACS, HALMD, HOOMD-Blue*, LAMMPS, Lattice Microbes*, mdcore, MELD, miniMD, NAMD, OpenMM, PolyFTS, SOP-GPU* & more

QC: All key codes are ported or optimizing

- Focus on using GPU-accelerated math libraries, OpenACC directives
- GPU-accelerated and available today:
  - ABINIT, ACES III, ADF, BigDFT, CP2K, GAMESS, GAMESS-UK, GPAW, LATTE, LSDalton, LSMS, MOLCAS, MOPAC2012, NWChem, OCTOPUS*, PEtot, QUICK, Q-Chem, QMCPack, Quantum Espresso/PWscf, QUICK, TeraChem*
- Active GPU acceleration projects:
  - CASTEP, GAMESS, Gaussian, ONETEP, Quantum Supercharger Library*, VASP & more

green* = application where >90% of the workload is on GPU

MD vs. QC on GPUs

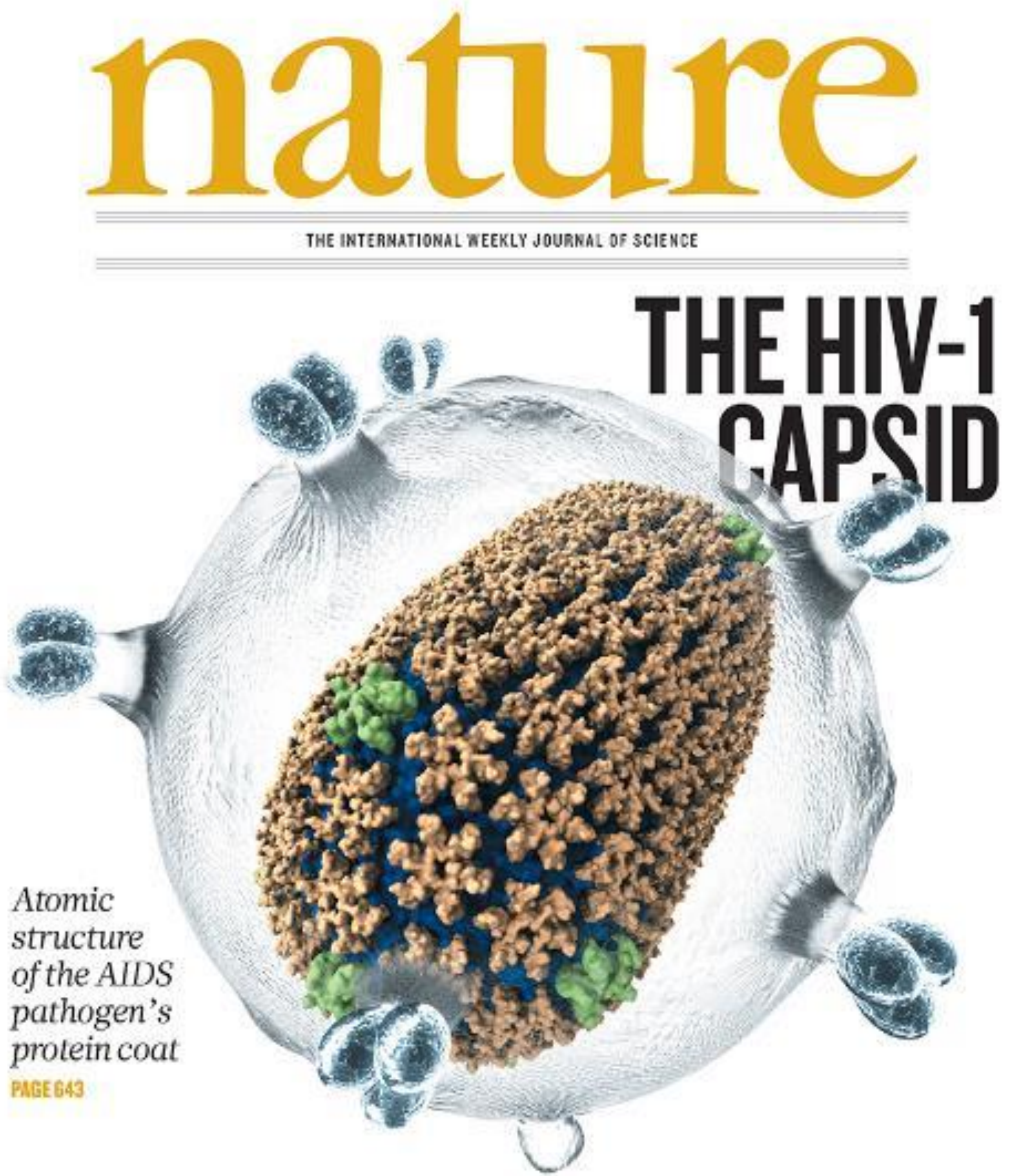
"Classical" Molecular Dynamics	Quantum Chemistry (MO, PW, DFT, Semi-Emp)
Simulates positions of atoms over time; chemical-biological or chemical-material behaviors	Calculates electronic properties; ground state, excited states, spectral properties, making/breaking bonds, physical properties
Forces calculated from simple empirical formulas (bond rearrangement generally forbidden)	Forces derived from electron wave function (bond rearrangement OK, e.g., bond energies)
Up to millions of atoms	Up to a few thousand atoms
Solvent included without difficulty	Generally in a vacuum but if needed, solvent treated classically (QM/MM) or using implicit methods
Single precision dominated	Double precision is important
Uses cuBLAS, cuFFT, CUDA	Uses cuBLAS, cuFFT, OpenACC
Geforce (Accademics), Tesla (Servers)	Tesla recommended
ECC off	ECC on

NVIDIA.

Accelerating Discoveries

Using a supercomputer powered by the Tesla Platform with over 3,000 Tesla accelerators, University of Illinois scientists performed the first all-atom simulation of the HIV virus and discovered the chemical structure of its capsid — "the perfect target for fighting the infection."

Without gpu, the supercomputer would need to be 5x larger for similar performance.



GPU-Accelerated Quantum Chemistry Apps

Green Lettering Indicates Performance Slides Included

- Abinit
- ACES III
- ADF
- BigDFT
- CP2K
- GAMESS-US
- Gaussian
- GPAW
- LATTE
- LSDalton
- MOLCAS
- Mopac2012
- NWChem
- Octopus
- ONETEP
- Petot
- Q-Chem
- QMCPACK
- Quantum Espresso
- Quantum SuperCharger Library
- TeraChem
- VASP
- WL-LSMS

GPU Perf compared against dual multi-core x86 CPU socket.

ABINIT

NVIDIA

ABINIT on GPUS

- Speed in the parallel version:
  - For ground-state calculations, GPUs can be used. This is based on CUDA+MAGMA
  - For ground-state calculations, the wavelet part of ABINIT (which is BigDFT) is also very well parallelized : MPI band parallelism, combined with GPUs

BigDFT

BigDFT
http://bigdft.org

Multiscale Modelling Methods for Applications in Materials Science CECAM JÜLICH, GERMANY

Introduction to Electronic Structure Calculations with BigDFT

Thierry Deutsch, Damien Caliste, Luigi Genovese

L_Sim - CEA Grenoble

17 September 2013

Courtesy of BigDFT team @ CEA

BigDFT version 1.7: capabilities

BigDFT
http://bigdft.org

http://bigdft.org

- Free, surface and periodic boundary conditions
- Geometry optimizations (with constraints)
- Born-Oppenheimer Molecular Dynamics
- Saddle point searches (Nudged-Elastic Band Method)
- Vibrations
- External electric fields
- Unoccupied KS orbitals
- Collinear and Non-collinear magnetism
- All XC functionals of the ABINIT package
- Hybrid functionals
- Empirical van der Waals interactions (many flavors)
- Also available within the ABINIT package

Courtesy of BigDFT team @ CEA

BigDFT version 1.7: capabilities

http://bigdft.org

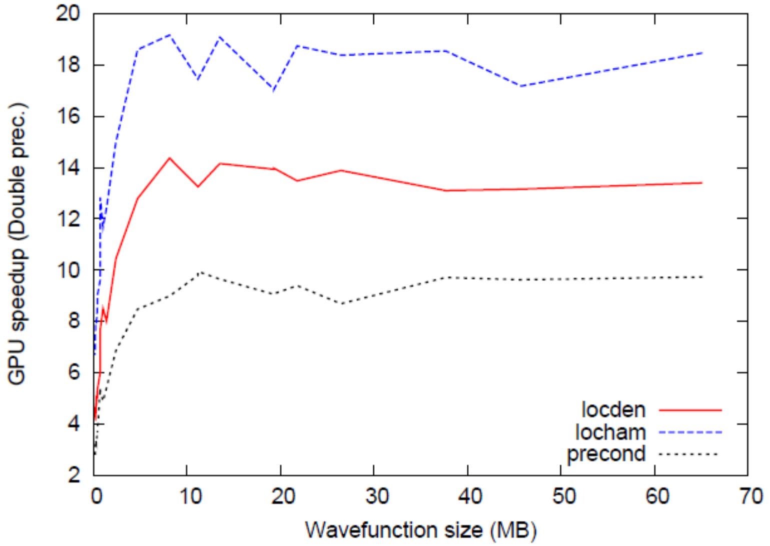
- Free, surface and periodic boundary conditions
- Geometry optimizations (with constraints)
- Born-Oppenheimer Molecular Dynamics
- Saddle point searches (Nudged-Elastic Band Method)
- Vibrations
- External electric fields
- Unoccupied KS orbitals
- Collinear and Non-collinear magnetism
- All XC functionals of the ABINIT package
- Hybrid functionals
- Empirical van der Waals interactions (many flavors)
- Also available within the ABINIT package

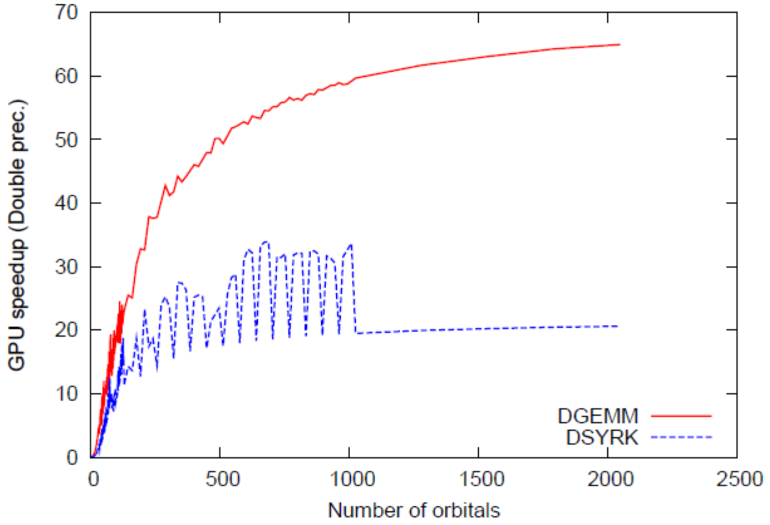
Courtesy of BigDFT team @ CEA

GPU-ported operations in BigDFT (double precision)

Convolutions (OpenCL rewritten)

GPU speedups between 10 and 20 can be obtained for different sections





Linear algebra (CUBLAS library)

The interfacing with CUBLAS is immediate, with considerable speedups

Courtesy of BigDFT team @ CEA

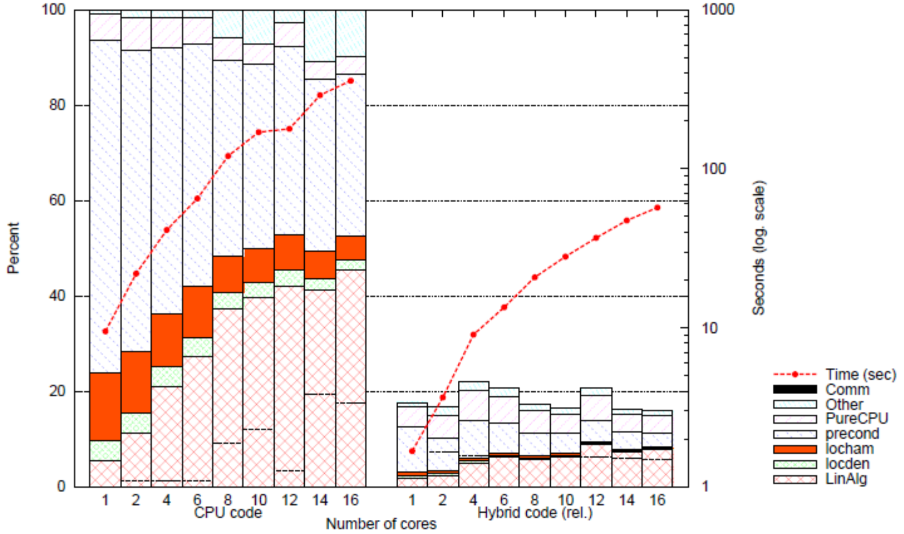
BigDFT code on Hybrid architectures

BigDFT
http://bigdft.org

BigDFT code can run on hybrid CPU/GPU supercomputers
In multi-GPU environments, double precision calculations

No Hot-spot operations

Different code sections can be ported on GPU
up to 20x speedup for some operations,
7x for the full parallel code



Courtesy of BigDFT team @ CEA

Hands on

See

http://bigdft.org/Wiki/index.php?title=Category:Tutorials

- First runs with BigDFT
- Basis-set convergence
- Acceleration example on different platforms:
  Kohn-Sham DFT Operation with GPU acceleration

Courtesy of BigDFT team @ CEA

GAMESS

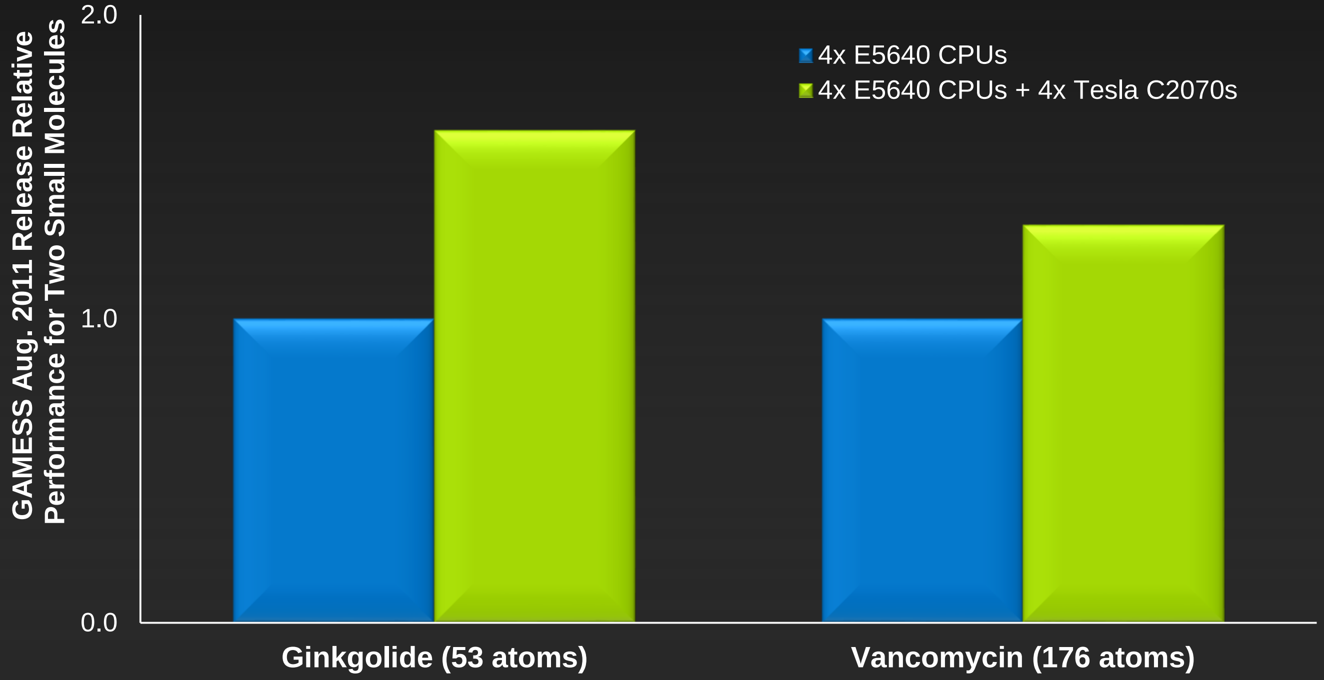
GAMESS Partnership Overview

- Mark Gordon and Andrey Asadchev, key developers of GAMESS, in collaboration with NVIDIA. Mark Gordon is a recipient of a NVIDIA Professor Partnership Award.
- Quantum Chemistry one of major consumers of CPU cycles at national supercomputer centers
- NVIDIA developer resources fully allocated to GAMESS code

> *"We like to push the envelope as much as we can in the direction of highly scalable efficient codes. GPU technology seems like a good way to achieve this goal. Also, since we are associated with a DOE Laboratory, energy efficiency is important, and this is another reason to explore quantum chemistry on GPUs."*
>
> Prof. Mark Gordon
>
> *Distinguished Professor, Department of Chemistry, Iowa State University and Director, Applied Mathematical Sciences Program, AMES Laboratory*

GAMESS August 2011 GPU Performance

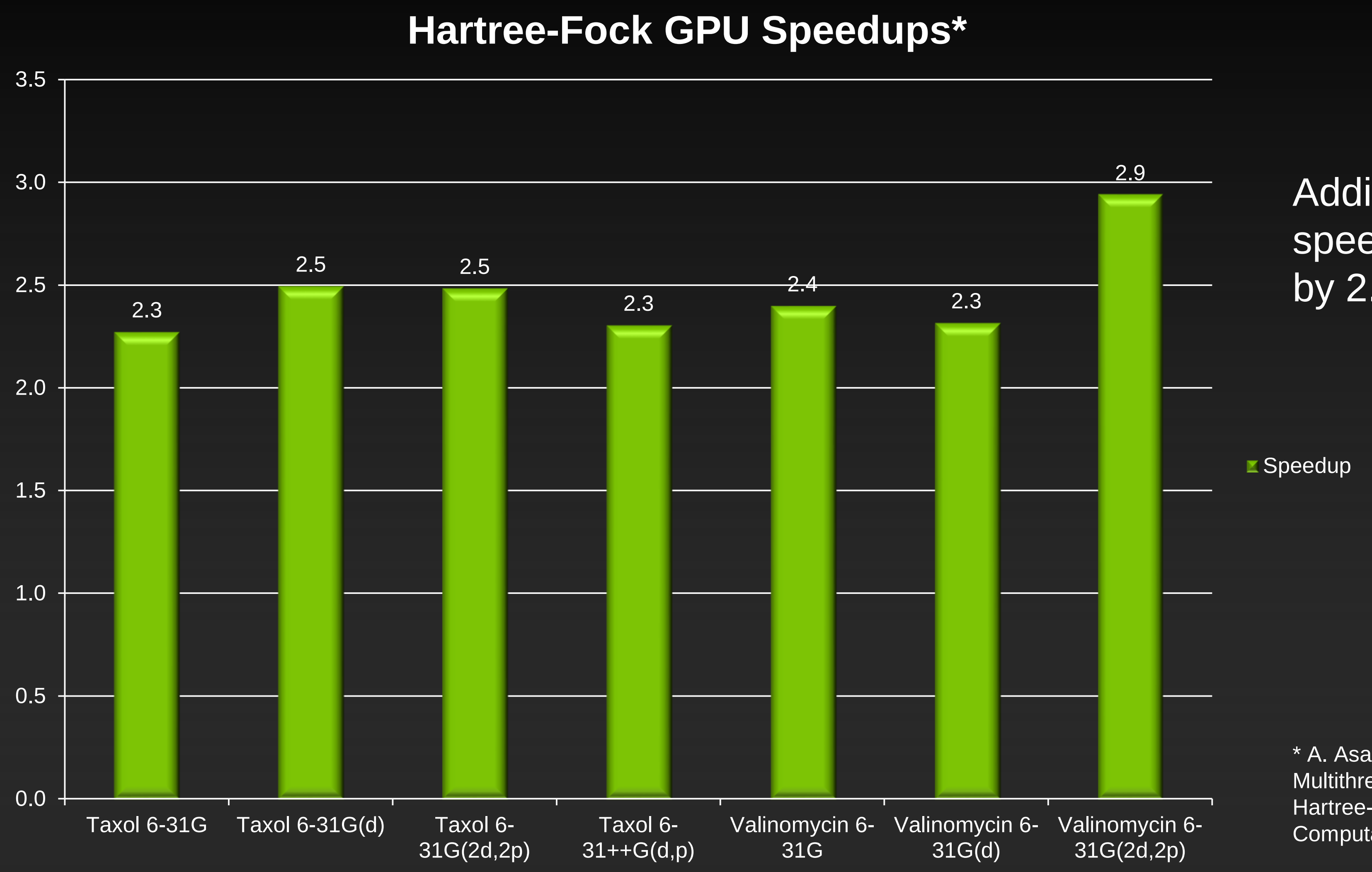
- First GPU supported GAMESS release via "libqc", a library for fast quantum chemistry on multiple NVIDIA GPUs in multiple nodes, with CUDA software
- 2e- AO integrals and their assembly into a closed shell Fock matrix



Upcoming GAMESS Q1 2016 Release

- Multi-nodes with multi-GPUs supported
- Rys Quadrature
- Hartree-Fock
  - 8 CPU cores: 8 CPU cores + M2070 yields 2.3-2.9x speedup. See 2012 publication
- Møller–Plesset perturbation theory (MP2):
  - Paper published
- Coupled Cluster SD(T): CCSD code completed, (T) in progress

GAMESS - New Multithreaded Hybrid CPU/GPU Approach to H-F



Adding 1x 2070 GPU speeds up computations by 2.3x to 2.9x

* A. Asadchev, M.S. Gordon, “New Multithreaded Hybrid CPU/GPU Approach to Hartree-Fock,” Journal of Chemical Theory and Computation (2012)

Gaussian

Gaussian

- ACS Fall 2011 press release
  - Joint collaboration between Gaussian, NVDA and PGI for GPU acceleration: http://www.gaussian.com/g_press/nvidia_press.htm
  - No such press release exists for Intel MIC or AMD GPUs
  - Mike Frisch quote from press release:
    - *"Calculations using Gaussian are limited primarily by the available computing resources," said Dr. Michael Frisch, president of Gaussian, Inc. "By coordinating the development of hardware, compiler technology and application software among the three companies, the new application will bring the speed and cost-effectiveness of GPUs to the challenging problems and applications that Gaussian's customers need to address."*

GPU TECHNOLOGY CONFERENCE

April 4-7, 2016 | Silicon Valley

Excerpts from . . .

ENABLING THE ELECTRONIC STRUCTURE PROGRAM GAUSSIAN ON GPGPUS USING OPENACC

Roberto Gomperts (NVIDIA), Michael Frisch (Gaussian, Inc.), Giovanni Scalmani (Gaussian, Inc.), Brent Leback (NVIDIA/PGI)

PRESENTED BY

NVIDIA

GPU TECHNOLOGY CONFERENCE

PREVIOUSLY

Earlier Presentations

GRC Poster 2012

ACS Spring 2014

GTC Spring 2014 (recording at http://on-demand.gputechconf.com/gtc/2014/video/S4613-enabling-gaussian-09-gpgpus.mp4)

WATOC Fall 2014

Full presentation available

GTC Spring 2016 (this full recording at http://mygtc.gputechconf.com/quicklink/4r13O5r; requires registration)

TOPICS

Gaussian: Design Guidelines, Parallelism and Memory Model

Implementation: Top-Down/Bottom-Up

OpenACC: Extensions, Hints & Tricks

Early Performance

Closing Remarks

GPU TECHNOLOGY CONFERENCE

GAUSSIAN

A Computational Chemistry Package that provides state-of-the-art capabilities for electronic structure modeling

Gaussian 09 is licensed for a wide variety of computer systems

All versions of Gaussian 09 contain virtually every scientific/modeling feature, and none imposes any artificial limitations on calculations other than computational resources and time constraints

Researchers use Gaussian to, among others, study molecules and reactions; predict and interpret spectra; explore thermochemistry, photochemistry and other excited states; include solvent effects, and many more

DESIGN GUIDELINES

General

- Establish a Framework for the GPU-enabling of Gaussian

Code Maintainability (Code Unification)

- Leverage Existing code/algorithms, including Parallelism and Memory Model
    - Simplifies Resolving Problems
    - Simplifies Improvement on existing code
    - Simplifies Adding New Code

DESIGN GUIDELINES

- Accelerate Gaussian for Relevant and Appropriate Theories and Methods
  - Relevant: many users of Gaussian
  - Appropriate: time consuming and good mapping to GPUs
- Resource Utilization
  - Ensure efficient use of all available Computational Resources
    - CPU cores and memory
    - Available GPUs and memory

CURRENT STATUS

Single Node

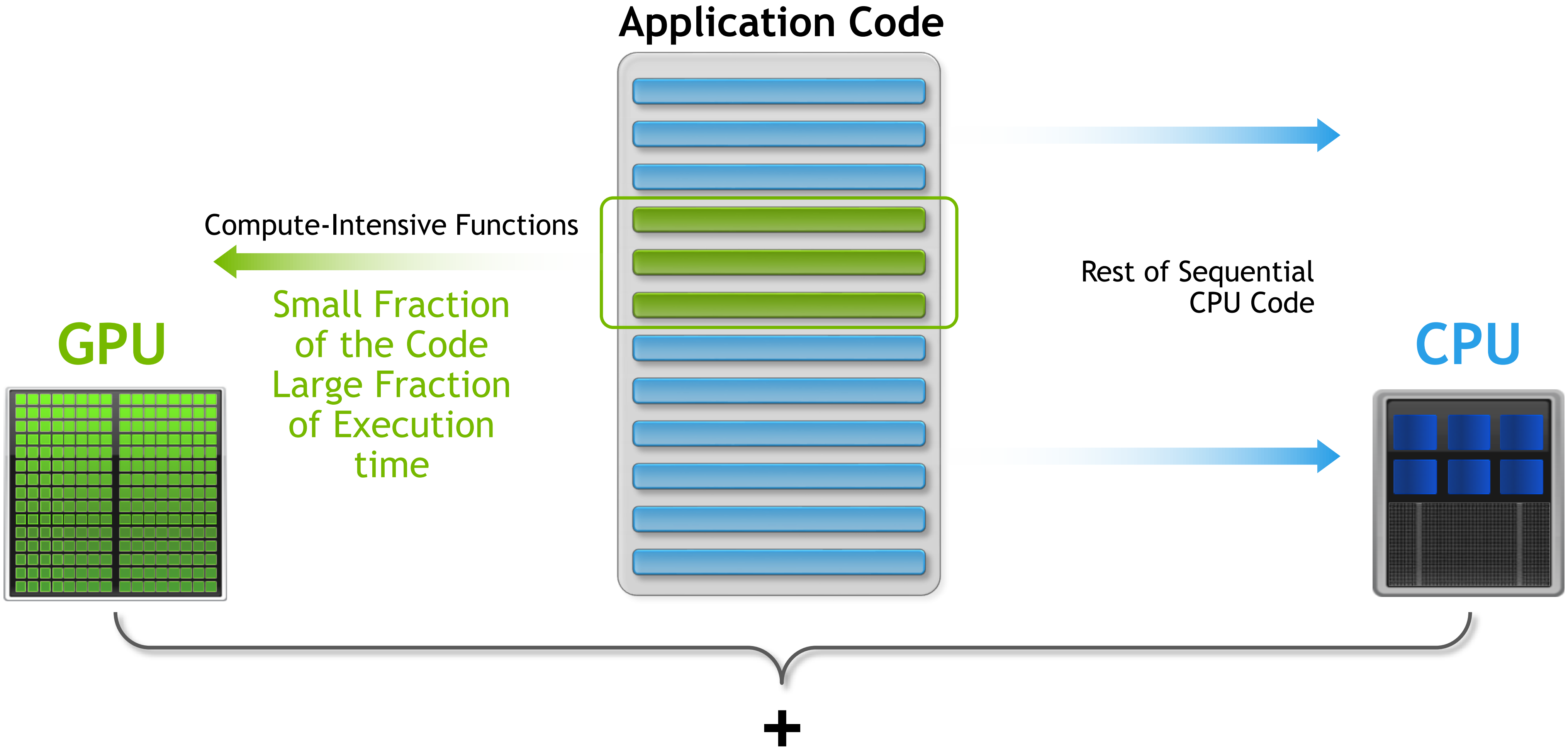
Implemented

- Energies for Closed and Open Shell HF and DFT (less than a handful of XC-functionals missing)
- First derivatives for the same as above
- Second derivatives for the same as above

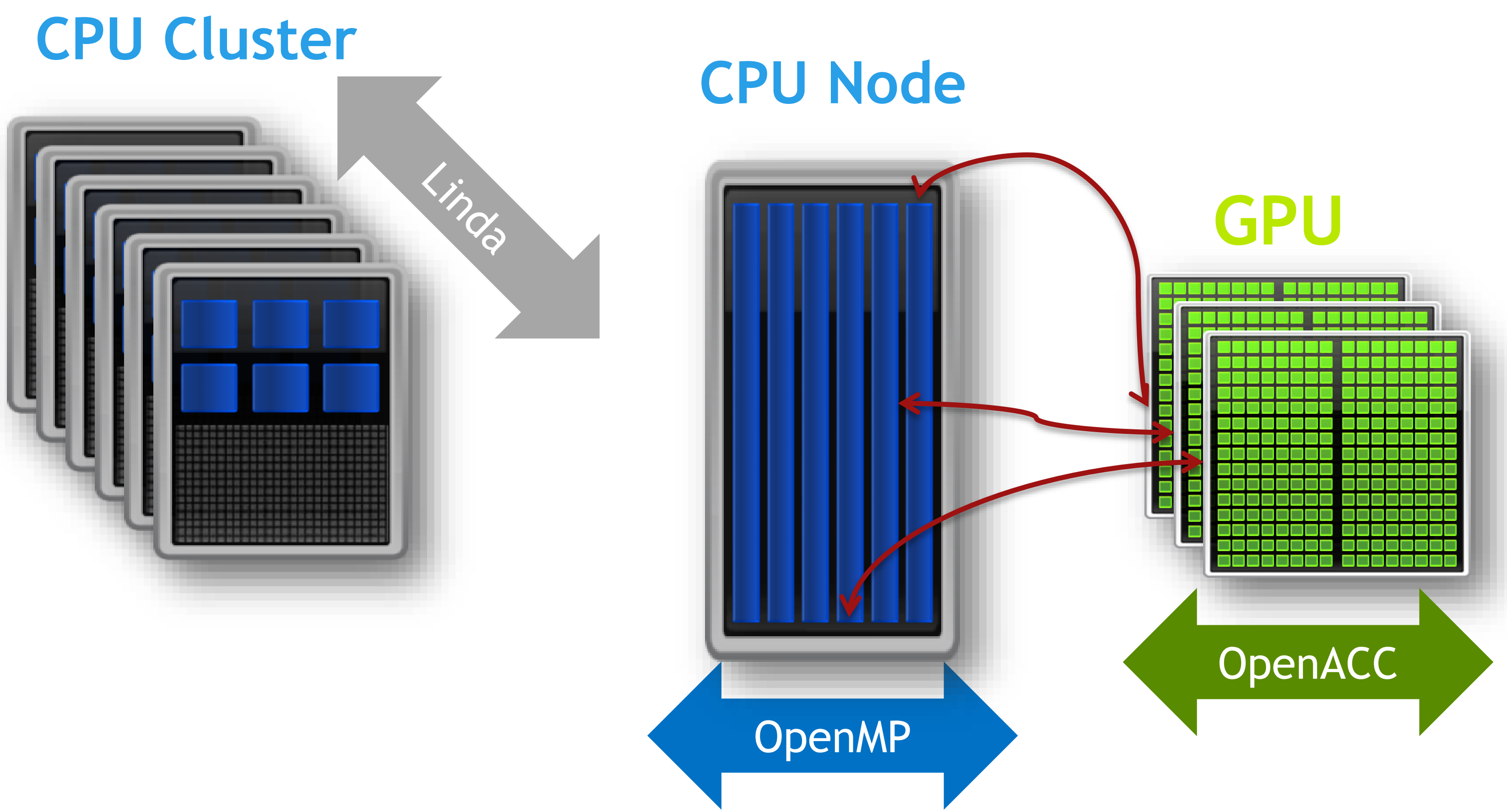
Using only

- OpenACC
- CUDA library calls (BLAS)

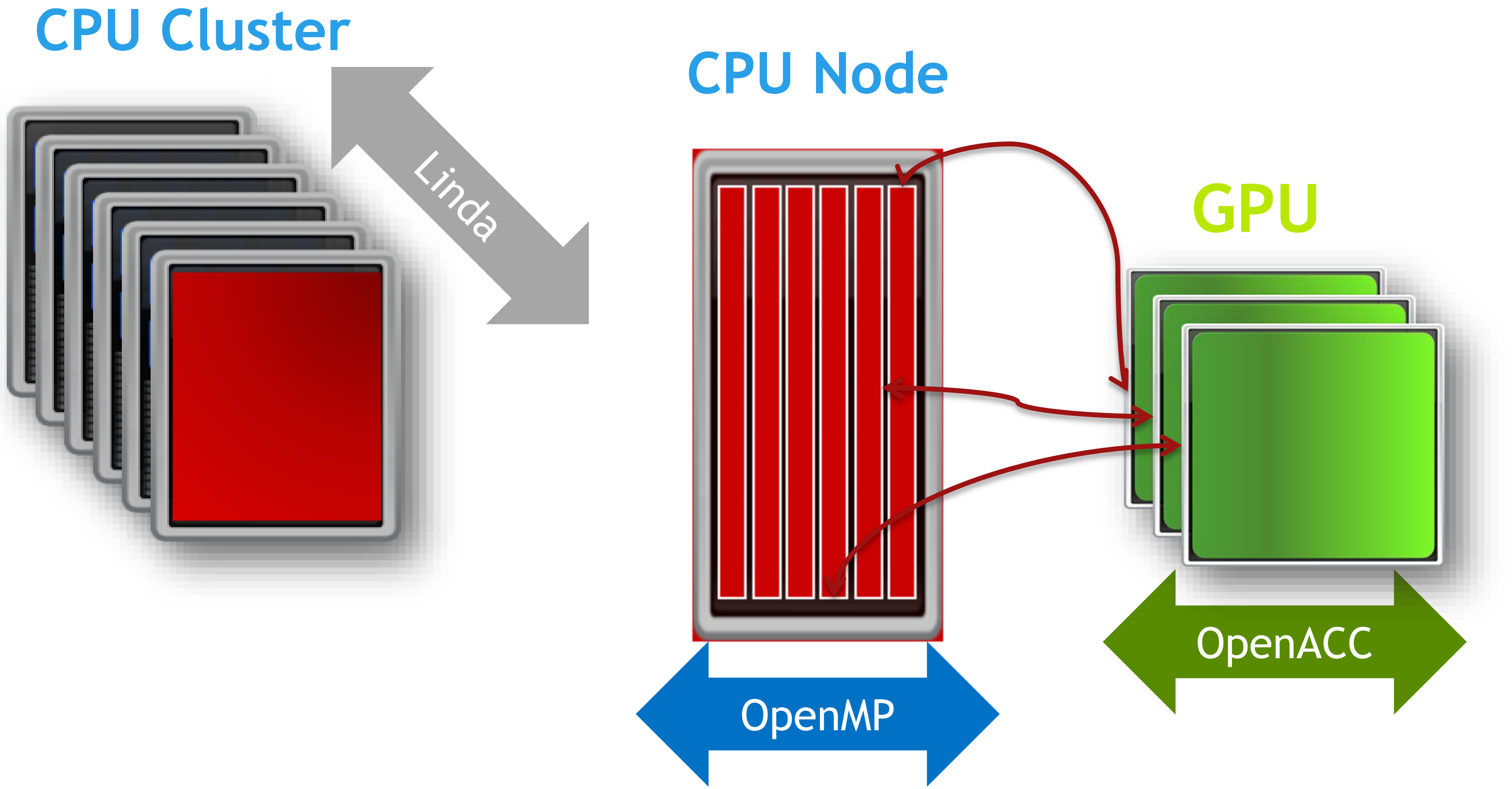
IMPLEMENTATION MODEL



GAUSSIAN PARALLELISM MODEL



GAUSSIAN: MEMORY MODEL



CLOSING REMARKS

Significant Progress has been made in enabling Gaussian on GPUs with OpenACC

OpenACC is increasingly becoming more versatile

Significant work lies ahead to improve performance

Expand feature set:

- PBC, Solvation, MP2, ONIOM, triples-Corrections

ACKNOWLEDGEMENTS

Development is taking place with:

Hewlett-Packard (HP) Series SL2500 Servers (Intel® Xeon® E5-2680 v2 (2.8GHz/10-core/25MB/8.0GT-s QPI/115W, DDR3-1866)

NVIDIA® Tesla® GPUs (K40 and later)

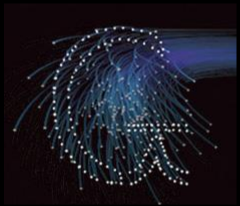
PGI Accelerator Compilers (16.x) with OpenACC (2.5 standard)

Select Slides from "Enabling Gaussian 09 on GPGPUs" at GTC March 2014

- In 2011 Gaussian, Inc., NVIDIA Corp. and PGI started a long-term project to enable all the performance critical paths of Gaussian on GPGPUs.
  - Ultimate goal is to show significant performance improvement by using accelerators in conjunction with CPUs
  - Initial efforts are directed towards creating an infrastructure that will leverage the current CPU code base and at the same time minimize the additional maintenance effort associated with running on GPUs.
- Current status of this work for Direct Hartree-Fock and triples-correction calculations as applied in for example Coupled Cluster calculations that uses mostly the directives based OpenACC framework.

- Slides & Audio: http://on-demand.gputechconf.com/gtc/2014/video/S4613-enabling-gaussian-09-gpgpus.mp4

GPU TECHNOLOGY CONFERENCE



PGI

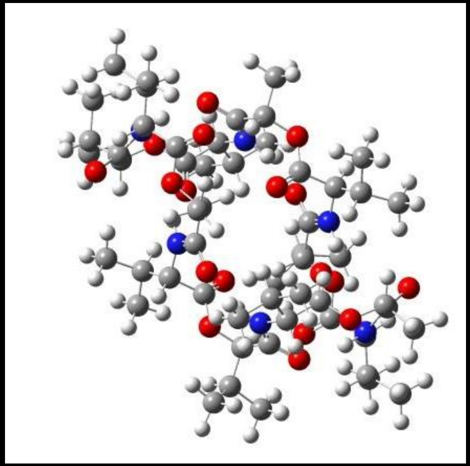
NVIDIA

CURRENT STATUS OF THE PROJECT TO ENABLE GAUSSIAN 09 ON GPGPUS

Roberto Gomperts (NVIDIA, Corp.)
Michael Frisch (Gaussian, Inc.)
Giovanni Scalmani (Gaussian, Inc.)
Brent Leback (PGI)

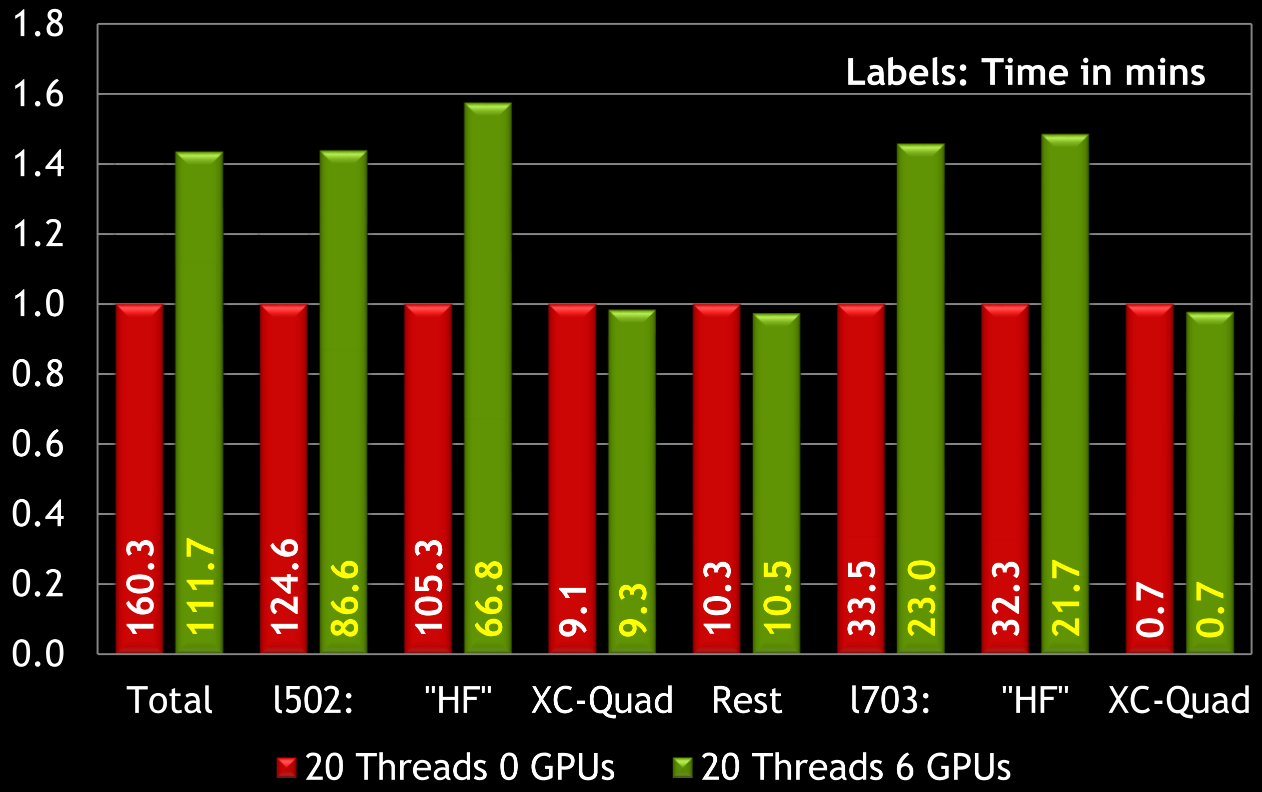
GPU TECHNOLOGY CONFERENCE

EARLY PERFORMANCE RESULTS (DIRECT SCF)



Method	rB3LYP
No. of Atoms	168
Basis Set	6-31G(3df,3p)
No. of Basis Funcs	3 642
No. of Cycles	17

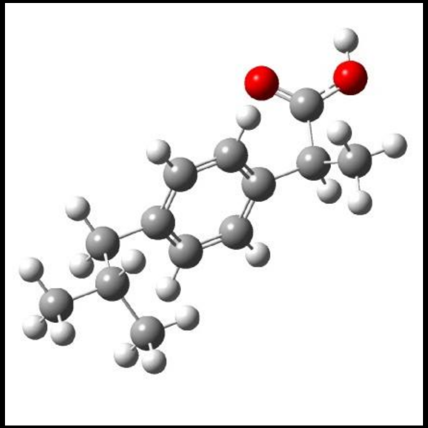
Valinomycin Force Calculation
Speed Ups Relative to CPU-Only Full Node



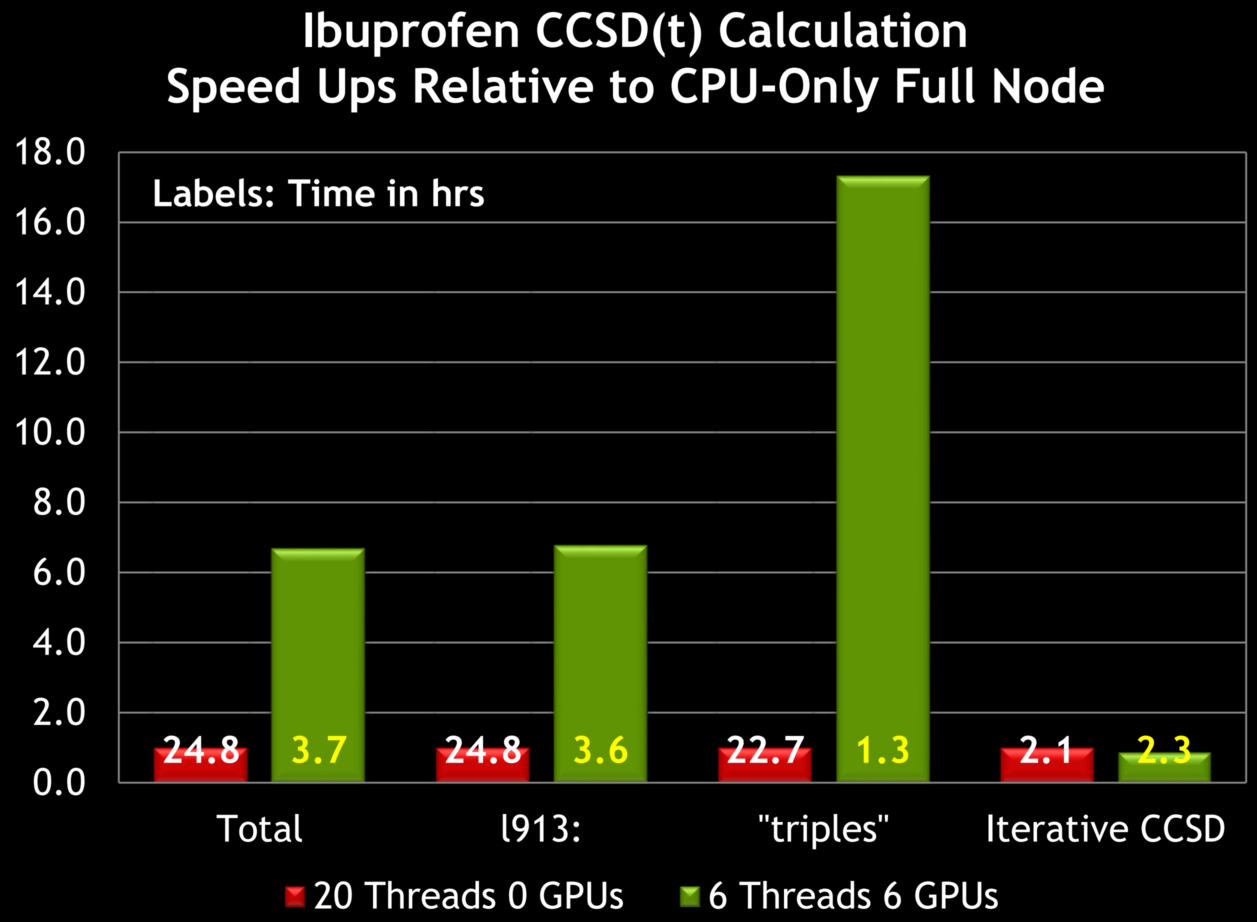
System: 2 Sockets E5-2690 V2 (2x 10 Cores @ 3.0 GHz); 128 GB RAM (DD3-1600); Used 108 GB
GPUs: 6 Tesla K40m (15 SMPs @ 875 MHz); 12 GB Global Memory

GPU TECHNOLOGY CONFERENCE

EARLY PERFORMANCE RESULTS (CCSD(T))



Method	CCSD(t)
No. of Atoms	33
Basis Set	6-31G(d,p)
No. of Basis Funcs	315
No. Occ Orbitals	41
No. Virt Orbitals	259
No. of Cycles	15
No. CCSD iters	16



System: 2 Sockets E5-2690 V2 (2x 10 Cores @ 3.0 GHz); 128 GB RAM (DD3-1600); Used 108 GB
GPUs: 6 Tesla K40m (15 SMPs @ 875 MHz); 12 GB Global Memory

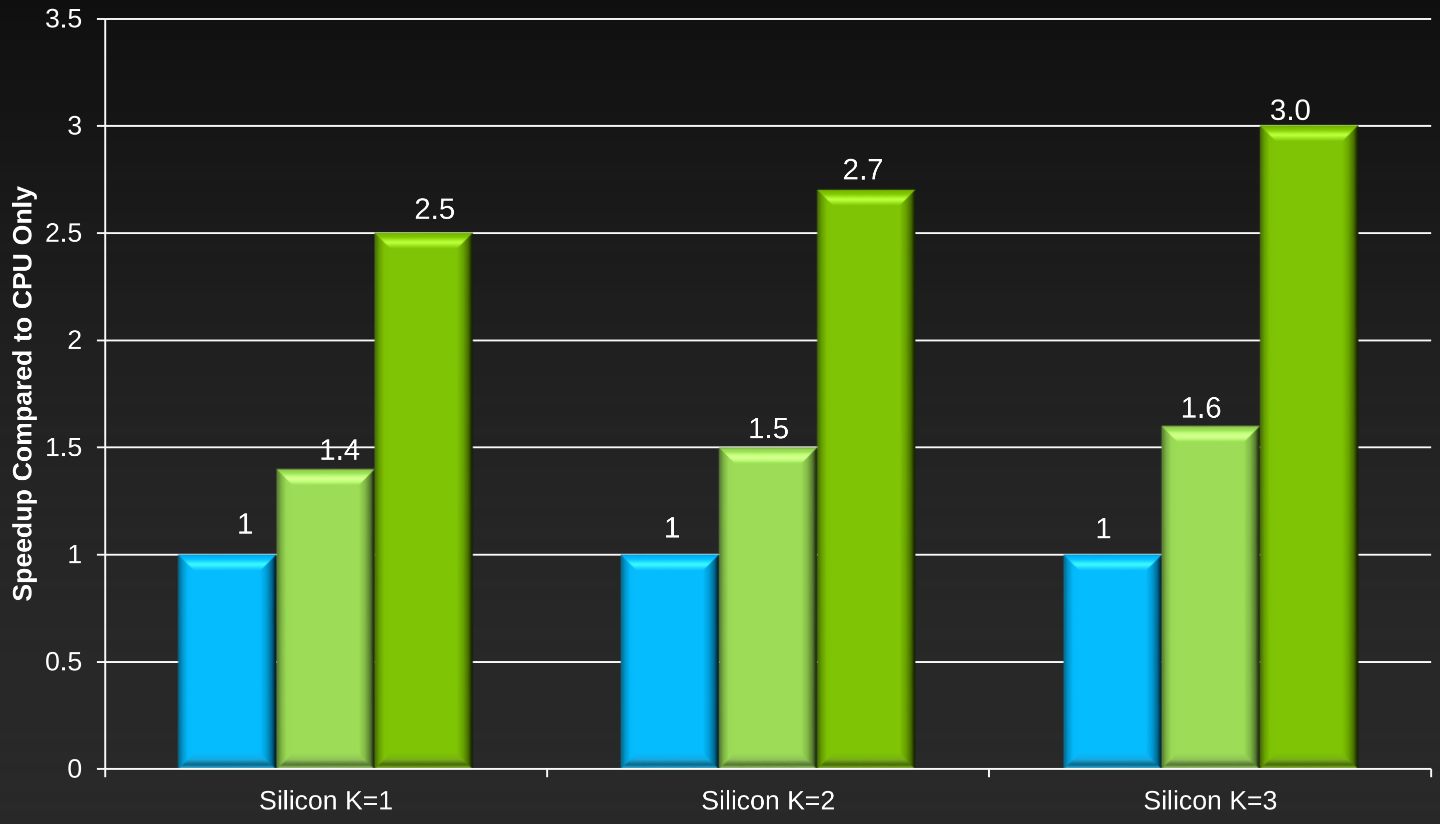
GPU TECHNOLOGY CONFERENCE

CLOSING REMARKS

- Significant progress has been made in creating a framework that keeps an unified code structure for GPU enabled Gaussian
- There is room for performance improvement in the Direct SCF work
- The (t) correction performance looks promising
- Further work: Continue working towards a “product” quality version of Gaussian to be released to customers
  - Continue unification of the code base
  - Tackle non-default paths of the currently enabled code
  - Expand enabling of other Gaussian functionality (2^{nd} Derivatives, XC-quadrature, TDDFT, MP2, etc.)
  - Performance tuning

GPAW

Increase Performance with Kepler

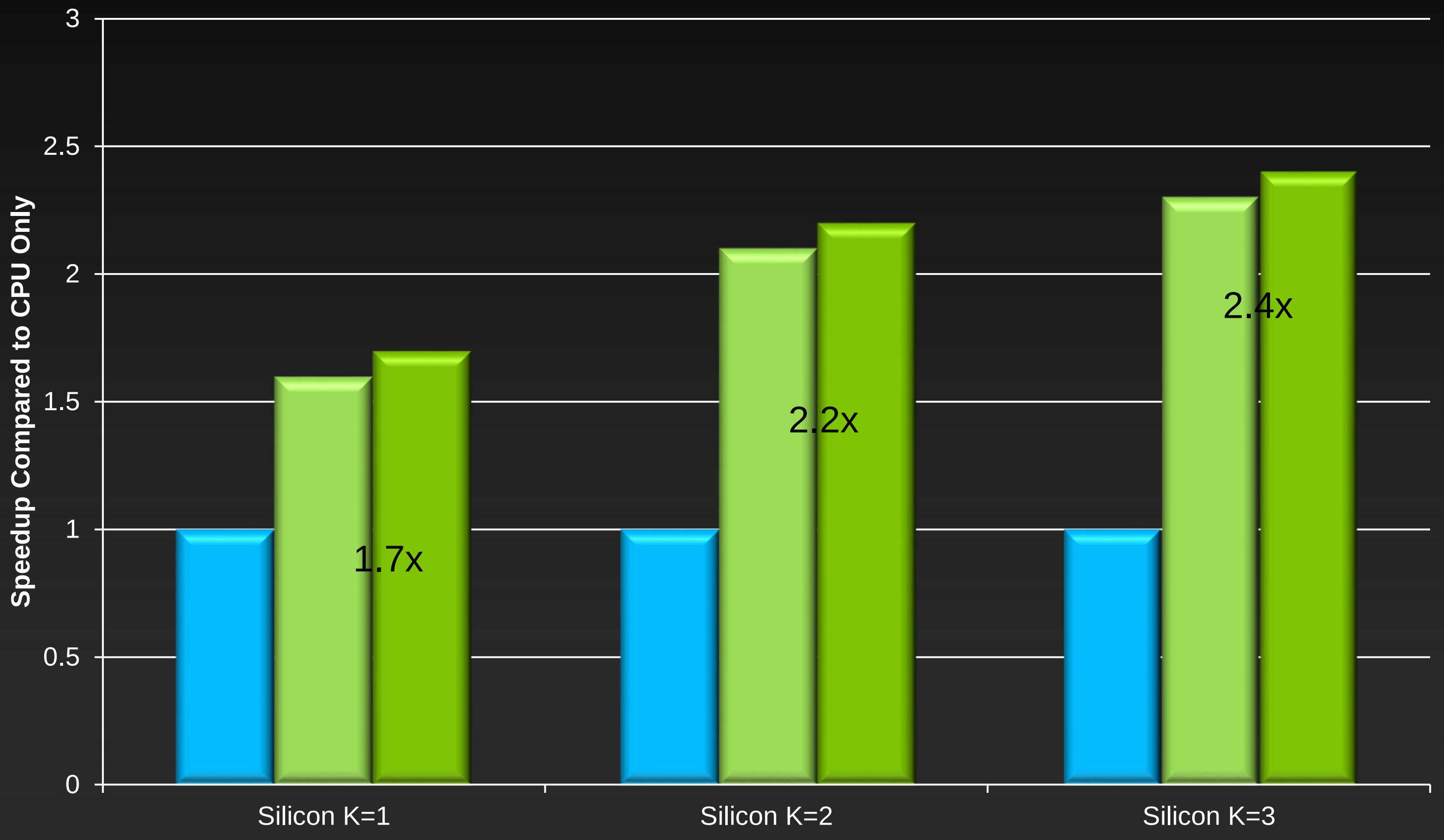


Running GPAW 10258

The blue nodes contain 1x E5-2687W CPU (8 Cores per CPU).

The green nodes contain 1x E5-2687W CPU (8 Cores per CPU) and 1x or 2x NVIDIA K20X for the GPU.

Increase Performance with Kepler

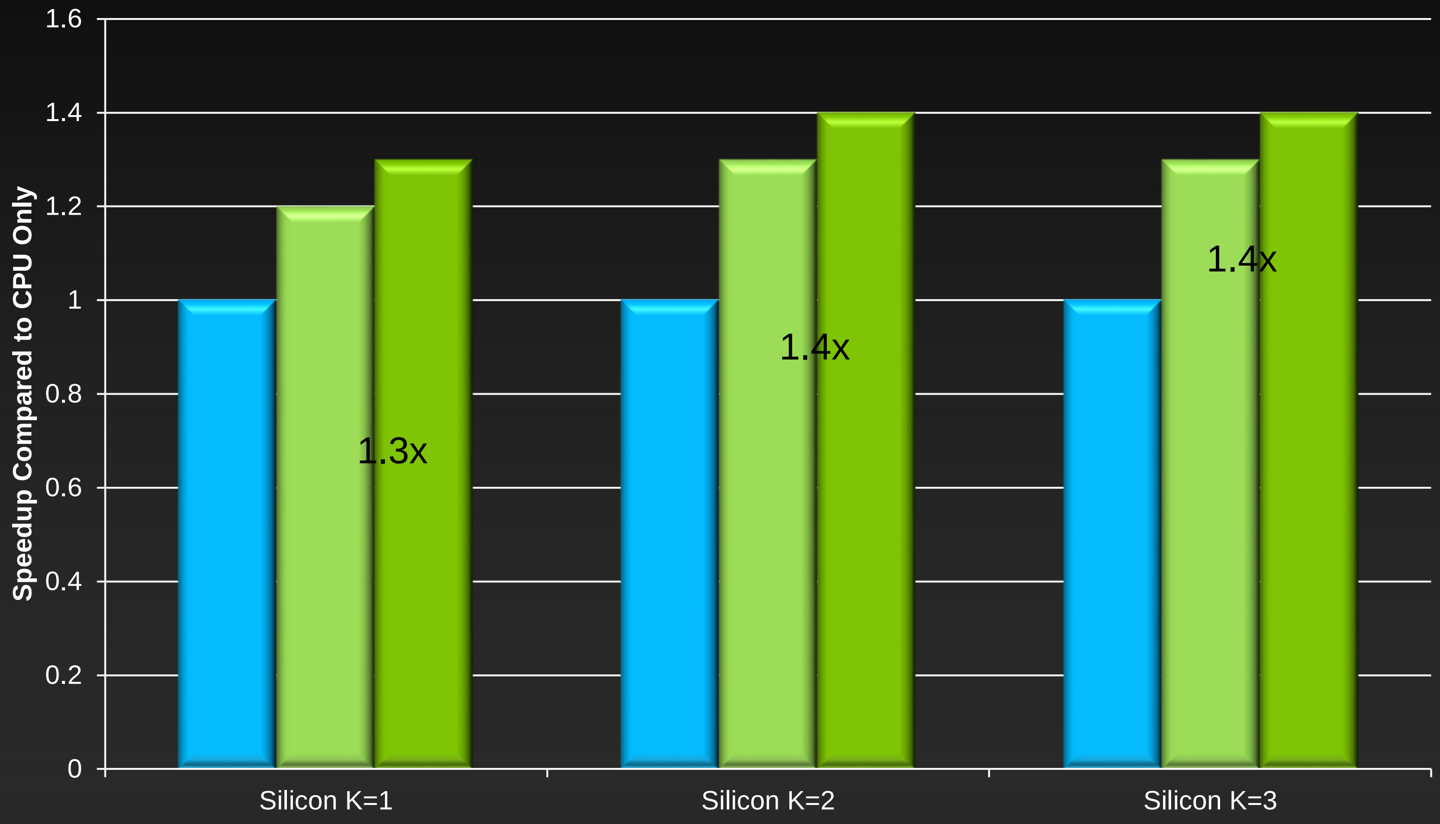


Running GPAW 10258

The blue nodes contain 1x E5-2687W CPU (8 Cores per CPU).

The green nodes contain 1x E5-2687W CPUs (8 Cores per CPU) and 2x NVIDIA K20 or K20X for the GPU.

Increase Performance with Kepler



Running GPAW 10258

The blue nodes contain 2x E5-2687W CPUs (8 Cores per CPU).

The green nodes contain 2x E5-2687W CPUs (8 Cores per CPU) and 2x NVIDIA K20 or K20X for the GPU.

Multi-GPU Accelerated Large Scale Electronic Structure Calculations

Used with permission from Samuli Hakala

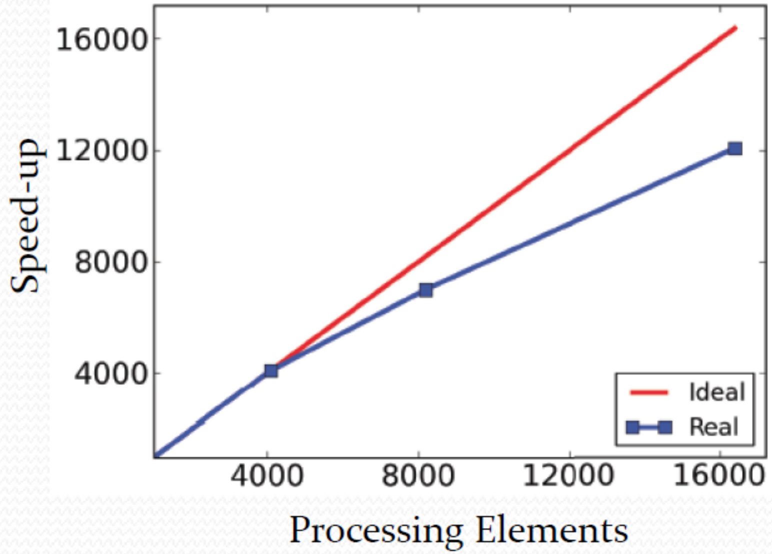
Samuli Hakala
COMP Centre of Excellence
Department of Applied Physics
Aalto University School of Science
Email: samuli.hakala@aalto.fi

GPU Technology Conference, March 2013

Aalto University
School of Science

GPAW

- Density Functional Theory (DFT) program package for electronic structure calculations
- Time-Dependent Density Functional Theory (TDDFT) is implemented in the linear response and time propagation schemes
- Can use real-space grids, atom centered basis functions or plane waves
- Random Phase Approximation (RPA) also available
- Scales to thousands of cores and suitable for large scale calculations
- Open Source software licensed under GPL



Ground state DFT calculation of 561 Au atom cluster on Blue Gene/P.

LibXC on GPUs

- A reusable library of >250 exchange-correlation functionals
- Used by 15 different codes (Abinit, GPAW, BigDFT, etc.)
- Can be a performance bottleneck for small systems
- Can "clone" existing functionals for GPU use with fairly minimal changes to existing LibXC code and parallelizes well over grid points
- More information:
  - https://confluence.slac.stanford.edu/display/SUNCAT/libxc+on+GPUs
- Work by Lin Li, Jun Yan, Christopher O'Grady (Stanford/SLAC)

Functional	Type	Speedup ((GPU+CPU)/CPU)
PW, PW Mode, OB PW, PW RPA	LDA Correlation	23,23,23,37
PBE, PBE sol, xPBE, PBE JRGX, RGE2, APBE	GGA Correlation	56, 58, 58, 58, 58, 58
RPBE	GGA Exchange	95
TPSS	MGGA Exchange	51

Ground State Performance

Bulk Silicon

- 95 atoms with periodic boundary conditions, 380 bands and 1 k-point. Grid size: 56x56x80.
- Time is in seconds per one SCF iteration.
- Intel Xeon X5650, NVIDIA Tesla M2070

Fullerene

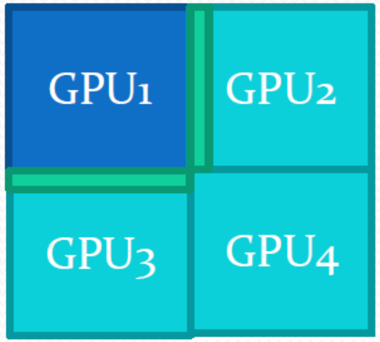
- C60 molecule with 240 valence electrons. Grid size: 84x84x84
- Intel Xeon X5650, NVIDIA Tesla M2070

Si95	CPU	GPU	%	S-Up
Poisson Solver	1.8	0.13	1%	**14**
Orthonormalization	23	3.0	23%	**7.7**
Precondition	9.4	0.77	6%	**12**
RMM-DIIS other	32	3.2	25%	**10**
Subspace Diag	23	2.1	16%	**11**
Other	2.7	2.7	21%	**1.0**
Total (SCF-Iter)	93	13		9.7/7.7

C60	CPU	GPU	%	S-Up
	13	0.64	7%	**20**
	11	1.2	13%	**9.2**
	16	0.99	11%	**16**
	8.1	0.6	7%	**13**
	22	2.1	23%	**10**
	3.5	3.2	35%	**1.1**
	76	9.1		13/8.3

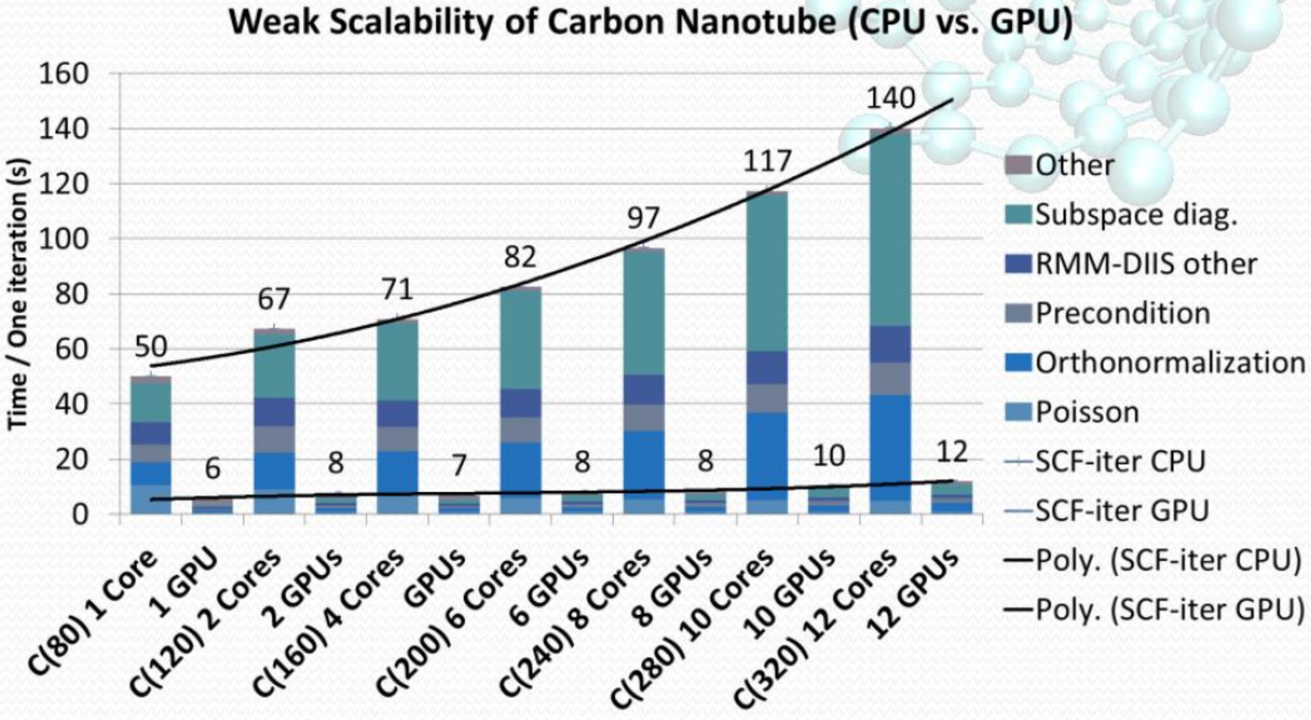
Multi-GPU Parallelization

- Parallelization is done with MPI
- Multiple GPUs can be used by domain decomposition or parallelization over k-points or spins
- Domain decomposition for the stencil operations involves exchanging boundary regions between neighboring nodes
- Communications between nodes require data movement: device memory → host memory → destinations node host memory → destinations node device memory.
- Overlaps receives, sends and computations in the middle part of the grid, BUT this causes issues with small grids
  - Small grids: Synchronous transfers
  - Medium grids: Asynchronous transfers
  - Large grids: Overlap calculations and asynchronous transfers
  - Combine of several wave functions and boundary regions into few large transfers



Weak Scalability (Carbon)

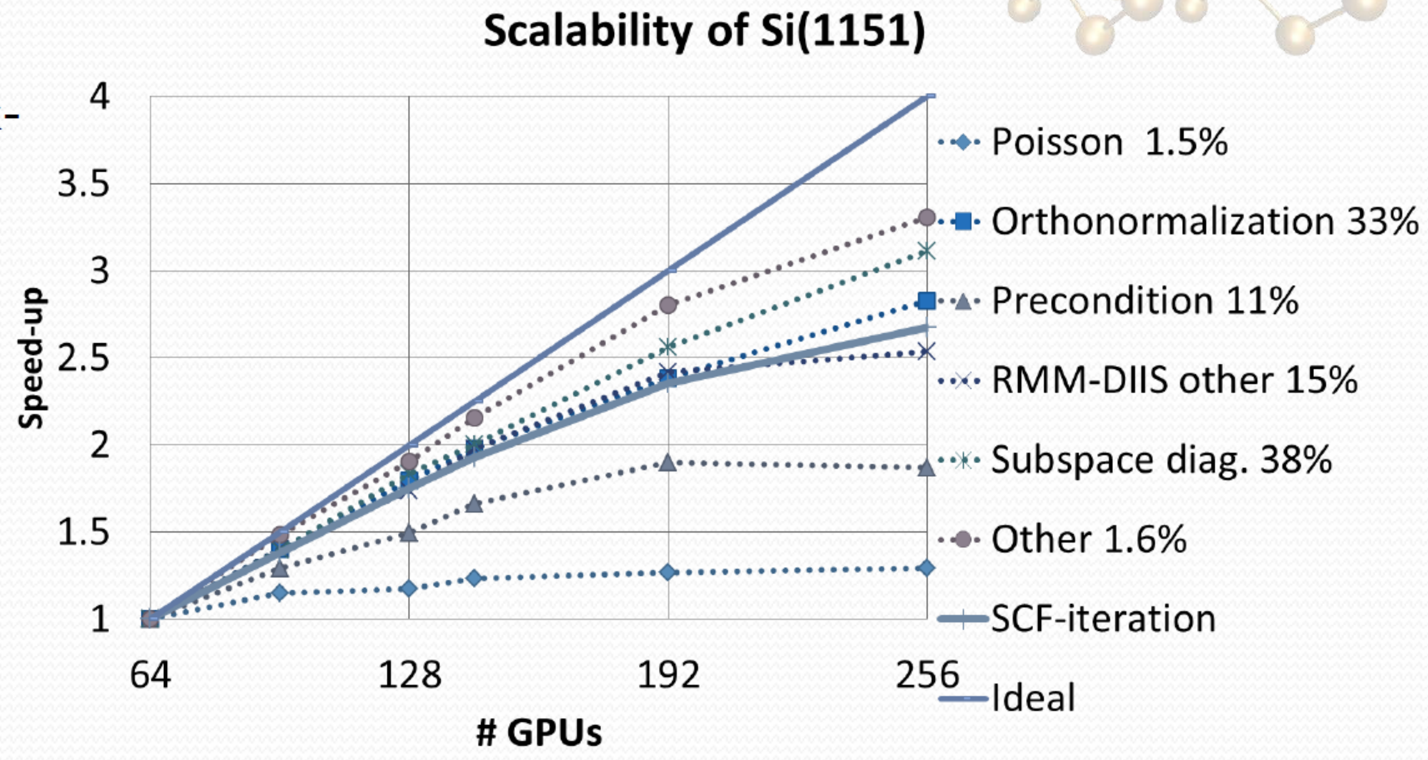
- The size of a carbon nanotube and the number of MPI tasks are varied from 80 atoms (240 states) to 320 atoms (1280 states) and 1 task to 12 tasks.
- Comparison between equal number of GPUs and CPU cores.
- CPU: Intel Xeon X5650 GPU: NVIDIA Tesla M2070
- Calculations performed on Vuori cluster at CSC



# MPI tasks	1	2	4	6	8	10	12
Speed-Up	8.8	8.7	10.5	10.2	11.5	11.3	11.9

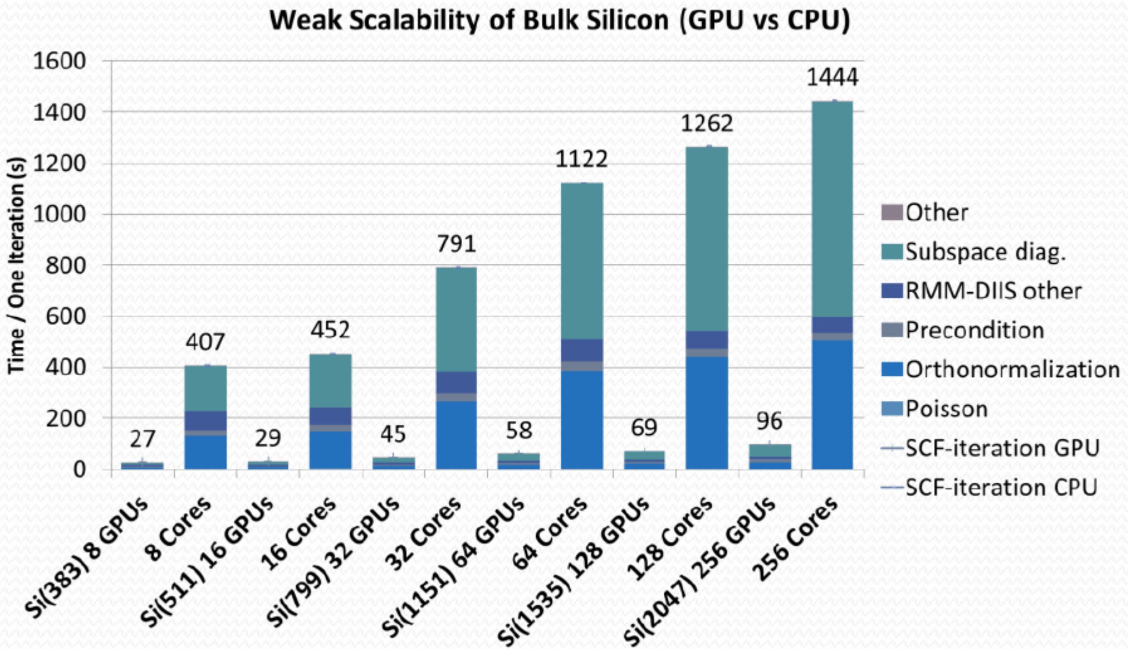
Strong Scalability

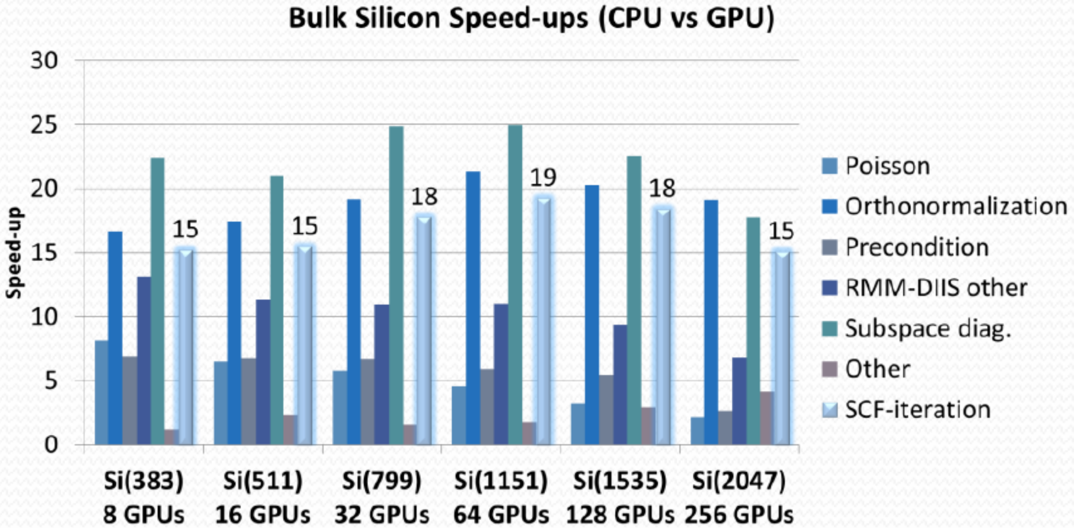
- Bulk silicon with 1151 atoms with periodic boundary conditions, 4604 bands and 1 k-point in the Brillouin zone.
- The number of GPUs is increased from 64 to 256.
- Grid size: 164x164x108
- Speed-up comparison to 64 GPUs.
- NVIDIA Tesla M2090
- Calculations performed on CURIE cluster in France at GENCI/CEA



Weak Scalability (Silicon)

- The size of bulk silicon system and the number of MPI tasks are varied from 383 atoms (1532 bands) to 2046 atoms (8188 bands) and 8 task to 256 tasks with periodic boundary conditions.
- The largest system requires about 1.3TB of memory for calculations.
- CPU: Intel Xeon E5640 GPU: NVIDIA Tesla M2090





Random Phase Approximation

GPAW Random Phase Approximation (RPA) code:

- 6000 lines of python, 1000 lines of C/CUDA (and re-uses many GPAW functions)
- Better than DFT for correlated materials, but more computationally expensive
- Useful for oxides, Van der Waals systems, etc.

GPU Techniques:

- Use BLAS3 "zherk" instead of BLAS2 "zher"
- Batch FFTs
- GPU kernels parallelized over atoms/bands/projector-functions
- No thunking: all calculations on GPU

Preliminary ((GPU+CPU)/CPU) speedup for 202-electron N_2-on-Ru: 30x

Work by Jun Yan, Lin Li, Christopher O'Grady (Stanford/SLAC)

Summary

- We have accelerated the most numerically intensive parts of ground state DFT calculations
- Overall speed-ups in our tests varied from 8.8 to 19 depending on system size.
- Our multi-GPU implementation scales well even on large hybrid clusters.
- Code is available at GPAW Subversion repository.
- Acknowledgements to CSC and PRACE for computing resources

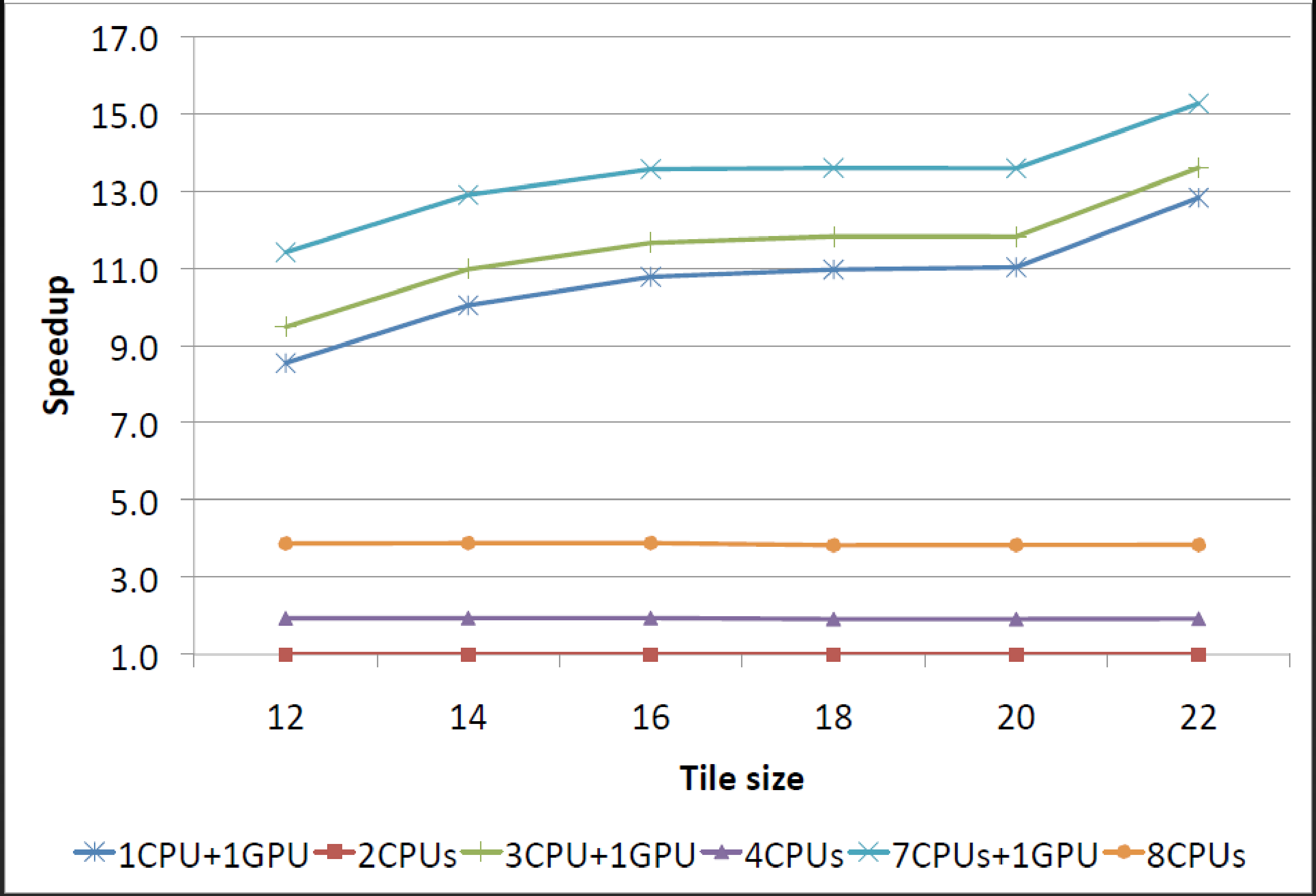
Hakala S., Havu V., Enkovaara J., Nieminen R. M. ”Parallel Electronic Structure Calculations Using Multiple Graphics Processing Units (GPUs)” In: Manninen, P., Öster, P. (eds.) PARA 2012. LNCS, vol. 7782, pp. 63--76. Springer, Heidelberg (2013)

NWChem

NWChem 6.3 Release with GPU Acceleration

- Addresses large complex and challenging molecular-scale scientific problems in the areas of catalysis, materials, geochemistry and biochemistry on highly scalable, parallel computing platforms to obtain the fastest time-to-solution
- Researchers can for the first time be able to perform large scale coupled cluster with perturbative triples calculations utilizing the NVIDIA GPU technology. A highly scalable multi-reference coupled cluster capability will also be available in NWChem 6.3.
- The software, released under the Educational Community License 2.0, can be downloaded from the NWChem website at www.nwchem-sw.org

NWChem - Speedup of the non-iterative calculation for various configurations/tile sizes



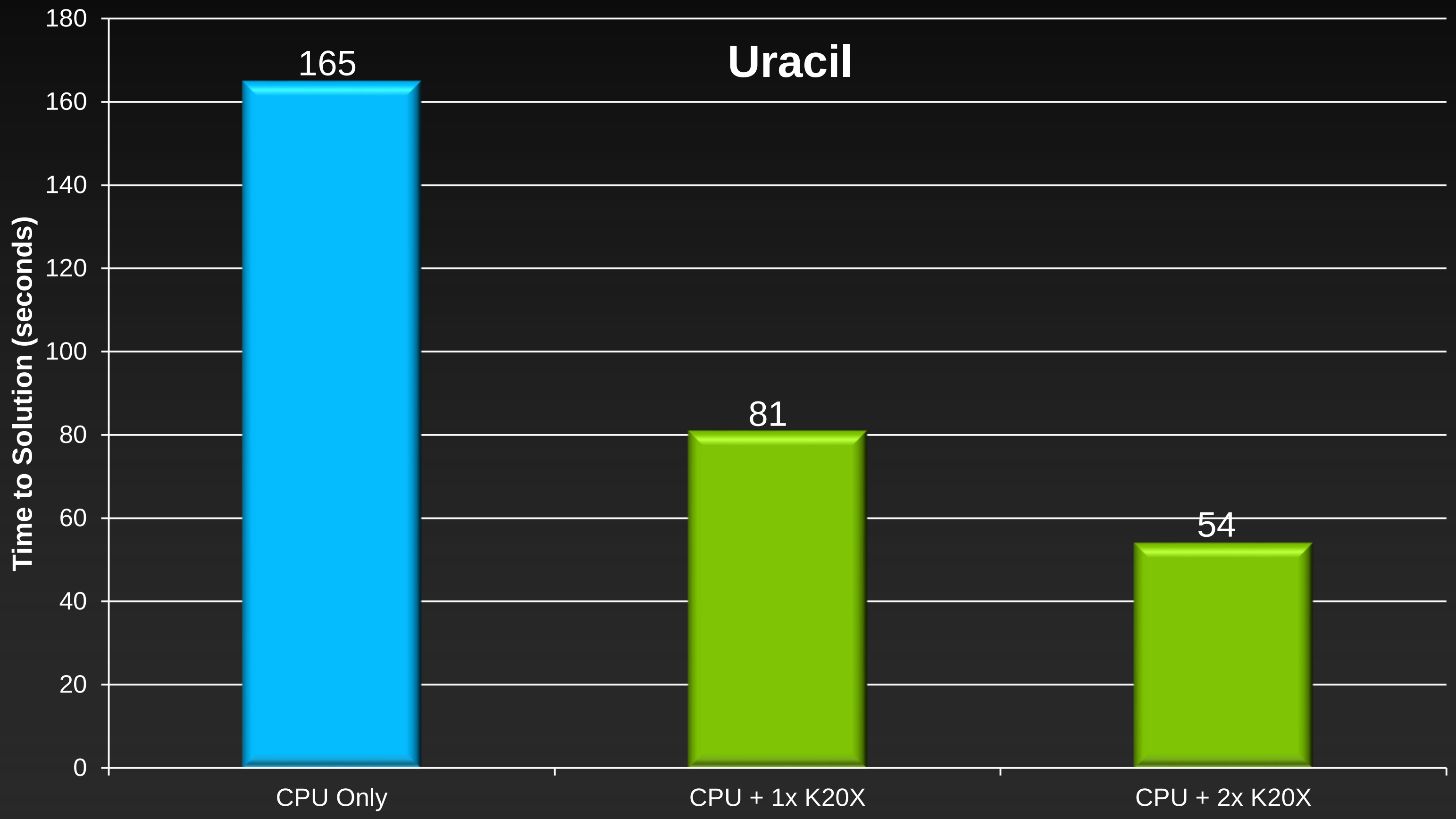
System: cluster consisting of dual-socket nodes constructed from:

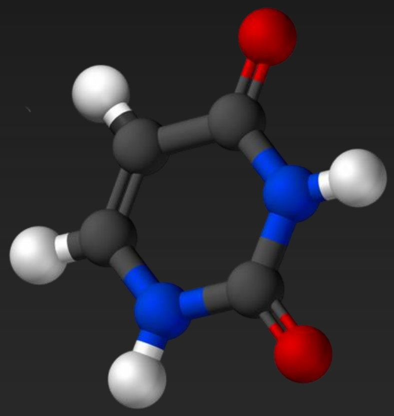
- 8-core AMD Interlagos processors
- 64 GB of memory
- Tesla M2090 (Fermi) GPUs

The nodes are connected using a high-performance QDR Infiniband interconnect

Courtesy of Kowolski, K., Bhaskaran-Nair, at al @ PNNL, JCTC (submitted)

Kepler, Faster Performance (NWChem)



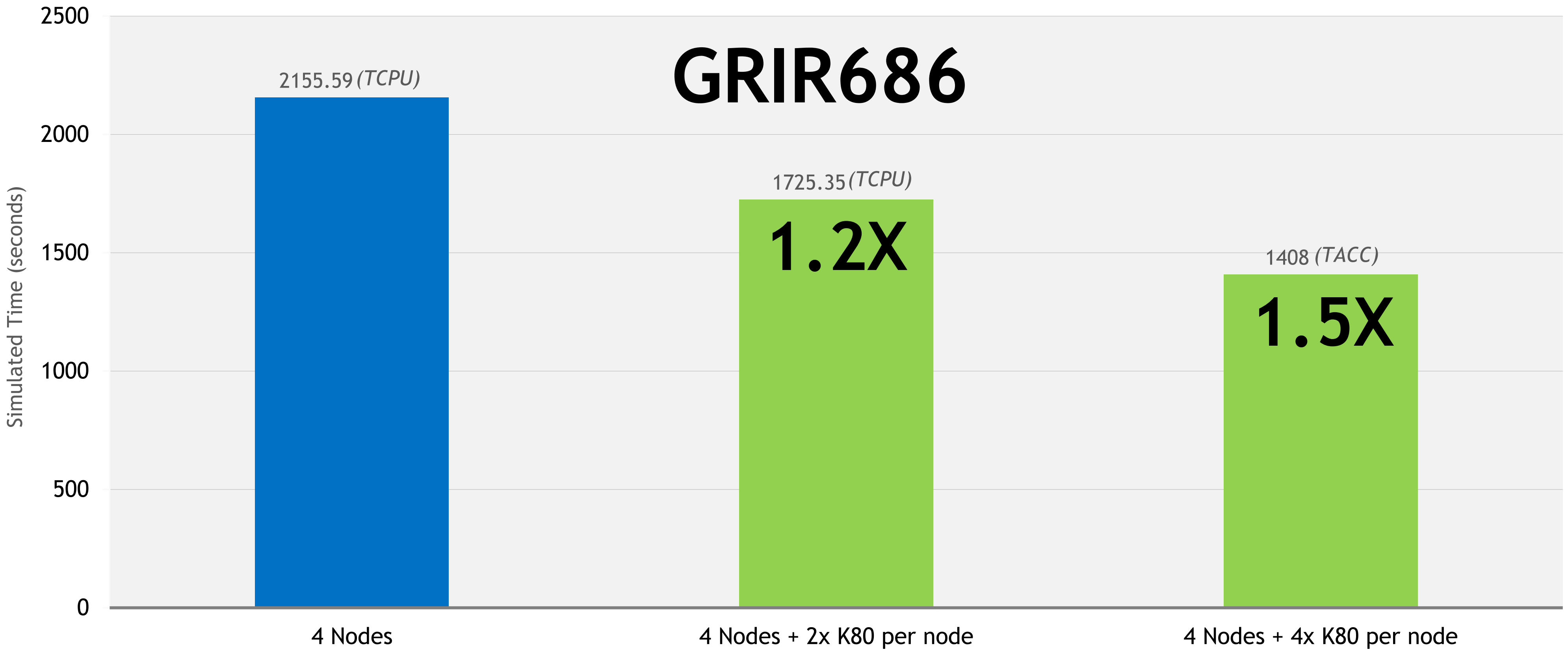


Uracil Molecule

Performance improves by 2x with one GPU and by 3.1x with 2 GPUs

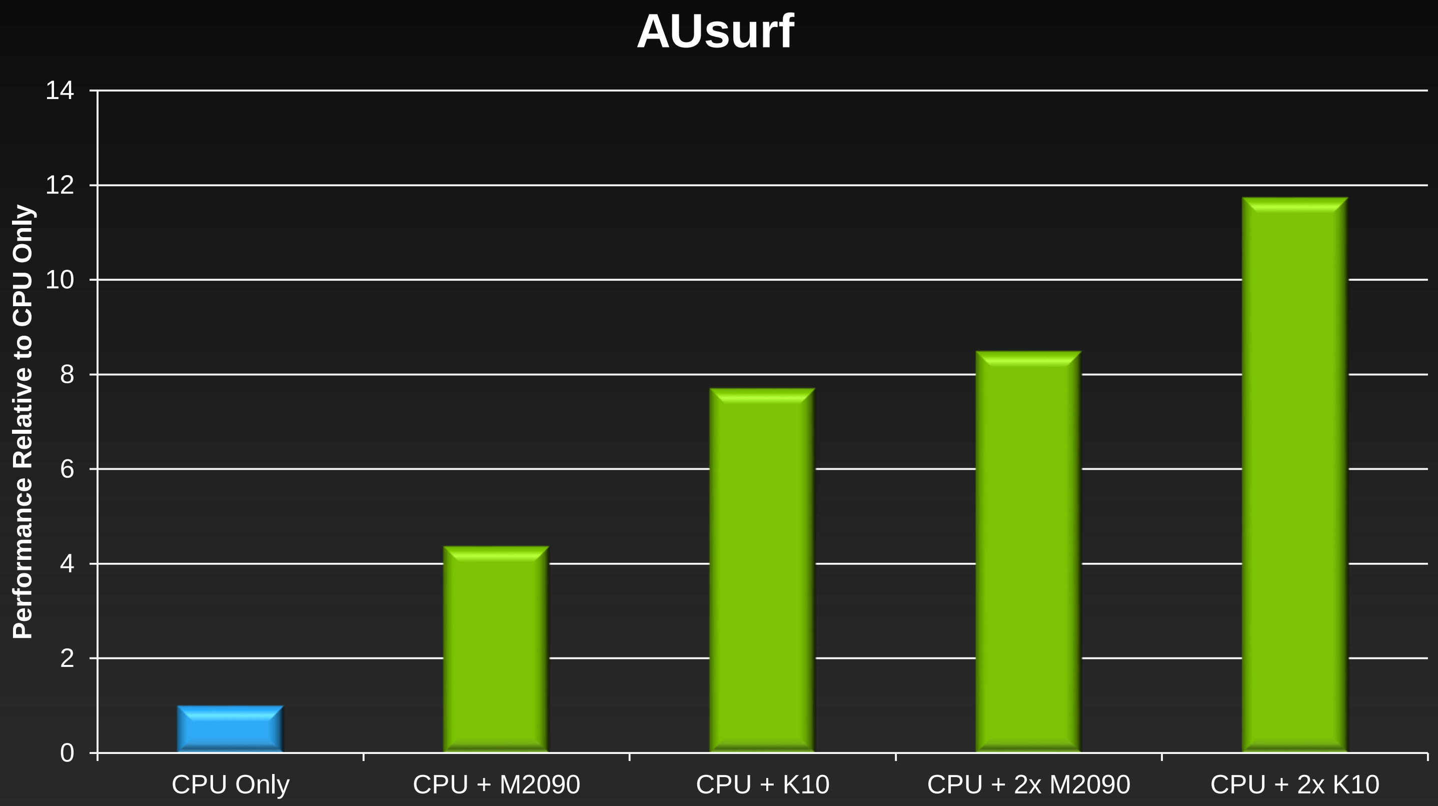
Quantum Espresso/PWscf

Quantum Espresso/PWscf 5.3



Blue node contains Dual Intel Xeon E5-2698 v3 @ 2.30GHz (Haswell) CPUs
Green nodes contain Dual Intel Xeon E5-2698 v3 @ 2.30GHz (Haswell) CPUs + Tesla K80 (autoboost) GPUs

Kepler, fast science



Running Quantum Espresso version 5.0-build7 on CUDA 5.0.36

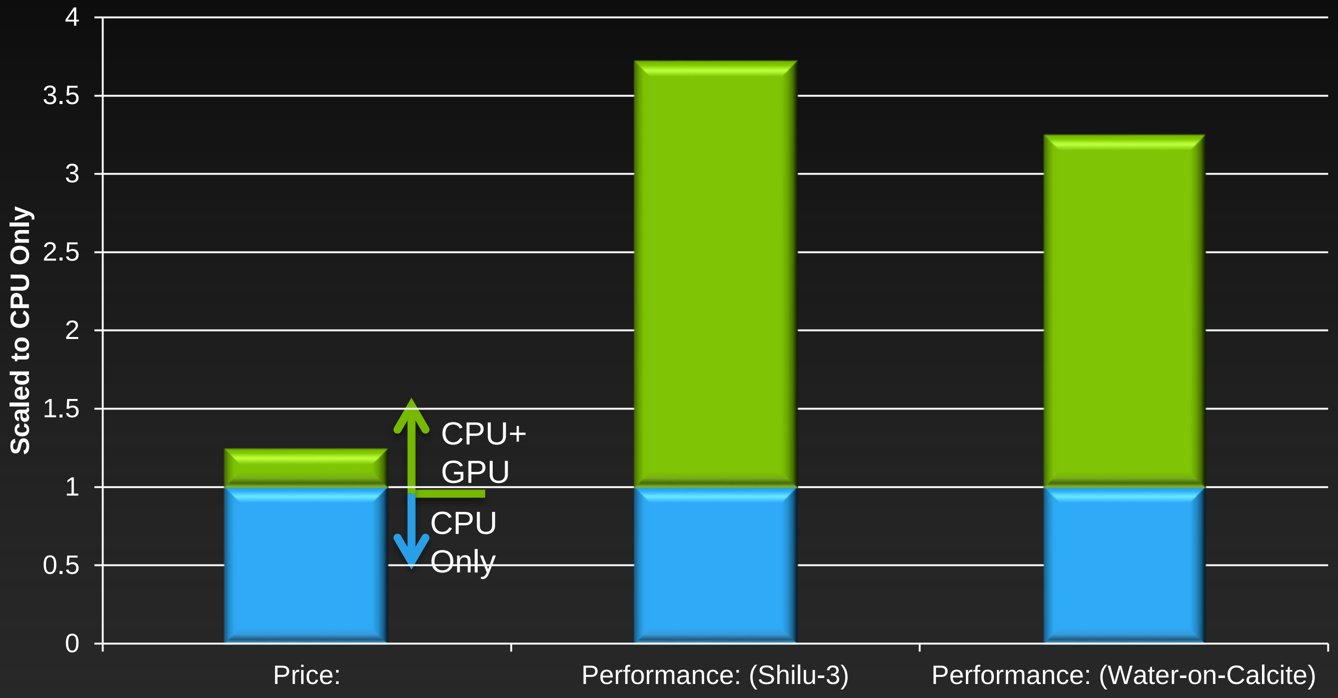
The blue node contains 2 E5-2687W CPUs (150W, 8 Cores per CPU).

The green nodes contain 2 E5-2687W CPUs and 1 or 2 NVIDIA M2090 or K10 GPUs (225W and 235W respectively).

Using K10s delivers up to 11.7x the performance per node over CPUs

And 1.7x the performance when compared to M2090s

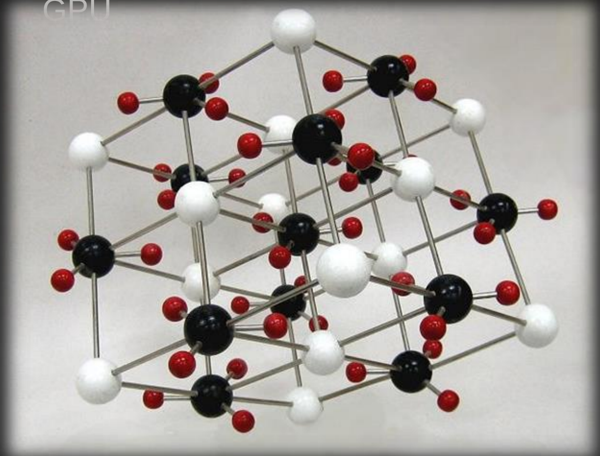
Extreme Performance/Price from 1 GPU



Quantum Espresso Simulations run on FERMI @ ICHEC.

A 6-Core 2.66 GHz Intel X5650 was used for the CPU

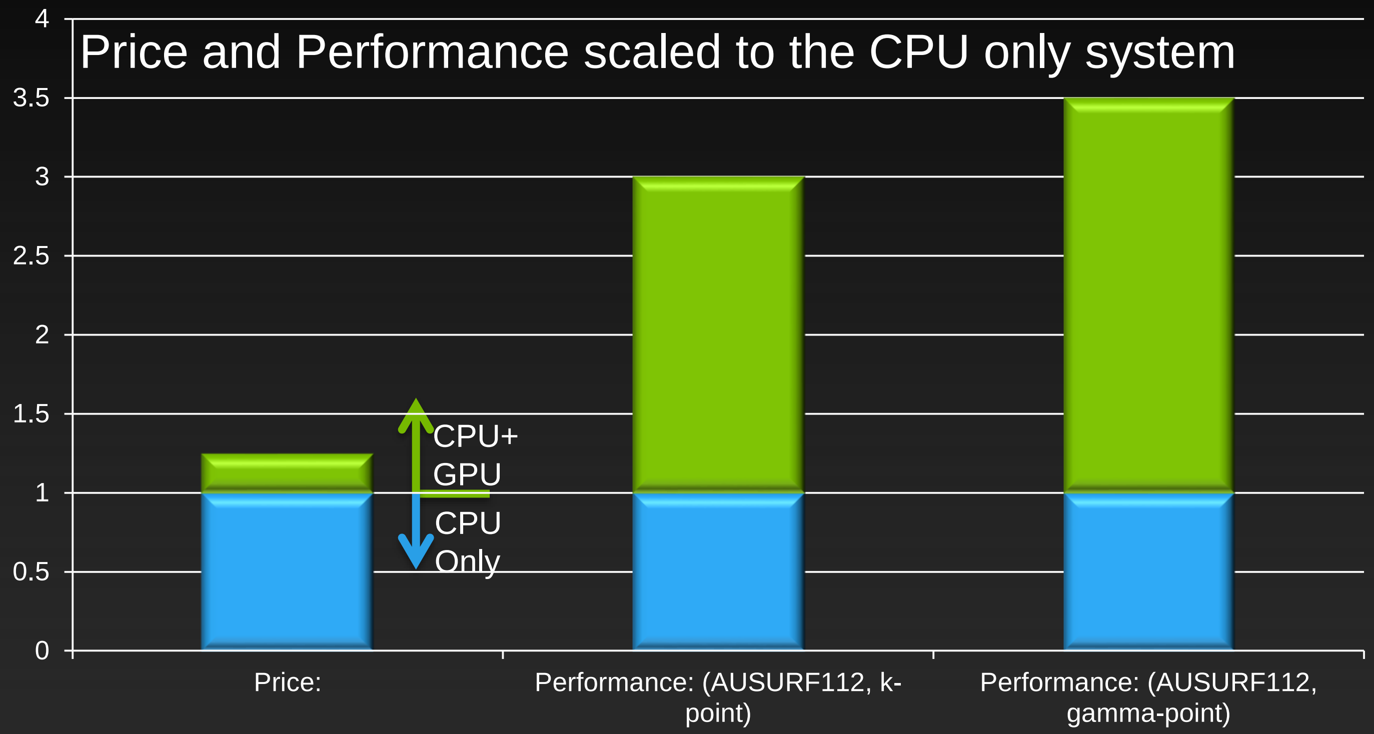
An NVIDIA C2050 was used for the GPU



Calcite structure

Adding a GPU can improve performance by 3.7x while only increasing price by 25%

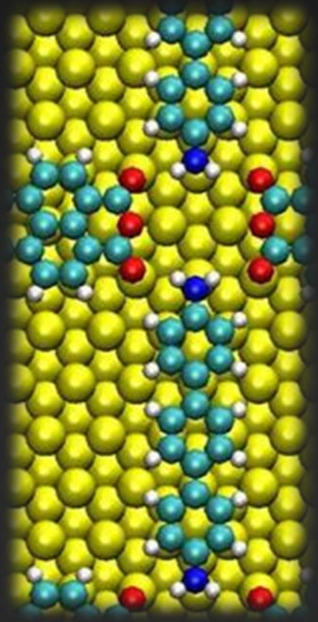
Extreme Performance/Price from 1 GPU



Quantum Espresso Simulations run on FERMI @ ICHEC.

A 6-Core 2.66 GHz Intel X5650 was used for the CPU

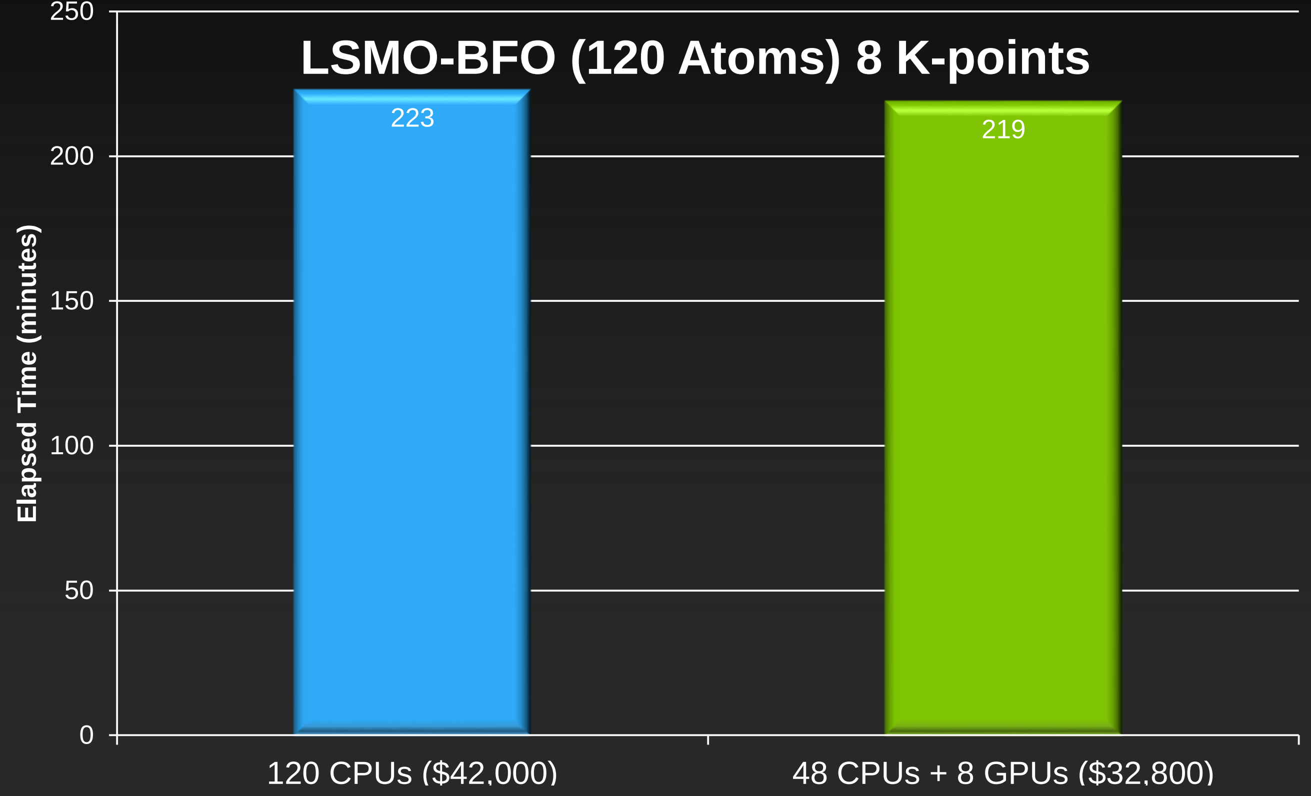
An NVIDIA C2050 was used for the GPU



Calculation done for a gold surface of 112 atoms

Adding a GPU can improve performance by 3.5x while only increasing price by 25%

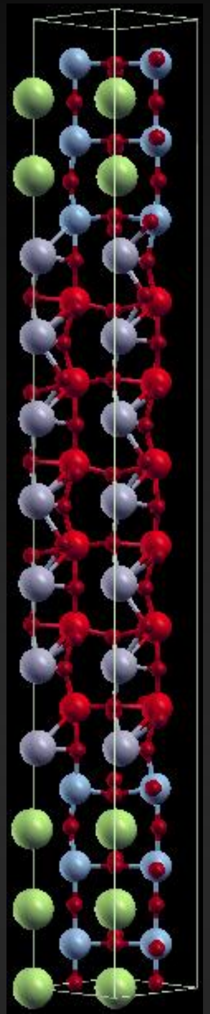
Replace 72 CPUs with 8 GPUs



Quantum Espresso Simulations run on PLX @ CINECA.

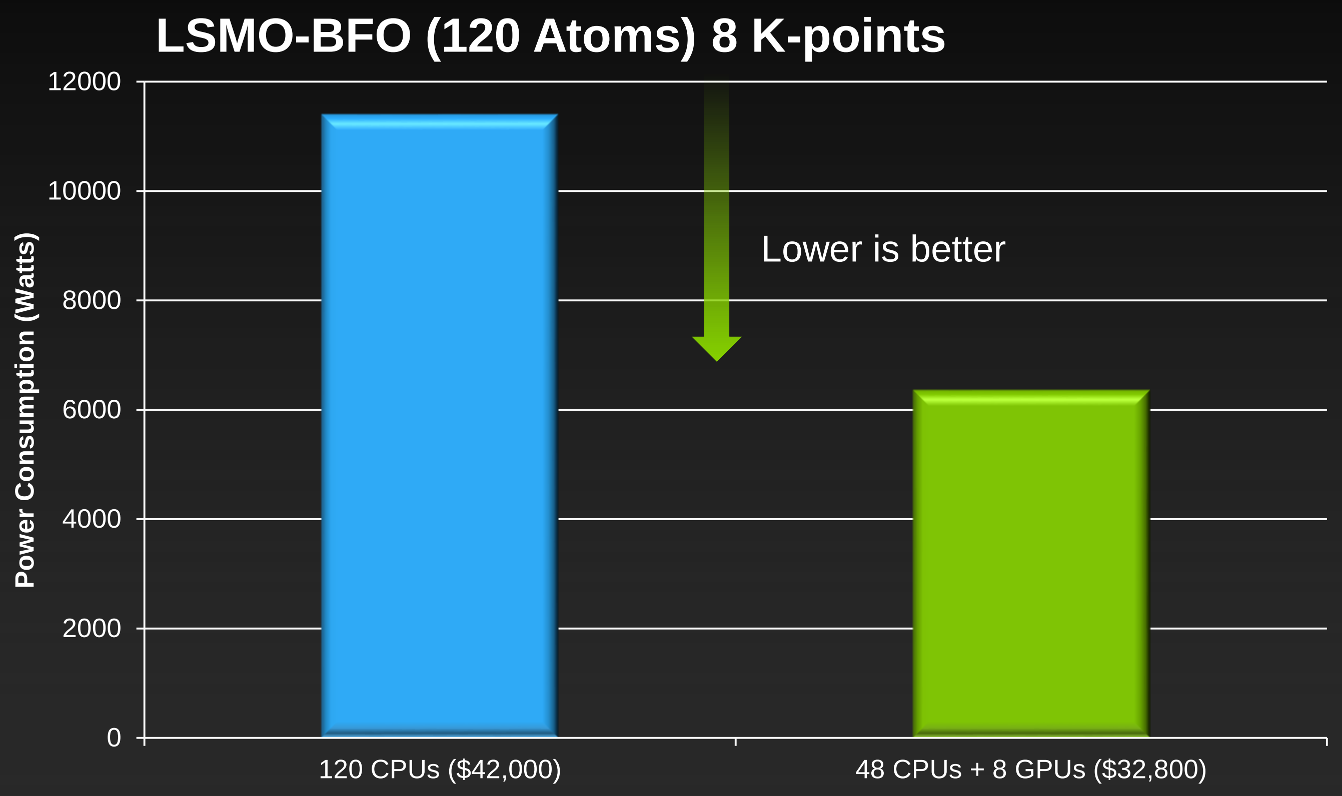
Intel 6-Core 2.66 GHz X5550 were used for the CPUs

NVIDIA M2070s were used for the GPUs



The GPU Accelerated setup performs faster and costs 24% less

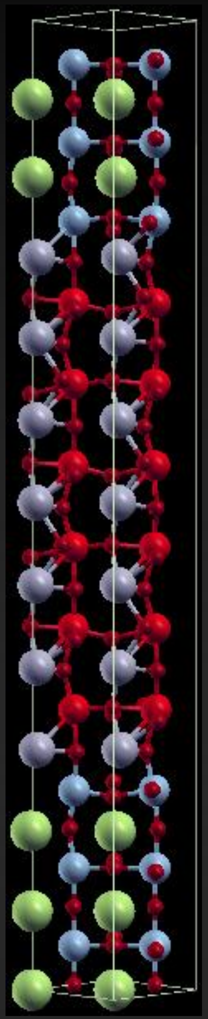
QE/PWscf - Green Science



Quantum Espresso Simulations run on PLX @ CINECA.

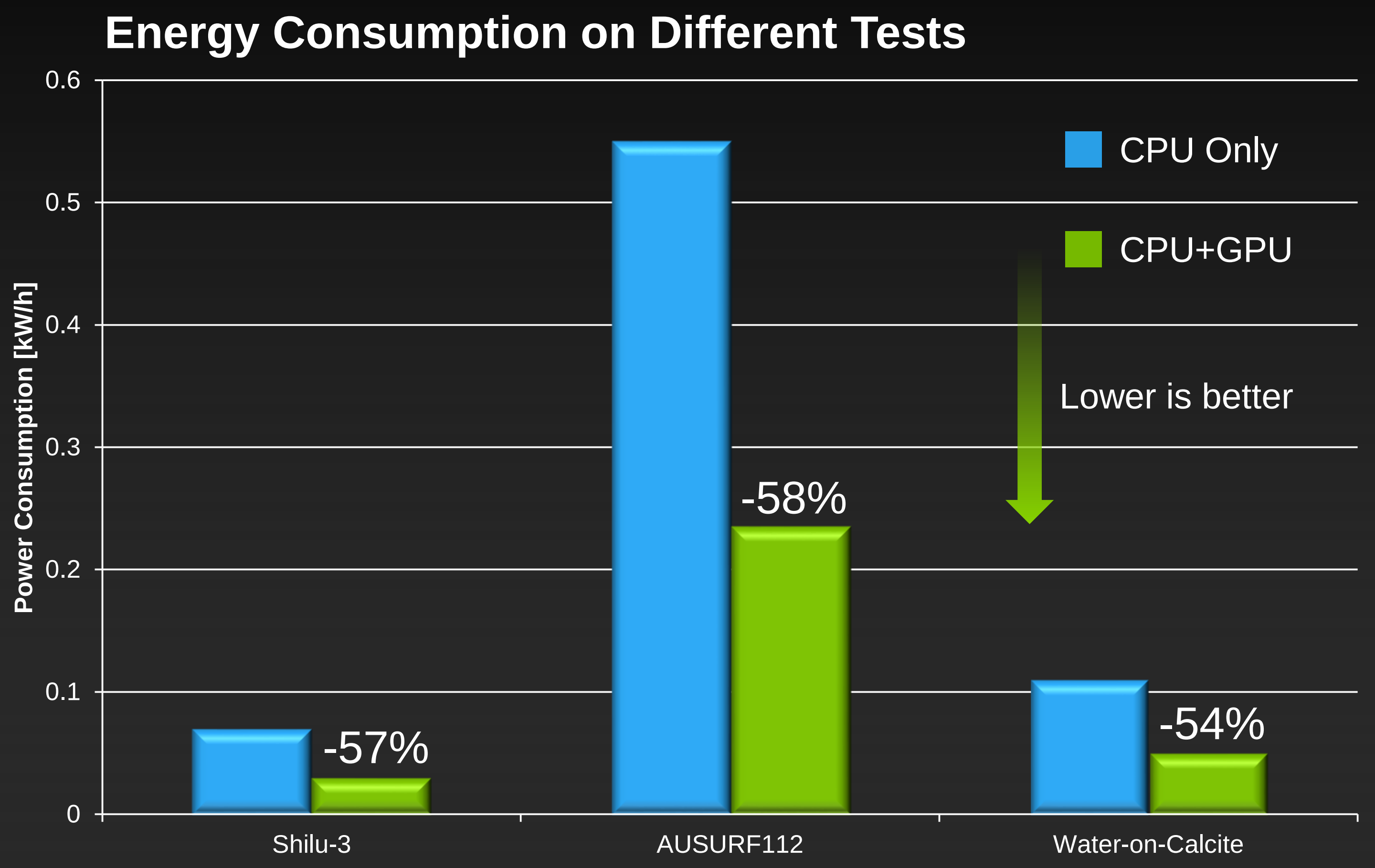
Intel 6-Core 2.66 GHz X5550 were used for the CPUs

NVIDIA M2070s were used for the GPUs



Over a year, the lower power consumption would save $4300 on energy bills

NVIDIA GPUs Use Less Energy



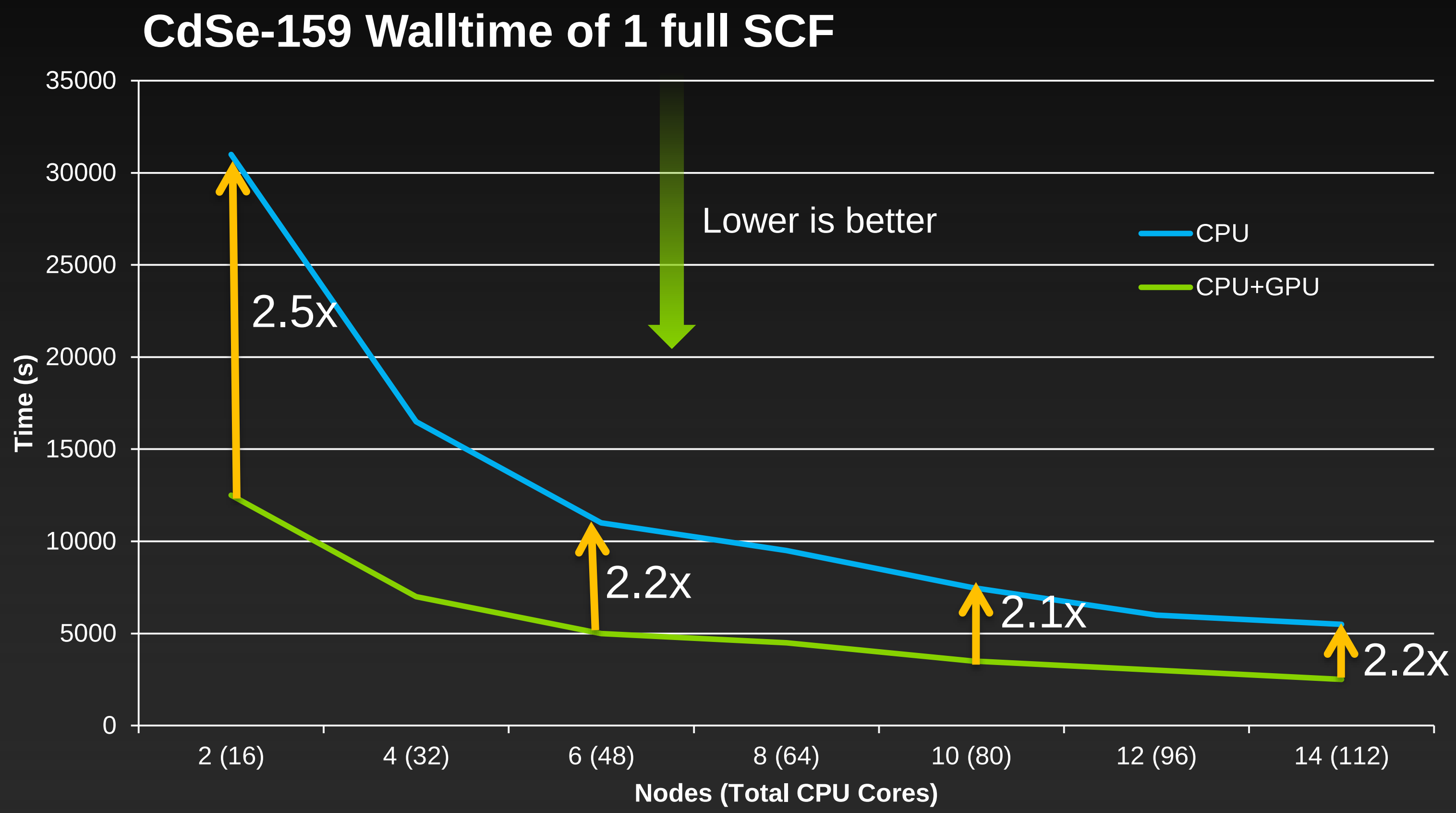
Quantum Espresso Simulations run on FERMI @ ICHEC.

A 6-Core 2.66 GHz Intel X5650 was used for the CPU

An NVIDIA C2050 was used for the GPU

In all tests, the GPU Accelerated system consumed less than half the power as the CPU Only

QE/PWscf - Great Strong Scaling in Parallel

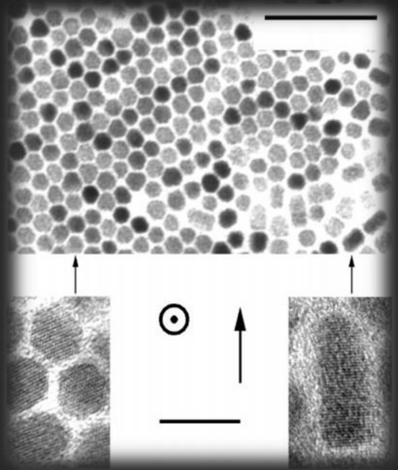


Speedups up to 2.5x with GPU Accelerations

Quantum Espresso Simulations run on STONEY @ ICHEC.

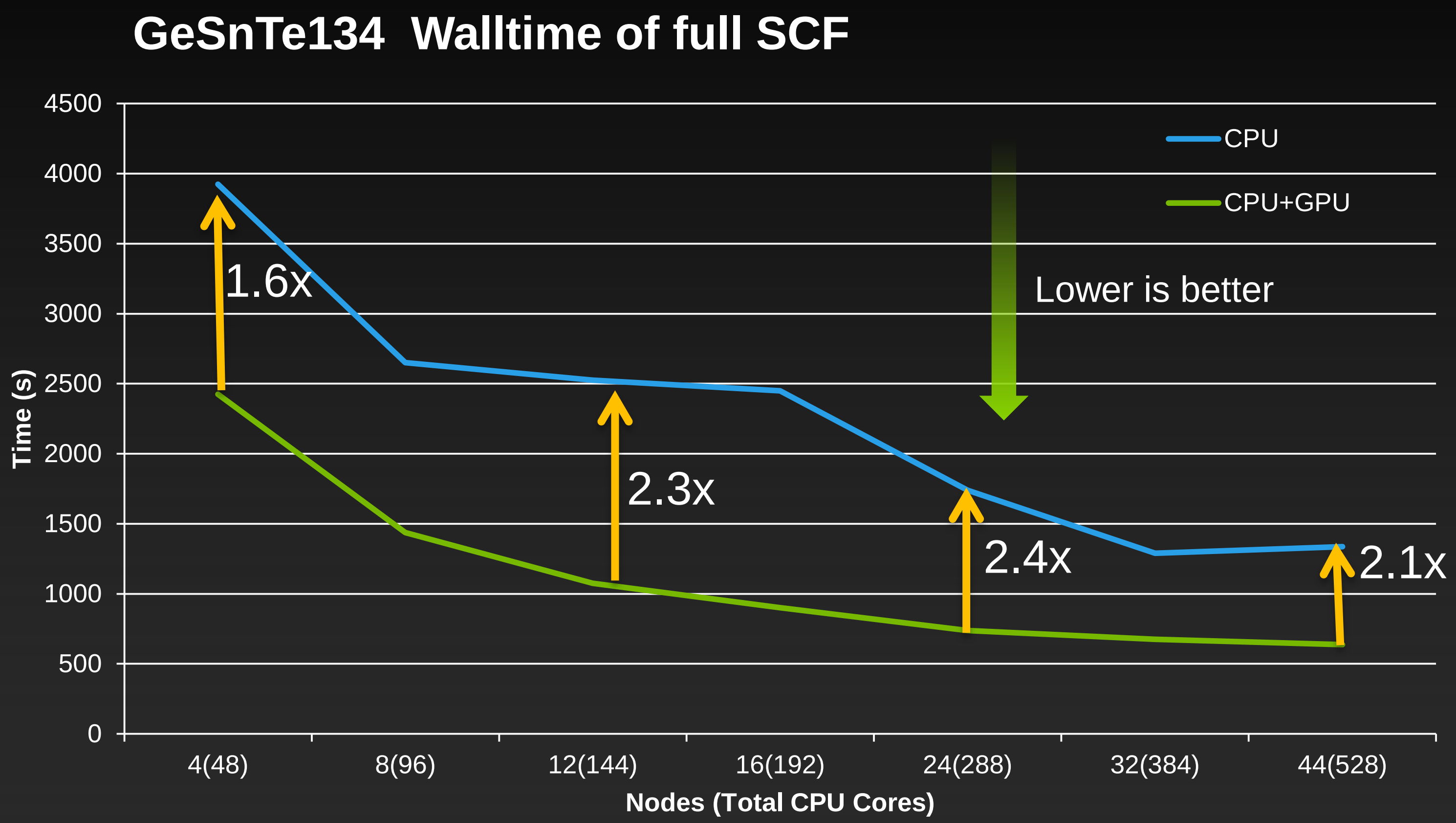
Two quad core 2.87 GHz Intel X5560s were used in each node

Two NVIDIA M2090s were used in each node for the CPU+GPU test



159 Cadmium Selenide nanodots

QE/PWscf - More Powerful Strong Scaling



Quantum Espresso Simulations run on PLX @ CINECA.

Two 6-Core 2.4 GHz Intel E5645s were used in each node

Two NVIDIA M2070s were used in each node for the CPU+GPU test

Accelerate your cluster by up to 2.1x with NVIDIA GPUs

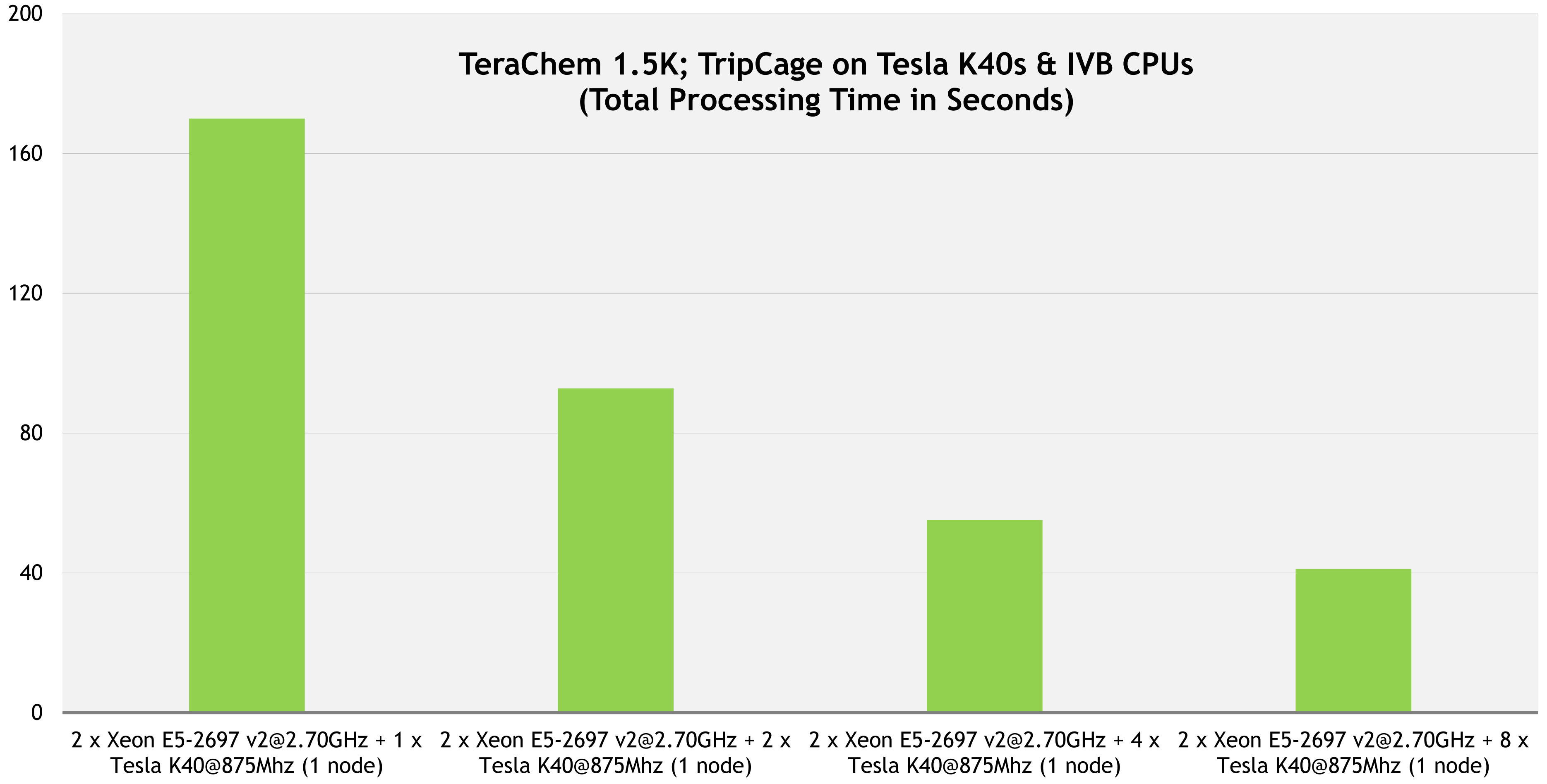
Try GPU accelerated Quantum Espresso for free – www.nvidia.com/GPUTestDrive

TeraChem 1.5K

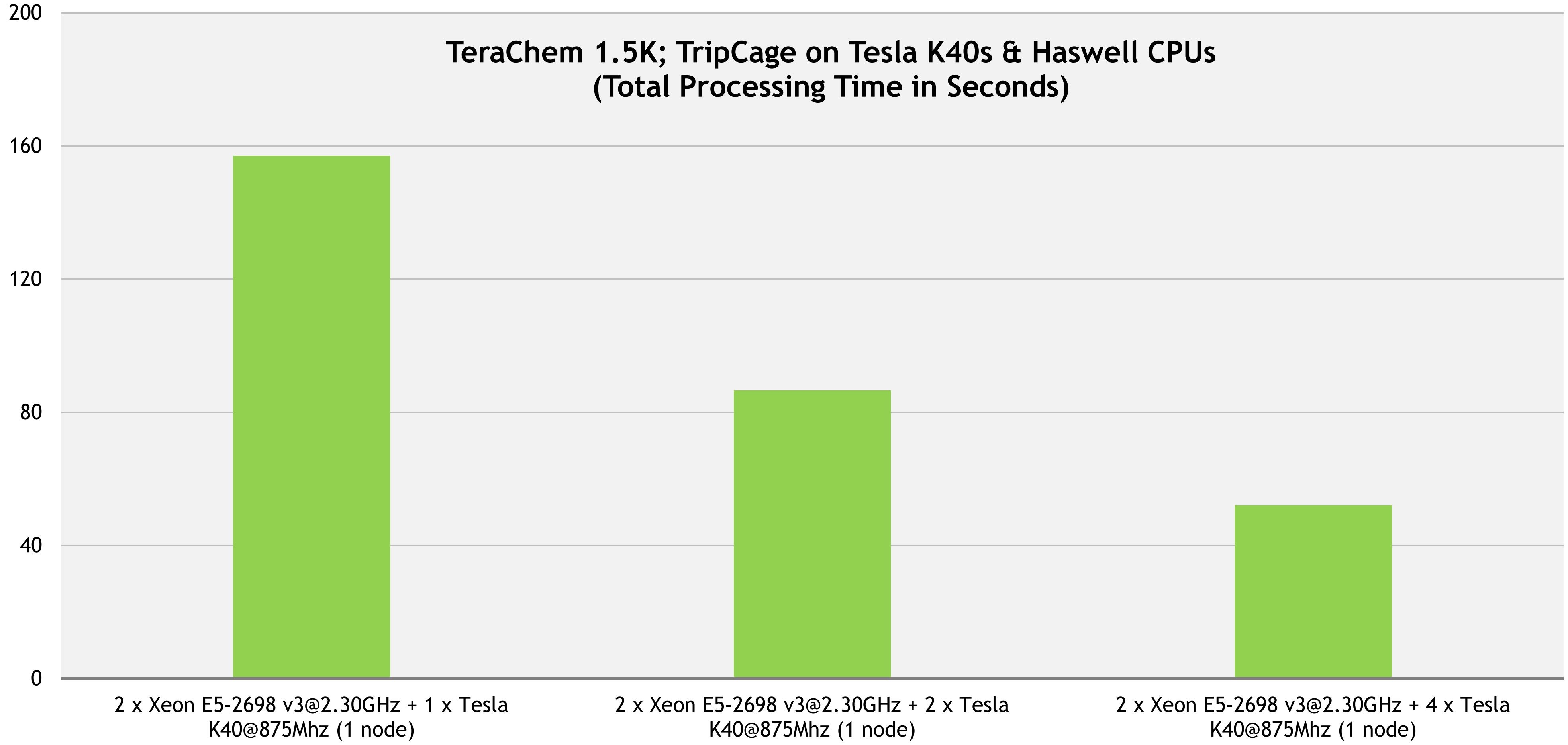
Speaker, Date

NVIDIA

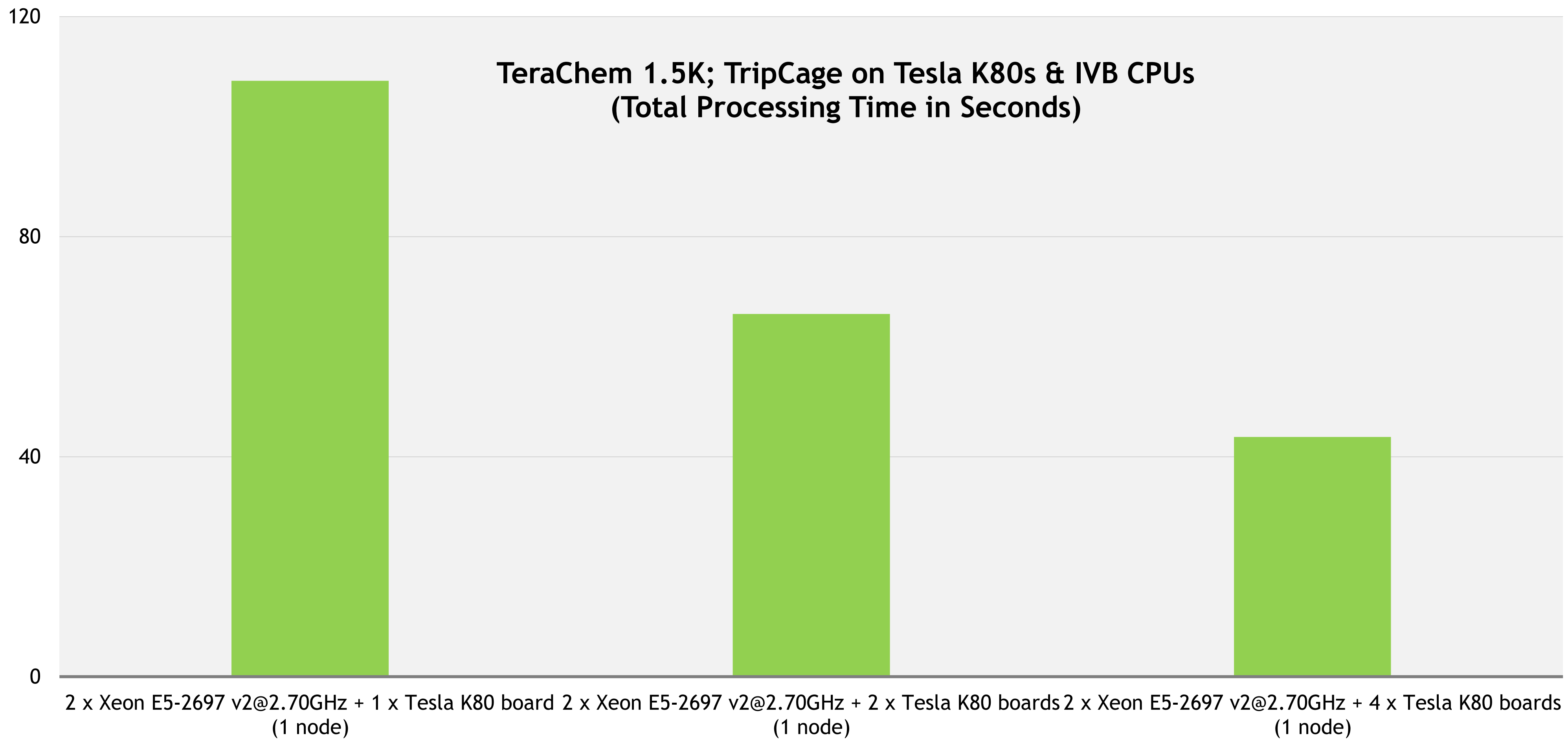
TERACHEM 1.5K; TRIPCAGE ON TESLA K40S



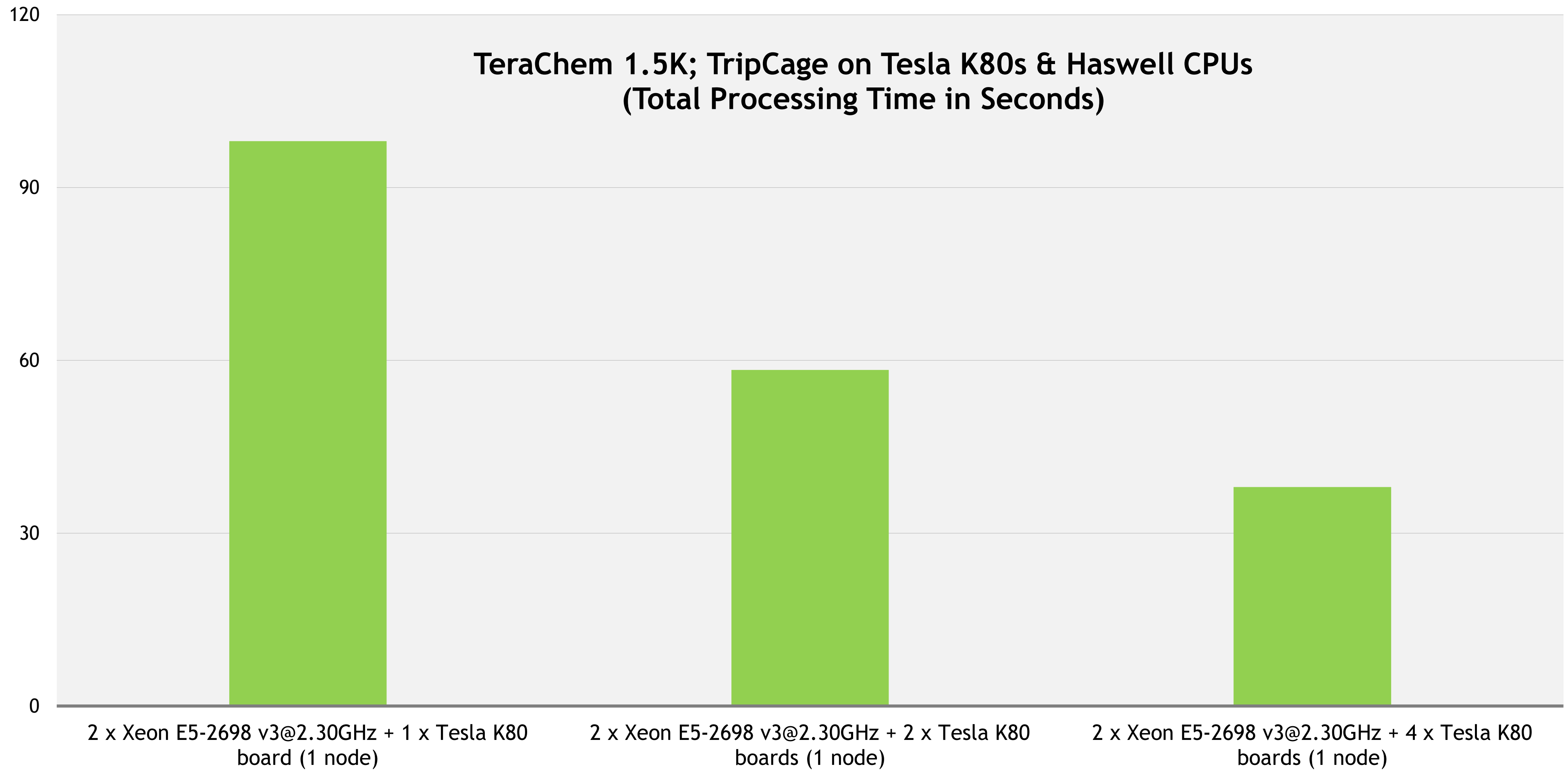
TERACHEM 1.5K; TRIPCAGE ON TESLA K40S & HASWELL CPUS



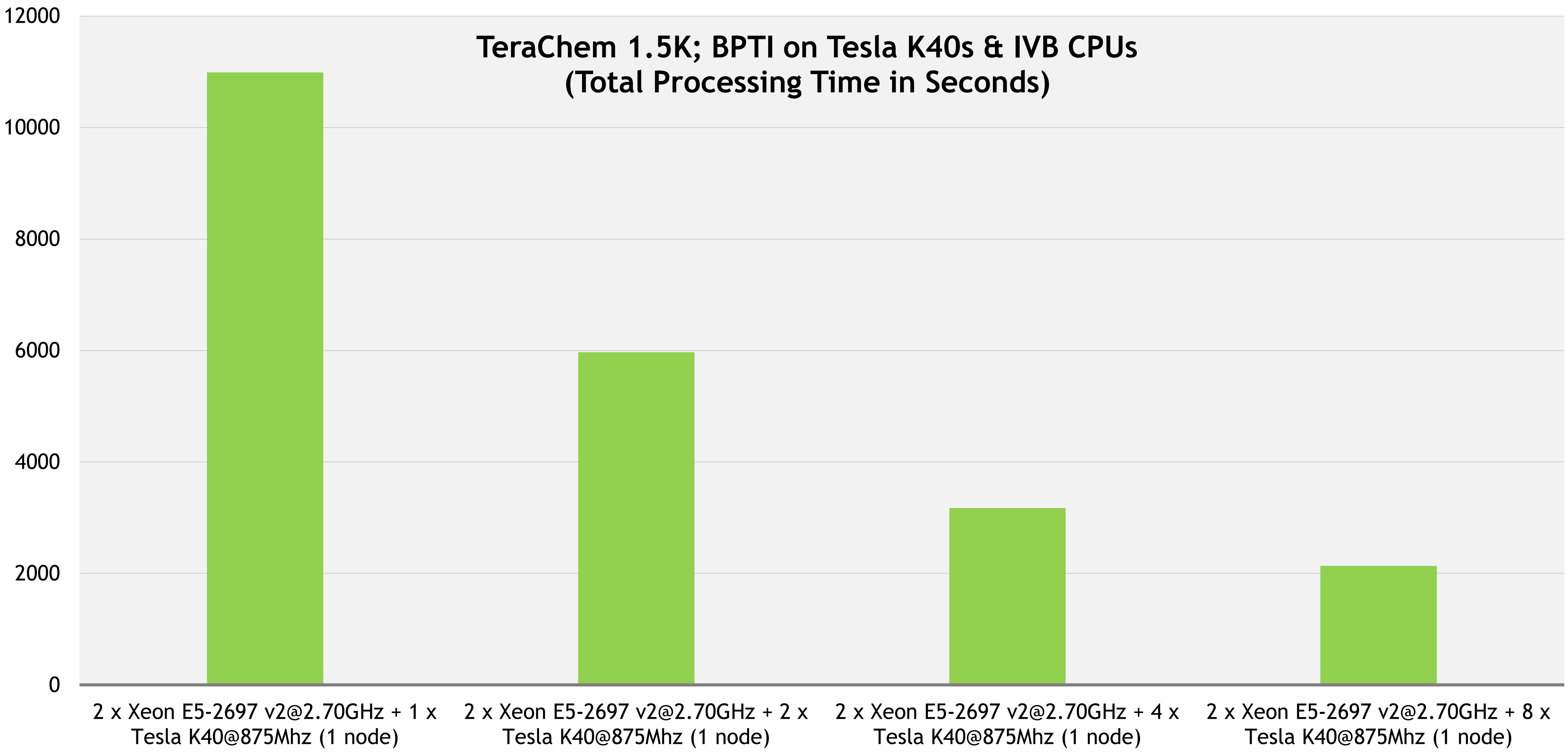
TERACHEM 1.5K; TRIPCAGE ON TESLA K80S & IVB CPUS



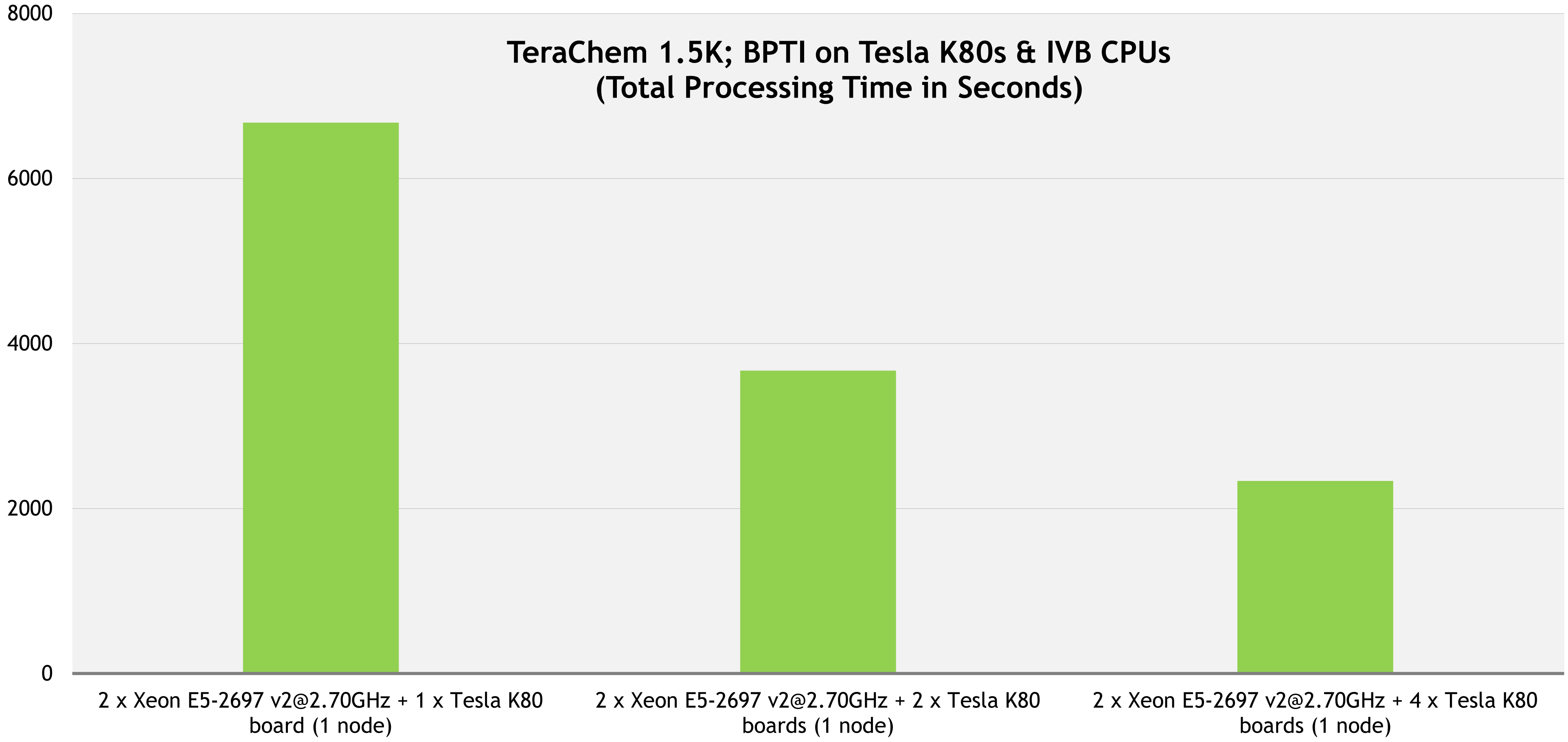
TERACHEM 1.5K; TRIPCAGE ON TESLA K80S & HASWELL CPUS



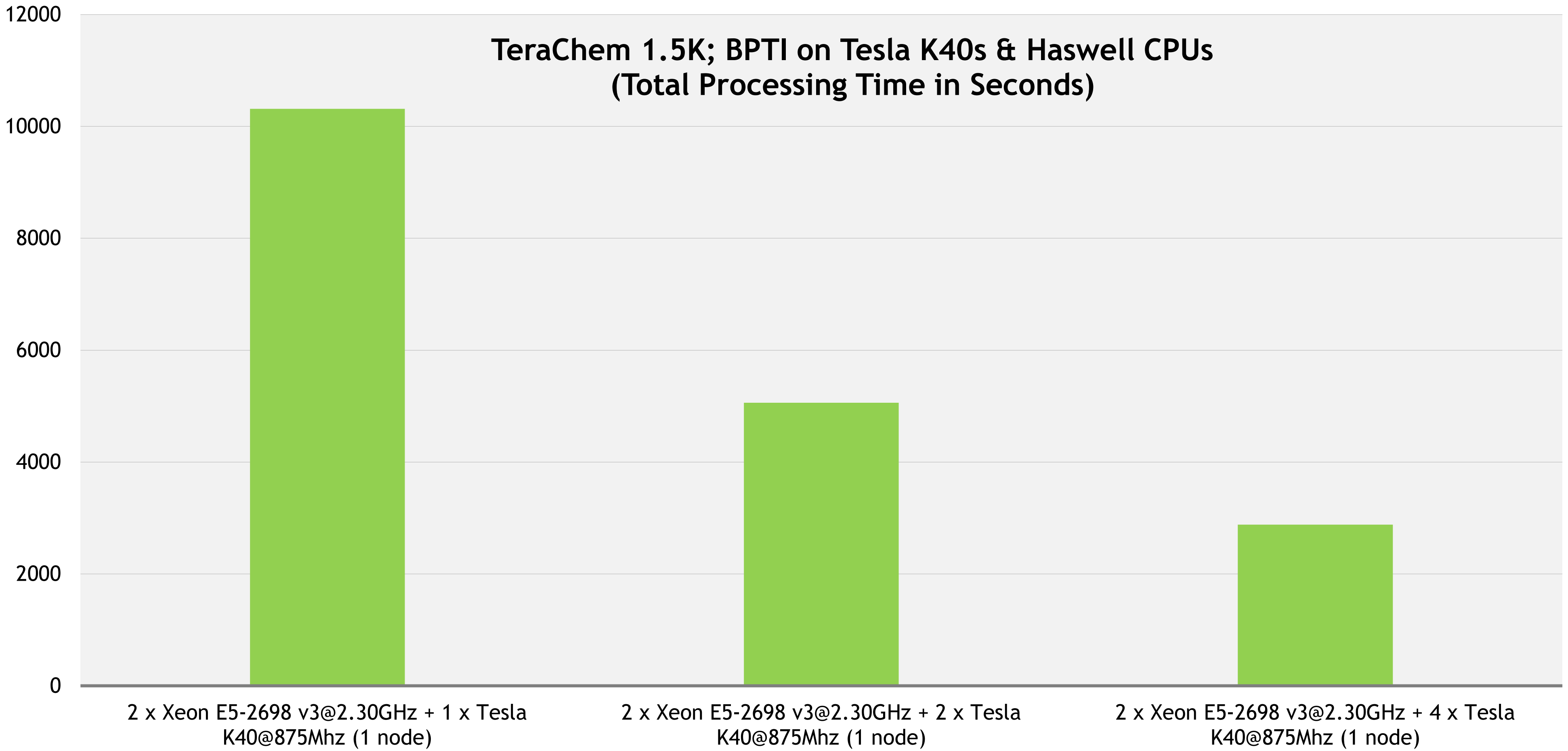
TERACHEM 1.5K; BPTI ON TESLA K40S & IVB CPUS



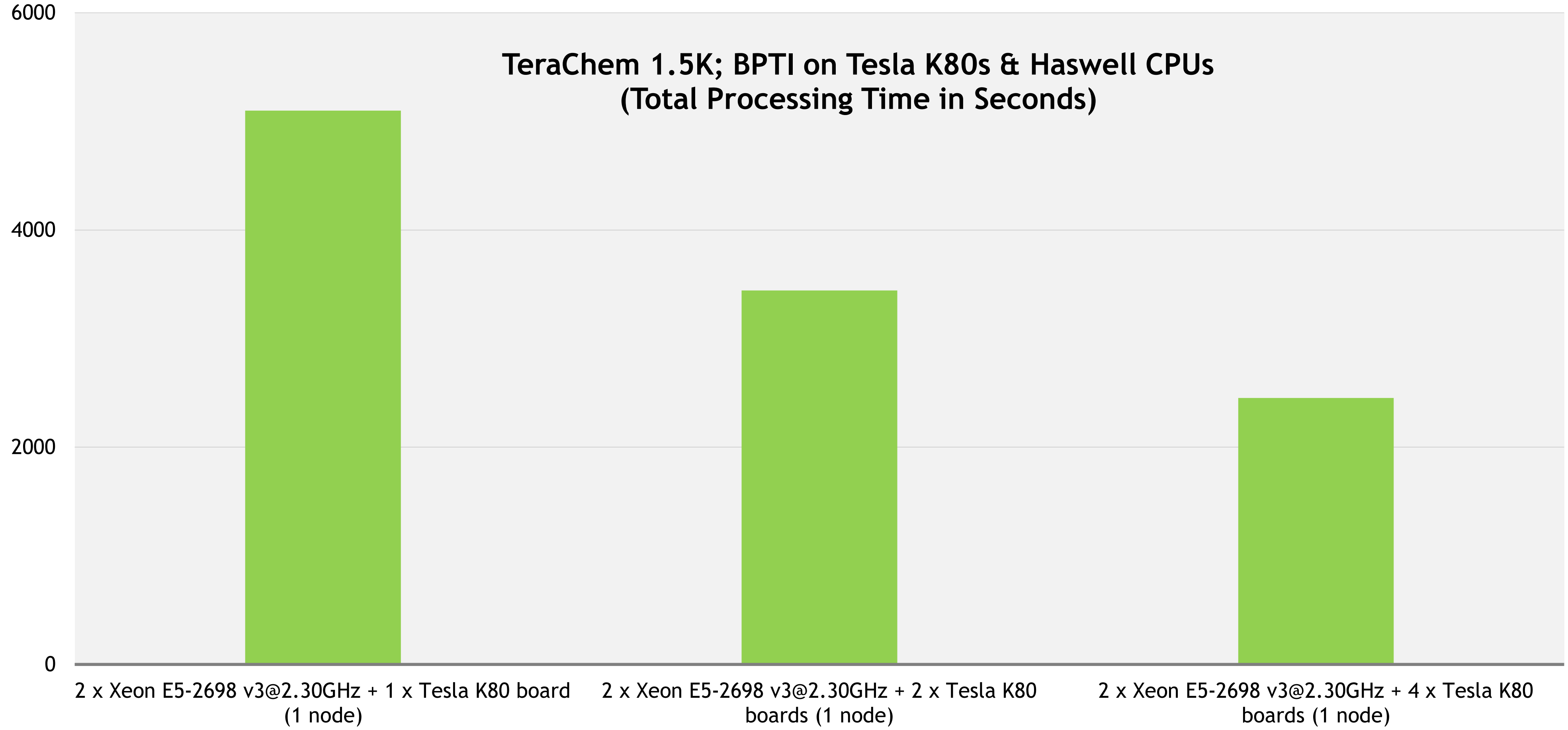
TERACHEM 1.5K; BPTI ON TESLA K80S & IVB CPUS



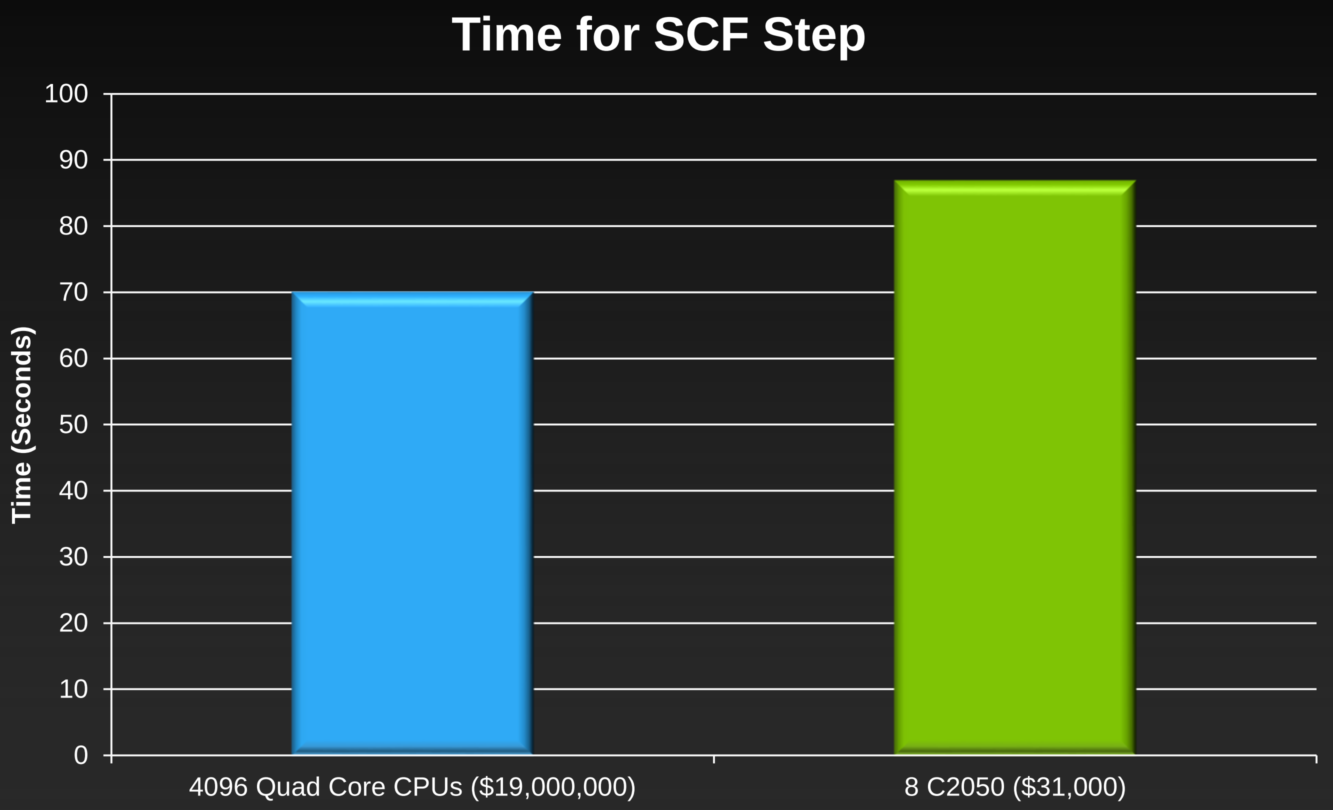
TERACHEM 1.5K; BPTI ON TESLA K40S & HASWELL CPUS



TERACHEM 1.5K; BPTI ON TESLA K80S & HASWELL CPUS



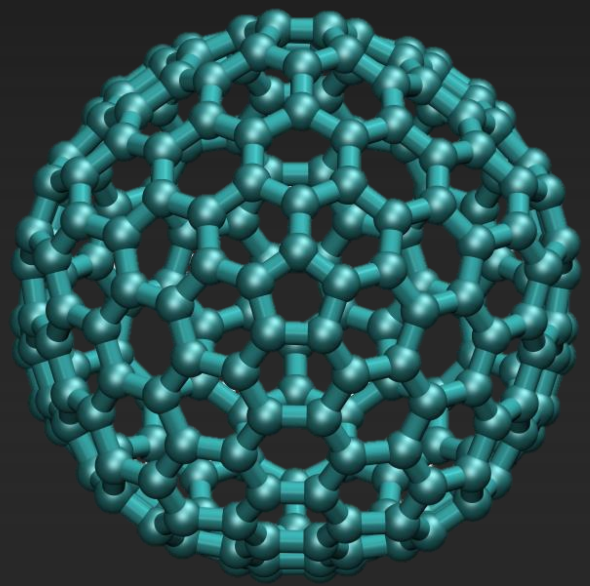
TeraChem
Supercomputer Speeds on GPUs



TeraChem running on 8 C2050s on 1 node

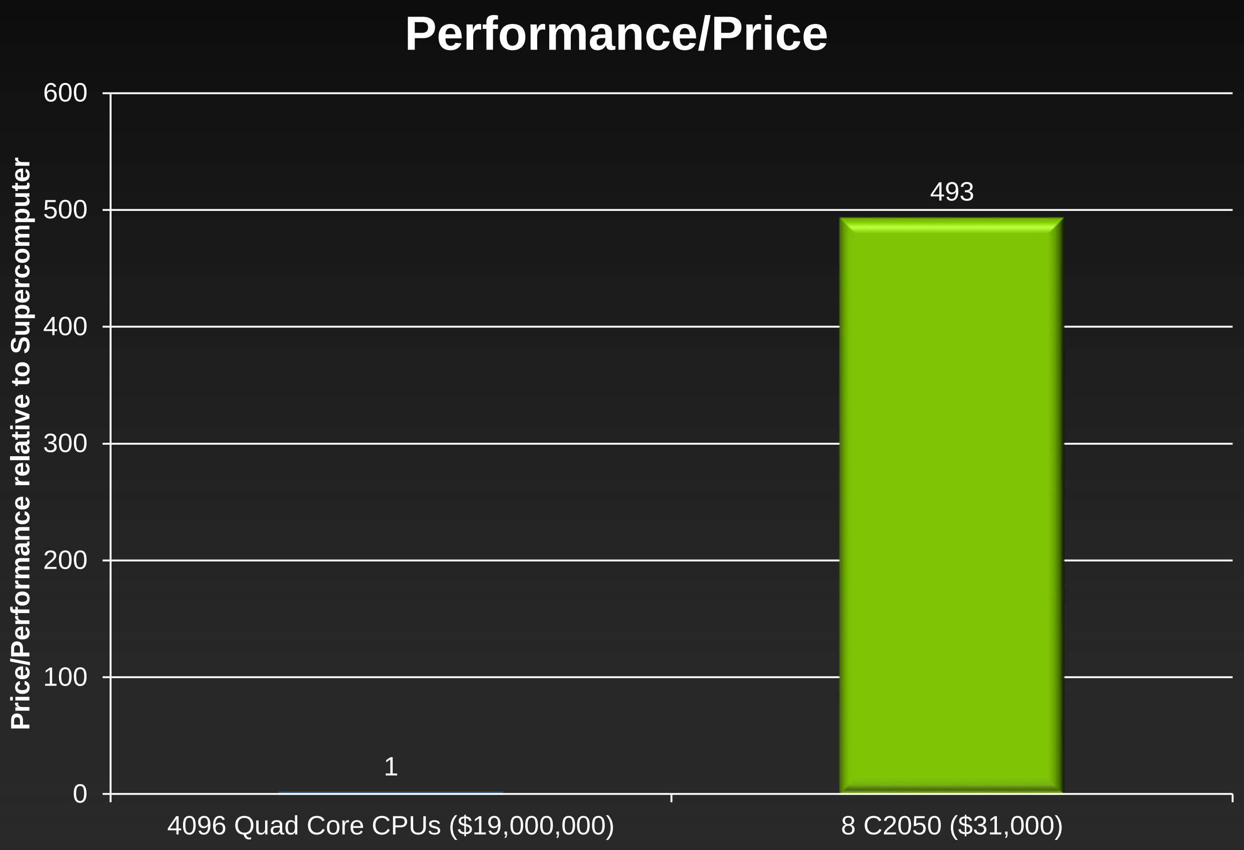
NWChem running on 4096 Quad Core CPUs
In the Chinook Supercomputer

Giant Fullerene C240 Molecule



Similar performance from just a handful of GPUs

TeraChem
Bang for the Buck

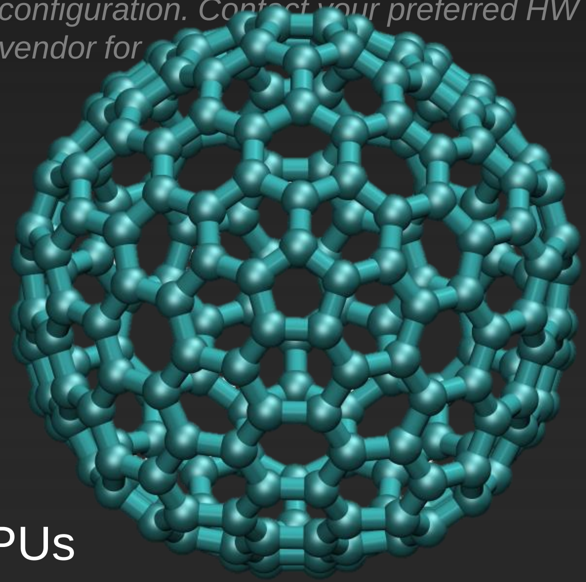


TeraChem running on 8 C2050s on 1 node

NWChem running on 4096 Quad Core CPUs
In the Chinook Supercomputer

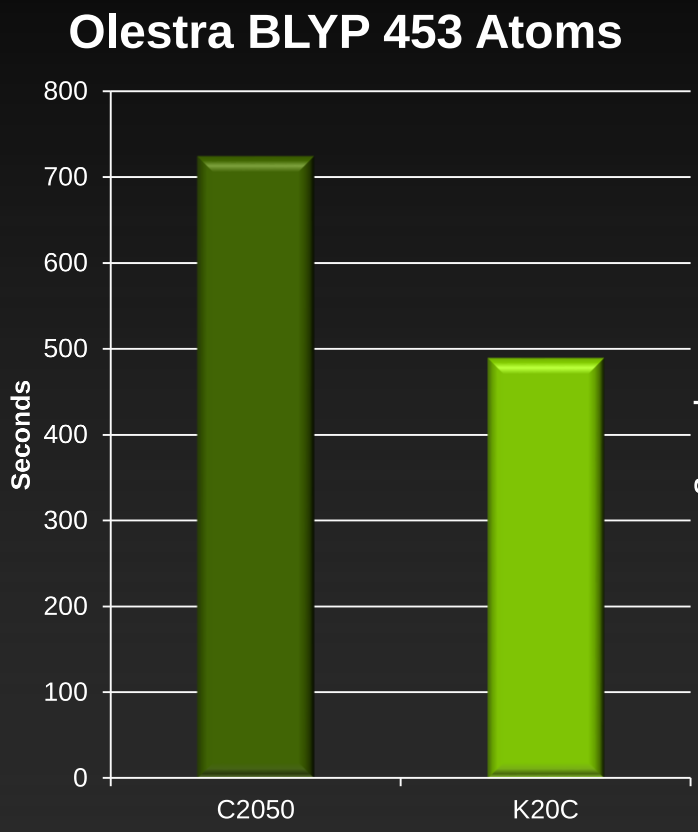
Giant Fullerene C240 Molecule

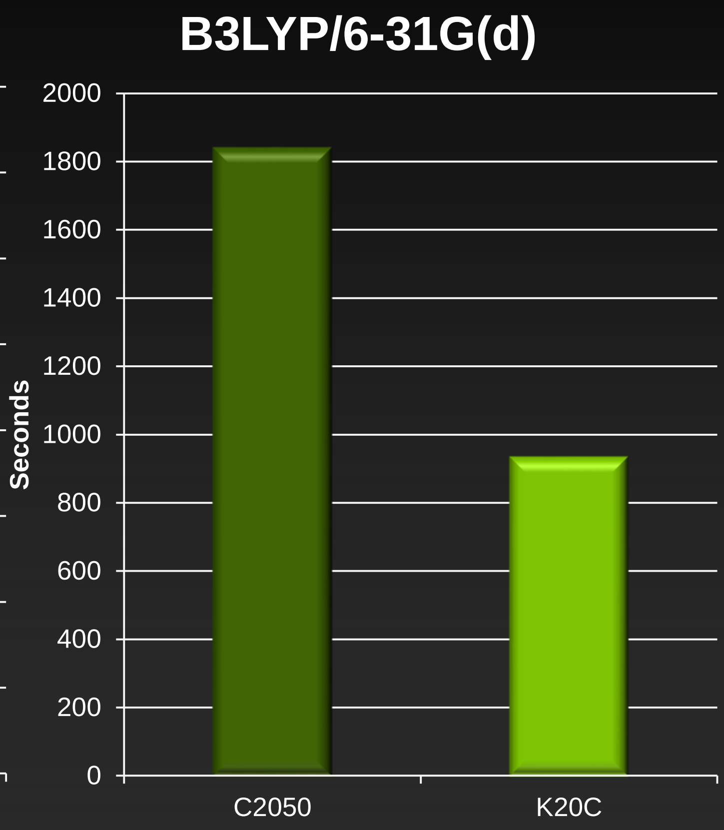
Note: Typical CPU and GPU node pricing used. Pricing may vary depending on node configuration. Contact your preferred HW vendor for



Dollars spent on GPUs do 500x more science than those spent on CPUs

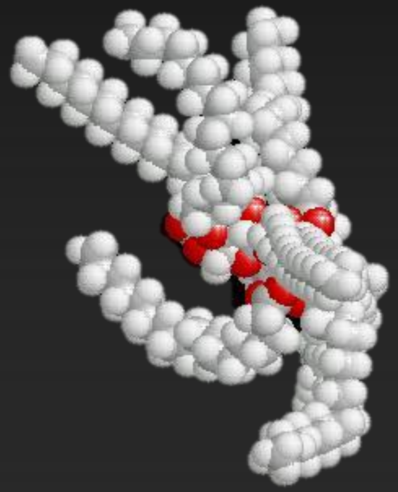
Kepler’s Even Better





TeraChem running on C2050 and K20C

First graph is of BLYP/G-31(d)
Second is B3LYP/6-31G(d)



Kepler performs 2x faster than Tesla

VASP

Select Slides from “VASP: A Case Study for Accelerating Plane Wave DFT Codes” at GTC March 2014 follow

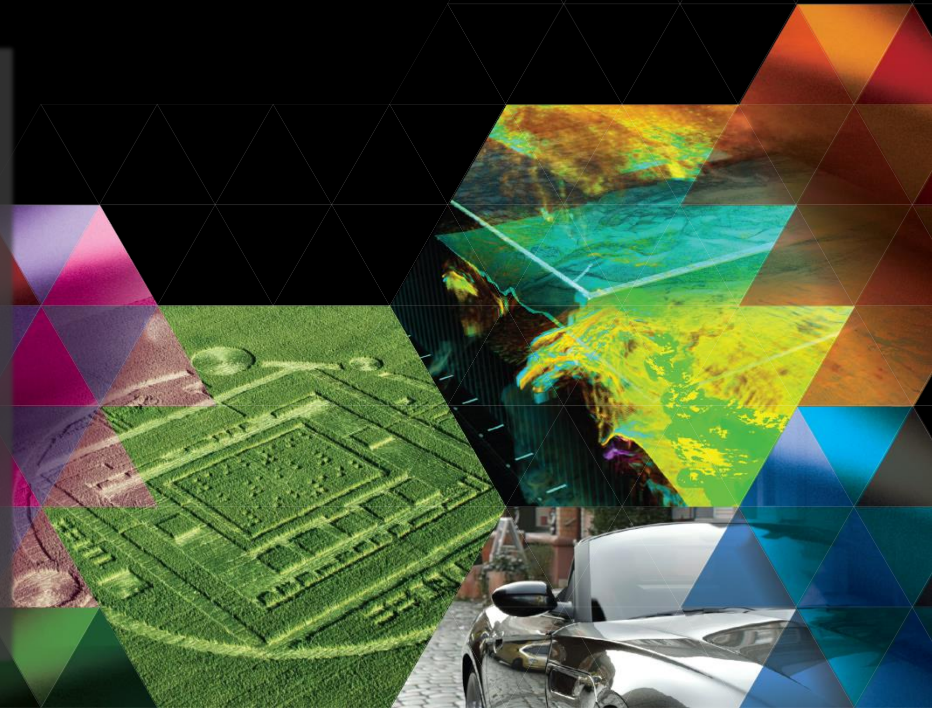
- Slides: http://on-demand.gputechconf.com/gtc/2014/video/S4692-vasp-accelerating-plane-wave-dft-codes.mp4
- Slides & Audio: http://on-demand.gputechconf.com/gtc/2014/presentations/S4692-vasp-accelerating-plane-wave-dft-codes.pdf

GPU TECHNOLOGY CONFERENCE

VASP: A CASE STUDY FOR ACCELERATING PLANE WAVE DFT CODES

Presenters: Sarah Tariq and Przemyslaw Tredak

Authors: Jeroen Bedorf, Przemyslaw Tredak , Dusan Stosic, Arash Ashari, Paul Springer, Darko Stosic, Sarah Tariq, Paul Fleurat-Lessard and Anciaux Sedrakian (Ens-lyon, IFPEN), Maxwell Hutchinson (University of Chicago) and Michael Widom (CMU)



GPU VASP Collaboration

- **Collaborators**

U of Chicago

- **2013-2014 Project Scope**

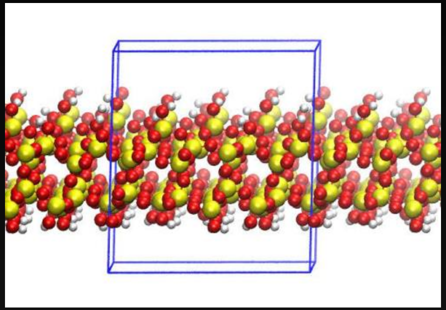
  Minimization algorithms to calculate electronic ground state

  - Blocked Davidson (ALGO = NORMAL & FAST)
  - RMM-DIIS (ALGO = VERYFAST & FAST)
  - K-Points
  - Optimization for critical step in exact exchange calculations

- **Earlier work**
  - *Speeding up plane-wave electronic-structure calculations using graphics-processing units*, Maintz, Eck, Dronskowski
  - *VASP on a GPU: application to exact-exchange calculations of the stability of elemental boron*, Hutchinson, Widom
  - *Accelerating VASP Electronic Structure Calculations Using Graphic Processing Units*, Hacene, Anciaux-Sedrakian, Rozanska, Klahr, Guignon, Fleurat-Lessard

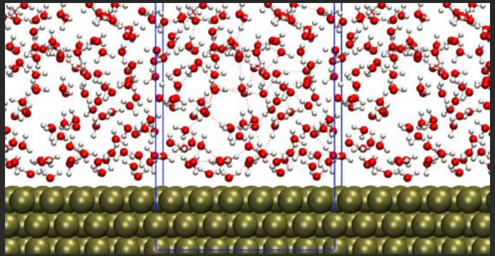
Target Workloads

- **Silica** *("medium")*
  - 7 Å thick slab of amorphous silica, 240 atoms ($Si_{68}O_{148}H_{24}$)
  - RMM-DIIS (ALGO = VERYFAST)



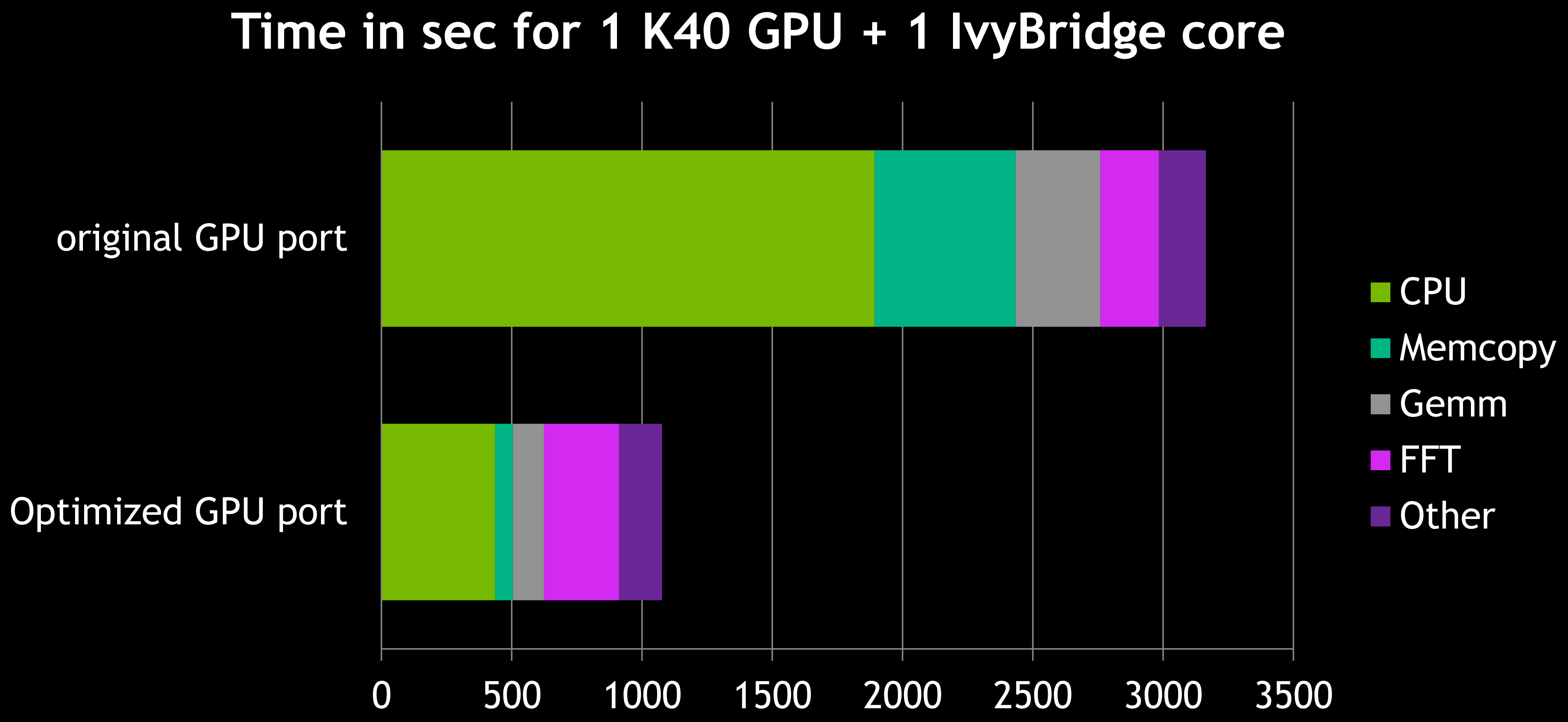
- **Nial-MD** *("large")*
  - Liquid metal molecular dynamics sample of Nickel-based superalloy
  - 500 atoms, 9 chemical species total
  - Blocked Davidson (ALGO = NORMAL)

- **INTERFACE** *("large")*
  - Interface of platinum metal with water
  - 108 Pt atoms, and 120 water molecules (468 atoms)
  - Blocked Davidson & RMM-DIIS (ALGO = FAST)



GPU TECHNOLOGY CONFERENCE

RUNTIME DISTRIBUTION FOR SILICA



OUTLINE

- Reduce communication
- Port more work to the GPU
- Optimize for small benchmarks
- Batch work
- Improve MPI scaling

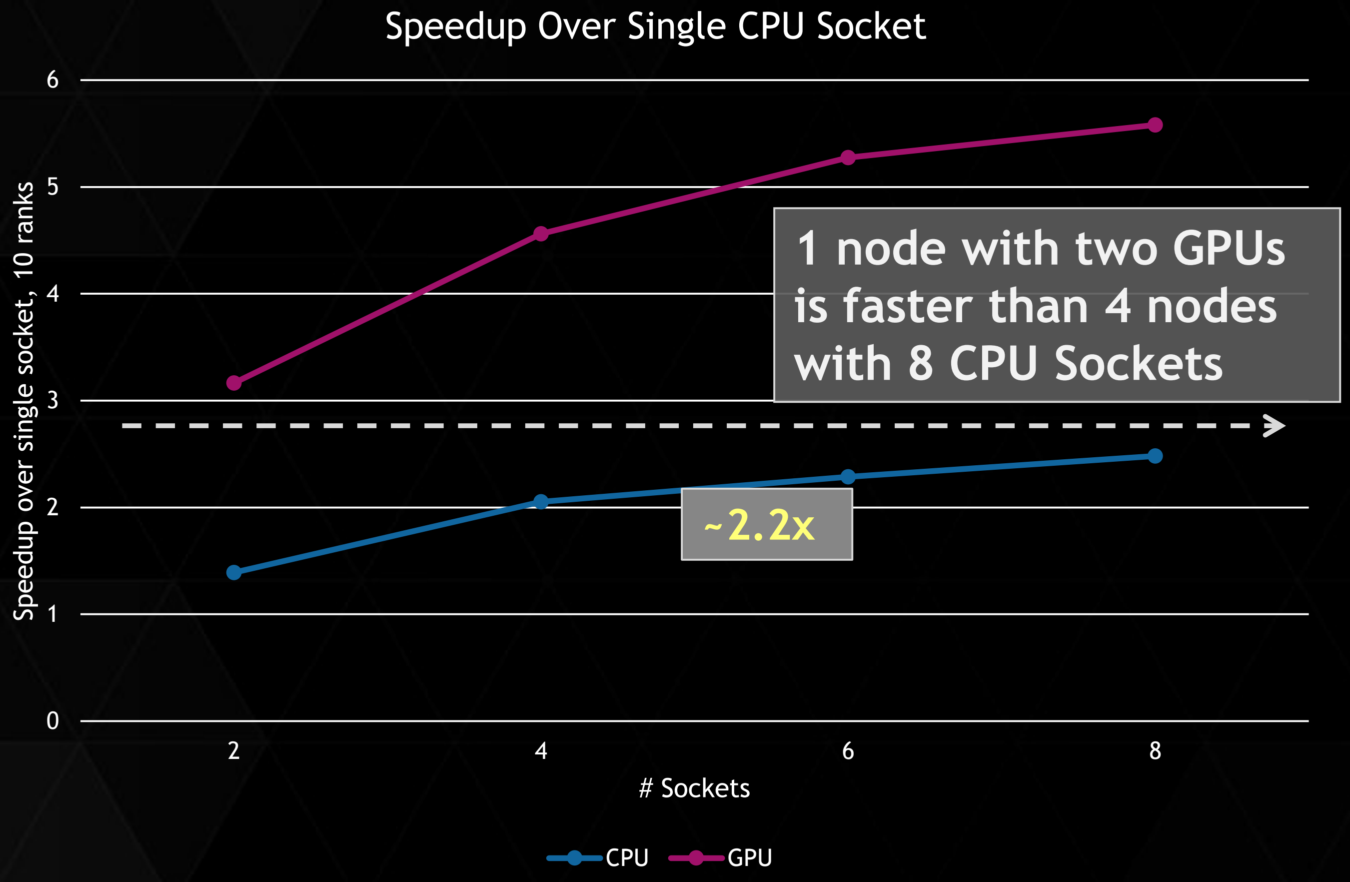
BENCHMARKING SYSTEM

- System
  - CPU: Xeon E5-2690 V2 (Ivy bridge), 2 sockets, 10 cores per socket, 3 GHz
  - GPUs: Tesla K80
- Local Storage
  - 1x 1 TB SATA 7200 rpm disks per server
- Shared storage
  - 16x 1 TB SATA 7200 rpm disks
  - Storage exported via NFS over Ethernet
  - Supermicro storage server
- Network
  - FDR InfiniBand
  - 10 GbE connection to storage server, 1 GbE everywhere else
  - Peak BW for single client ~100 MB/s under typical cluster conditions:

- Compilers
  - IPP
  - nvcc
- Libraries
  - MKL
  - CUBLAS
  - CUFFT

NVIDIA CONFIDENTIAL

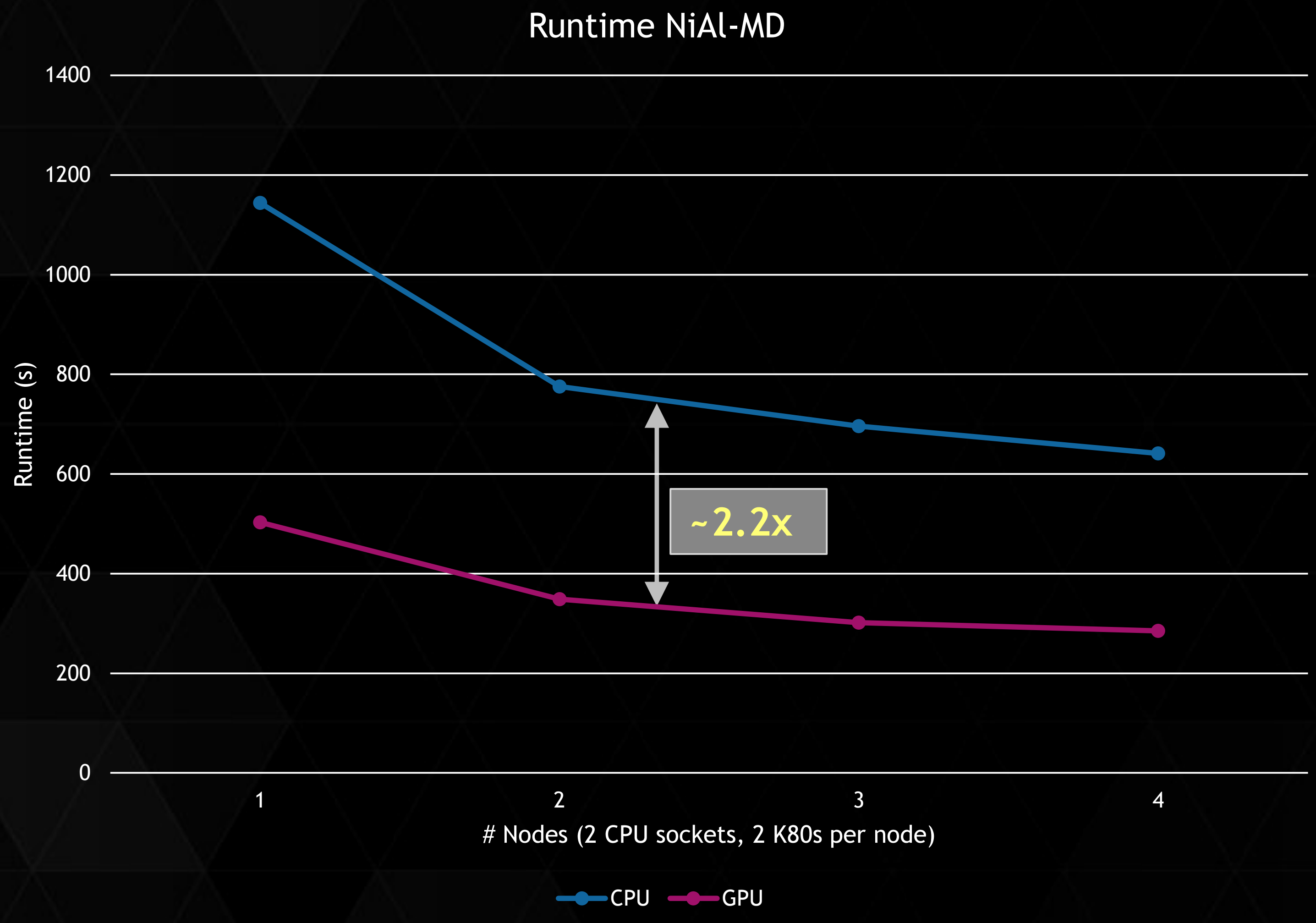
Preliminary Results NiAl-MD on Ivy Bridge vs. K80 (blocked Davidson)



- Measured on K80 and dual-socket Ivy Bridge (20 cores per socket @2.9GHz, two K80s per node)
- K80 with
  - Autoboost on,
  - ECC on
  - MPS on, 2 ranks per cuda device
- FDR Infiniband

NVIDIA CONFIDENTIAL

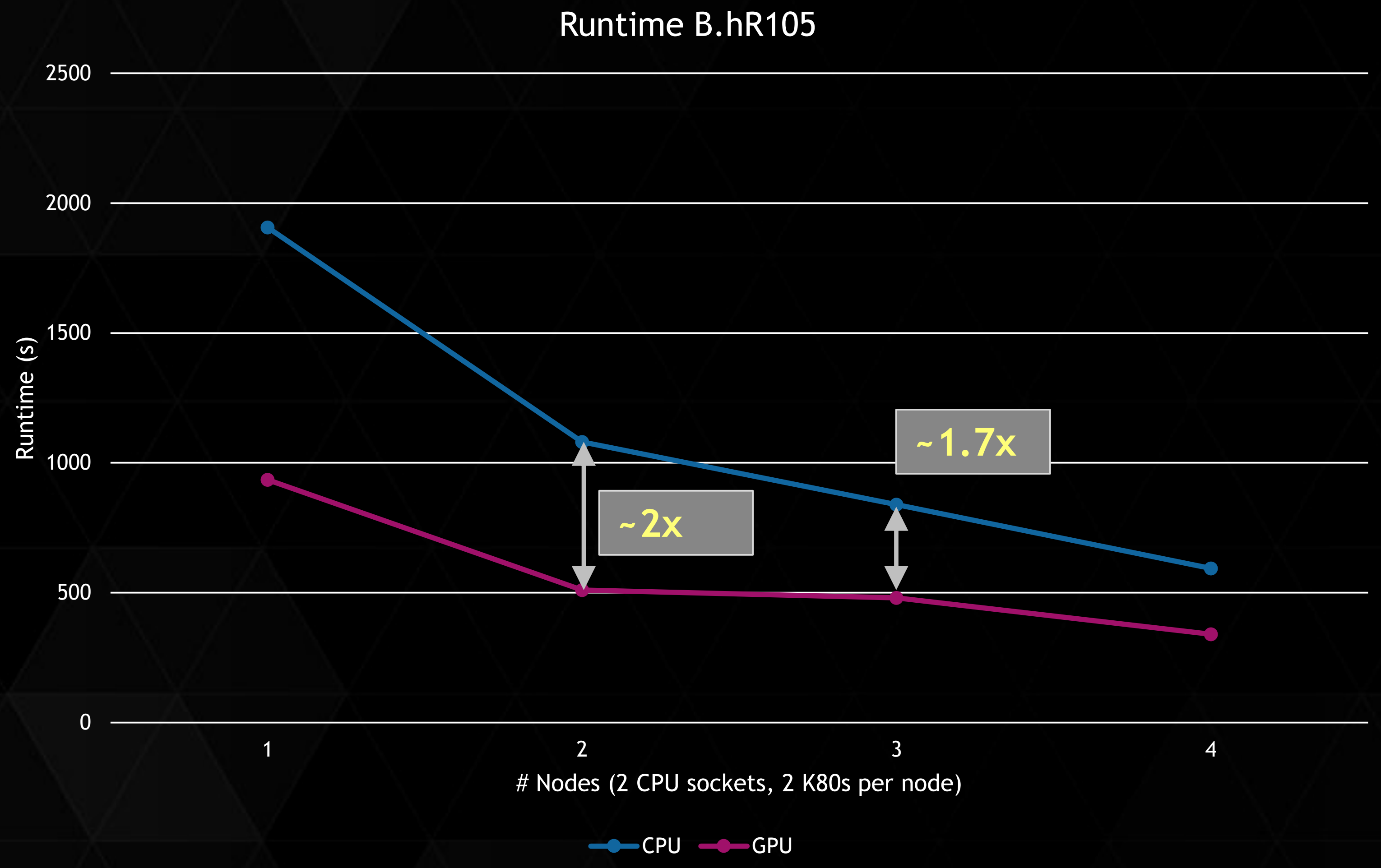
Preliminary Results NiAl-MD on Ivy Bridge vs. K80 (blocked Davidson)



- Measured on K80 and dual-socket Ivy Bridge (20 cores per socket @2.9GHz, two K80s per node)
- K80 with
  - Autoboost on,
  - ECC on
  - MPS on, 2 ranks per cuda device
- FDR Infiniband

NVIDIA CONFIDENTIAL

Preliminary Results B.hR105 on Ivy Bridge vs. K80 (exact exchange)



- Measured on K80 and dual-socket Ivy Bridge (20 cores per socket @2.9GHz, two K80s per node)
- K80 with
  - Autoboost on,
  - ECC on
  - no MPS
  - NSIM=32 for 1 and 2 nodes, NSIM=30 for 3 and 4 nodes
- FDR Infiniband

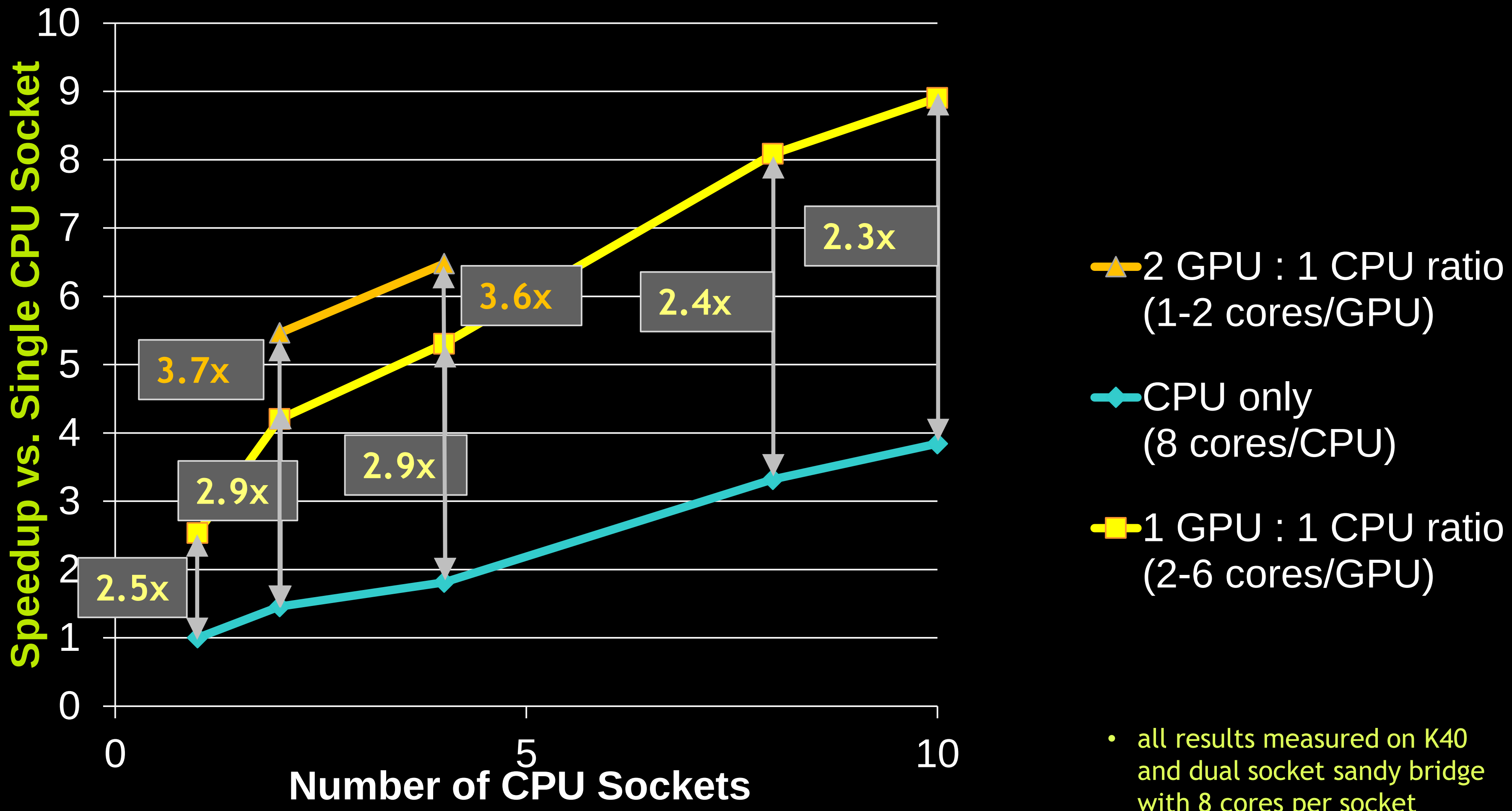
NVIDIA CONFIDENTIAL

Benchmarking System

- **System**
  - CPU: Xeon E5-2690, 8 cores, 2.9 GHz
  - GPUs: Tesla K40
- **Local Storage**
  - 1x 1 TB SATA 7200 rpm disks per server
- **Shared storage**
  - 16x 1 TB SATA 7200 rpm disks
  - Storage exported via NFS over Ethernet
  - Supermicro storage server
- **Network**
  - FDR InfiniBand
  - 10 GbE connection to storage server, 1 GbE everywhere else
  - Peak BW for single client ~100 MB/s under typical cluster conditions
- **Compilers**
  - IPP
  - nvcc
- **Libraries**
  - MKL
  - CUBLAS
  - CUFFT

RESULTS SILICA (RMM-DIIS) – VASP 5.2.2

NOTE: The results presented here are preliminary and are subject to change.

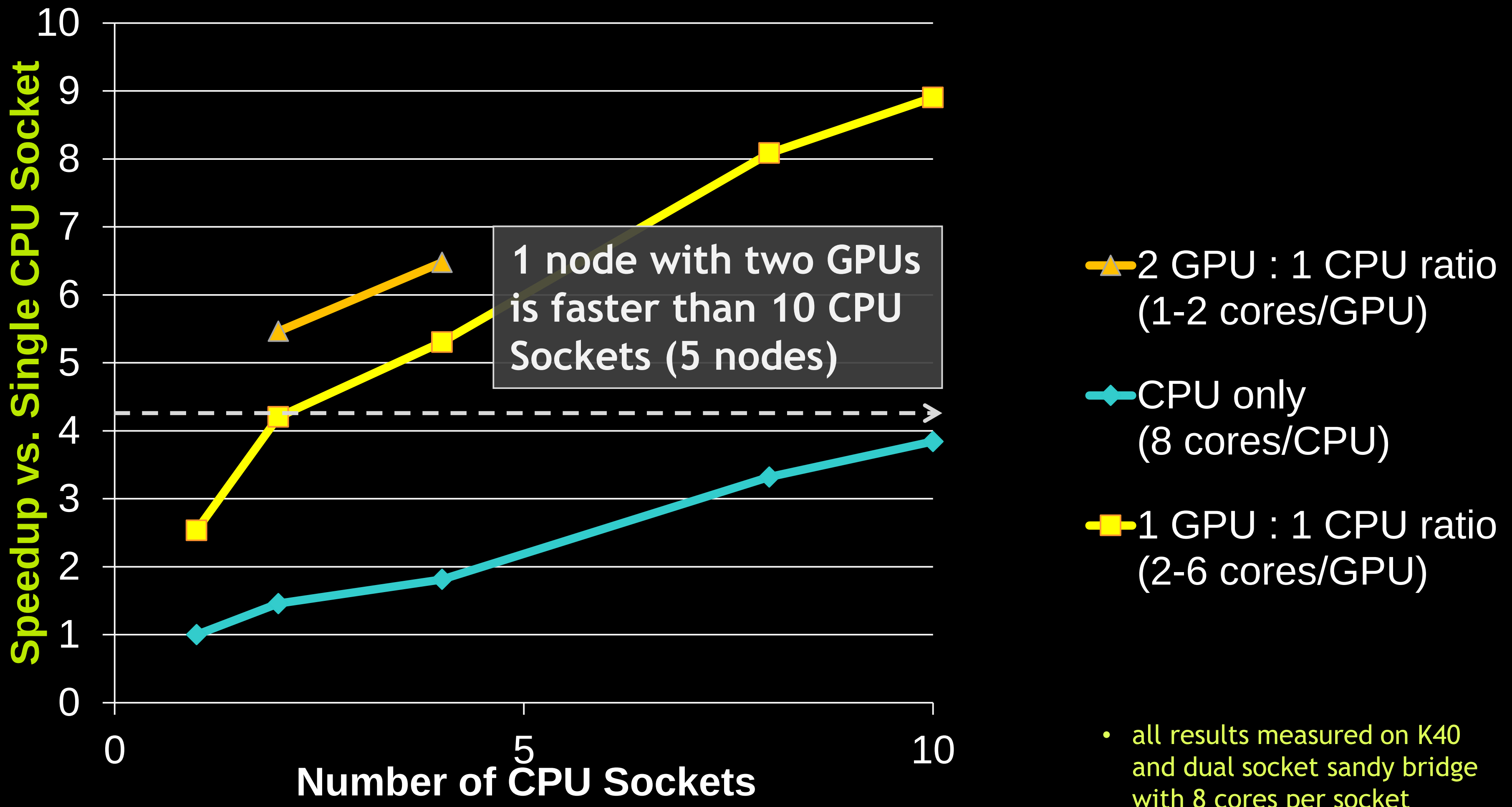


- all results measured on K40 and dual socket sandy bridge with 8 cores per socket running at 2.9GHz

GPU TECHNOLOGY CONFERENCE

RESULTS SILICA (RMM-DIIS) – VASP 5.2.2

NOTE: The results presented here are preliminary and are subject to change.



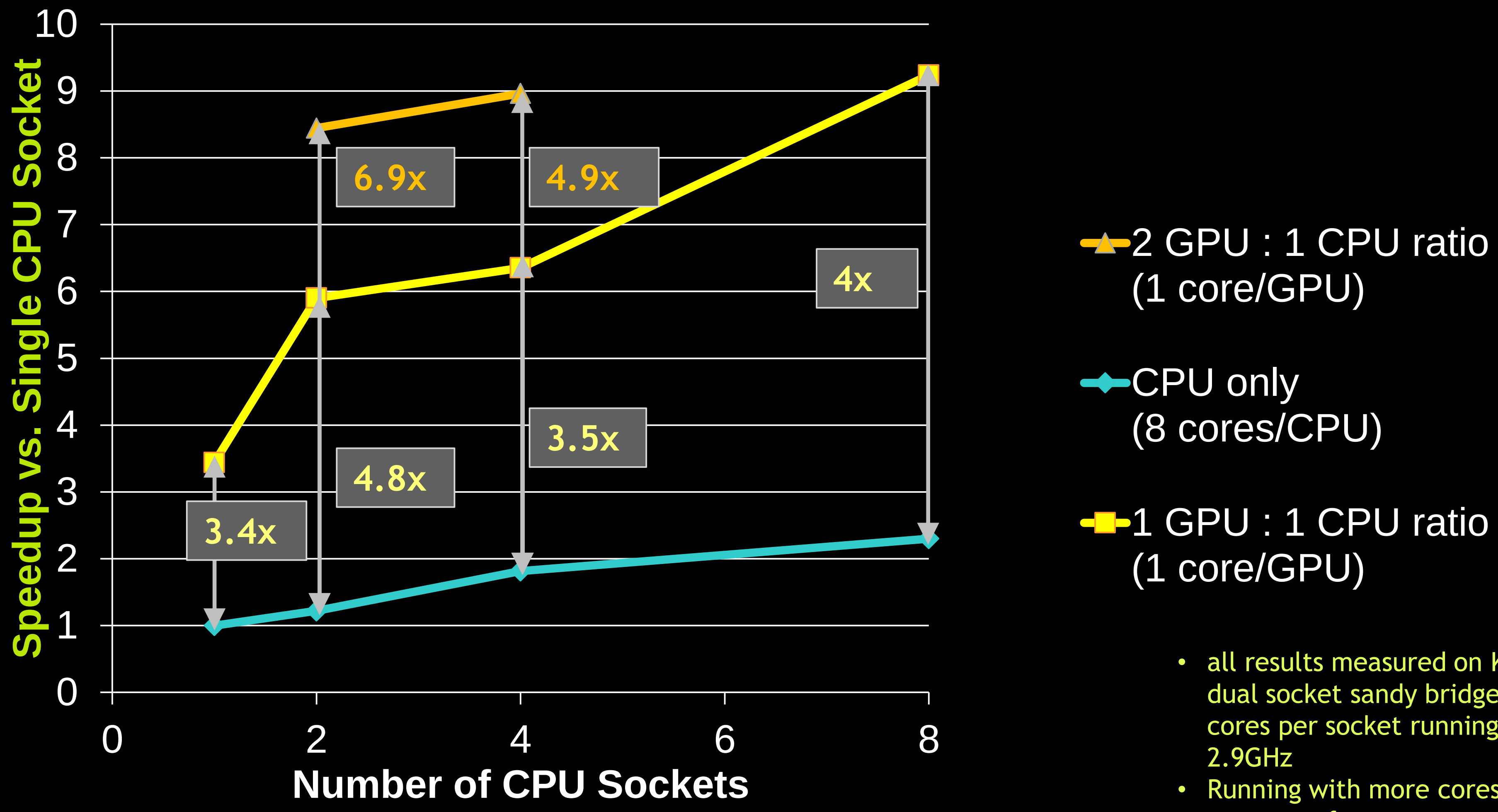
- all results measured on K40 and dual socket sandy bridge with 8 cores per socket running at 2.9GHz

GPU TECHNOLOGY CONFERENCE

GPU TECHNOLOGY CONFERENCE

RESULTS NIAL-MD (BLOCKED DAVIDSON) , VASP 5.2.2

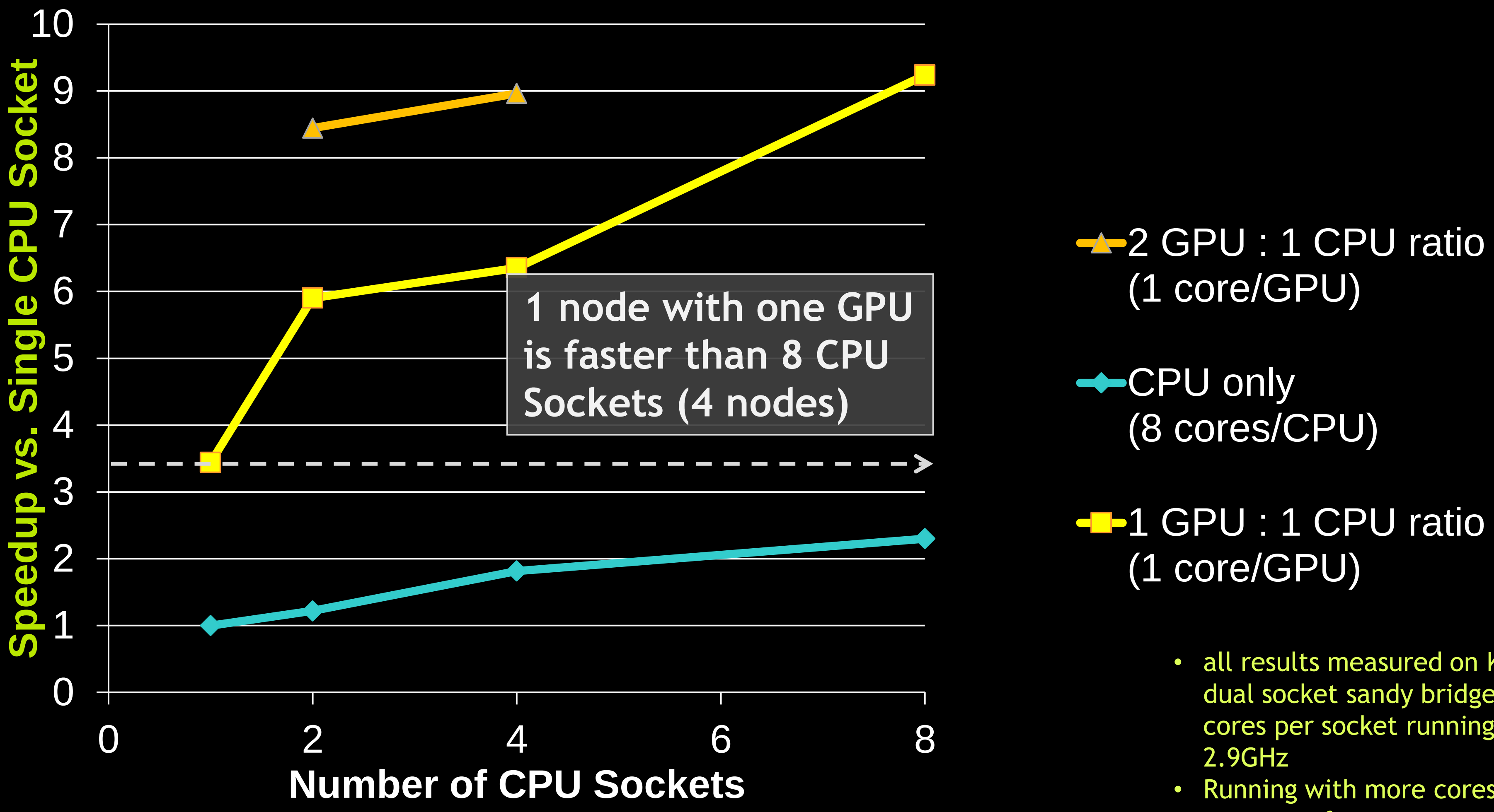
NOTE: The results presented here are preliminary and are subject to change.



- all results measured on K40 and dual socket sandy bridge with 8 cores per socket running at 2.9GHz
- Running with more cores per GPU runs out of memory

RESULTS NIAL-MD (BLOCKED DAVIDSON) , VASP 5.2.2

NOTE: The results presented here are preliminary and are subject to change.



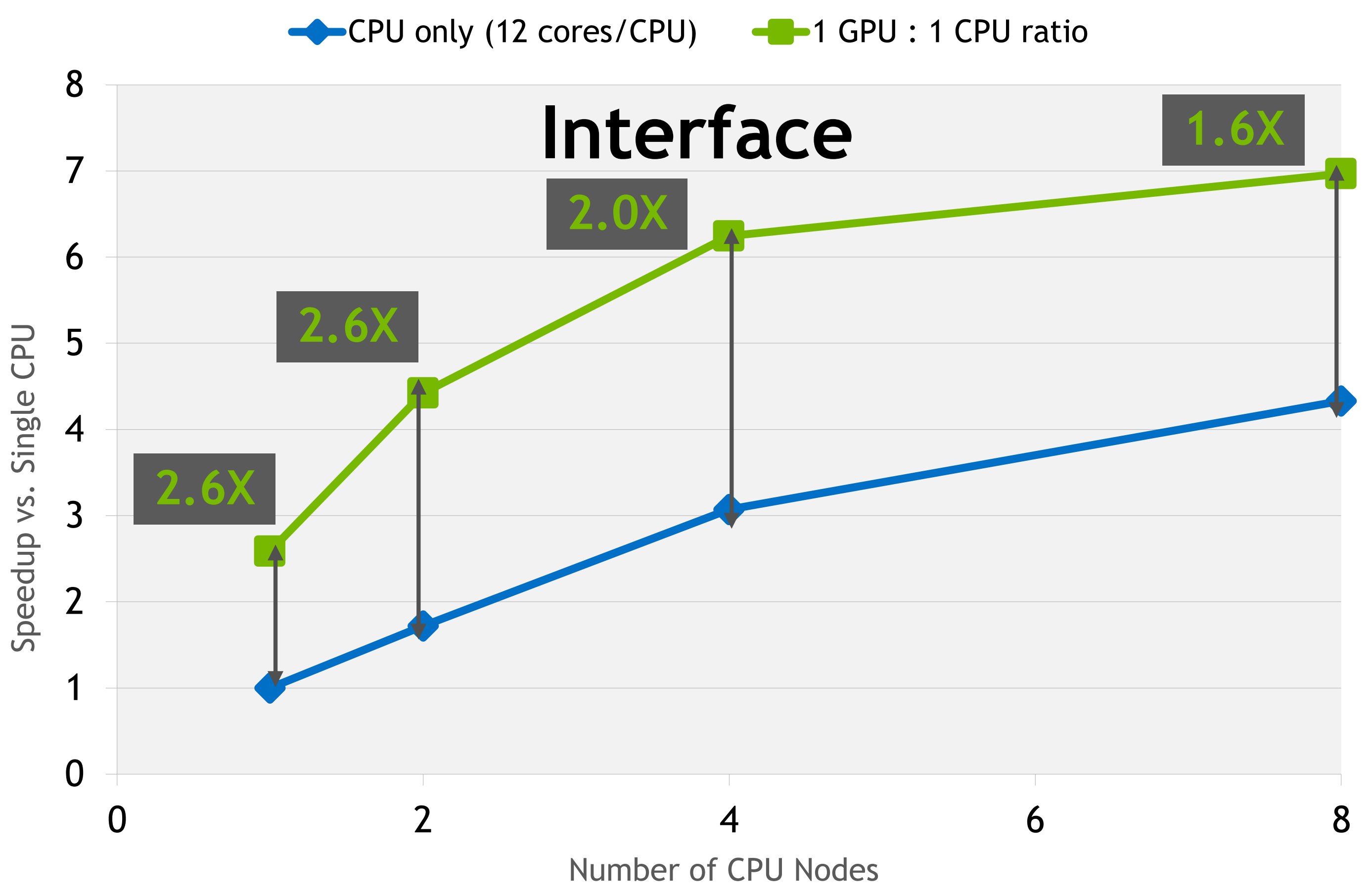
- all results measured on K40 and dual socket sandy bridge with 8 cores per socket running at 2.9GHz
- Running with more cores per GPU runs out of memory

GPU TECHNOLOGY CONFERENCE

VASP (Ivybridge & K40)

November 2015

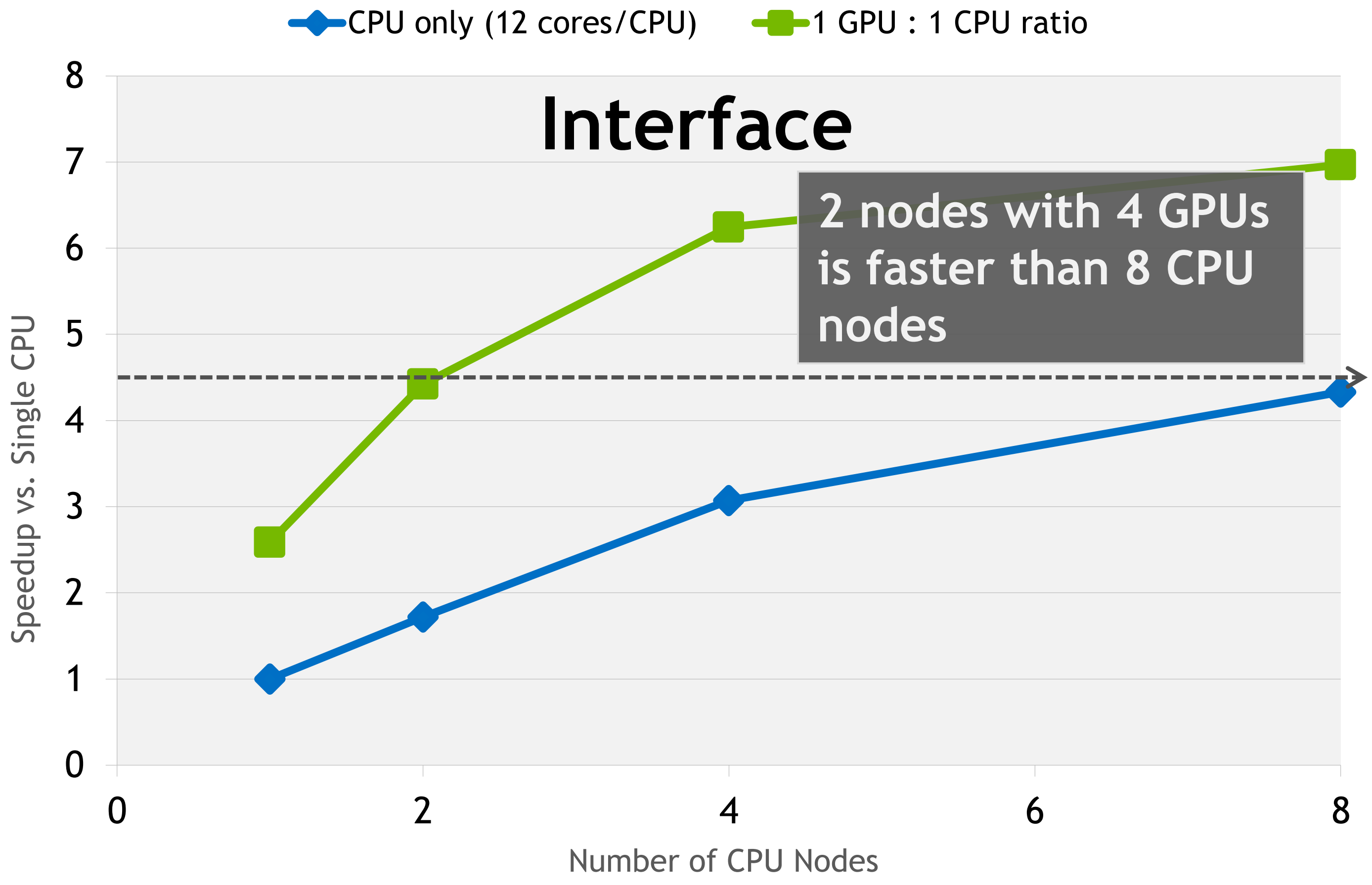
VASP Interface Benchmark



- Measured on Tesla K40 and dual-socket IvyBridge (12 cores per socket @2.7GHz)
- Default K40 clocks (GPU Autoboost off)
- FDR InfiniBand

➢ *Blocked Davidson + RMM-DIIS (ALGO=Fast)*

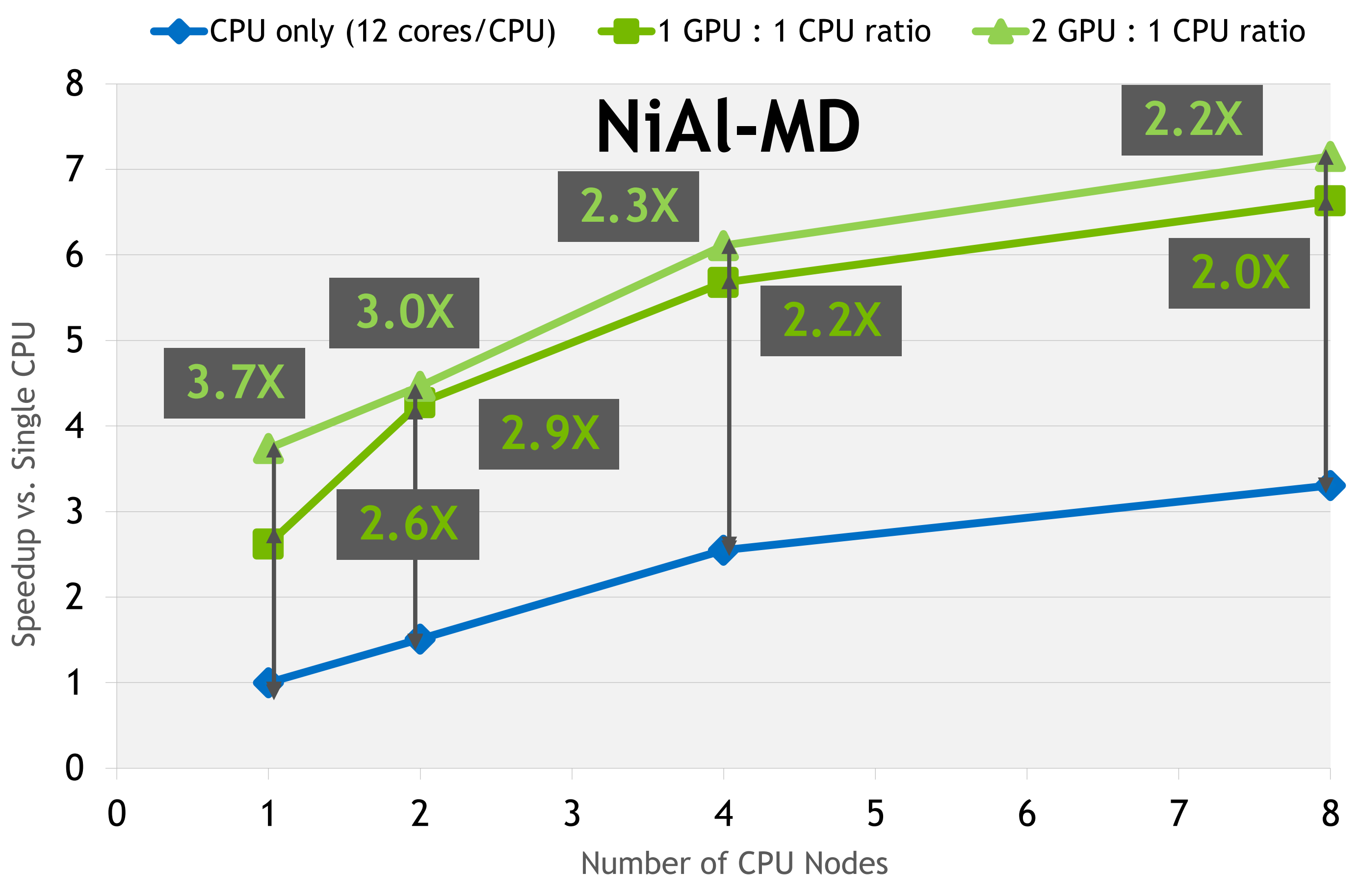
VASP Interface Benchmark



- Measured on Tesla K40 and dual-socket IvyBridge (12 cores per socket @2.7GHz)
- Default K40 clocks (GPU Autoboost off)
- FDR InfiniBand

➢ *Blocked Davidson + RMM-DIIS (ALGO=Fast)*

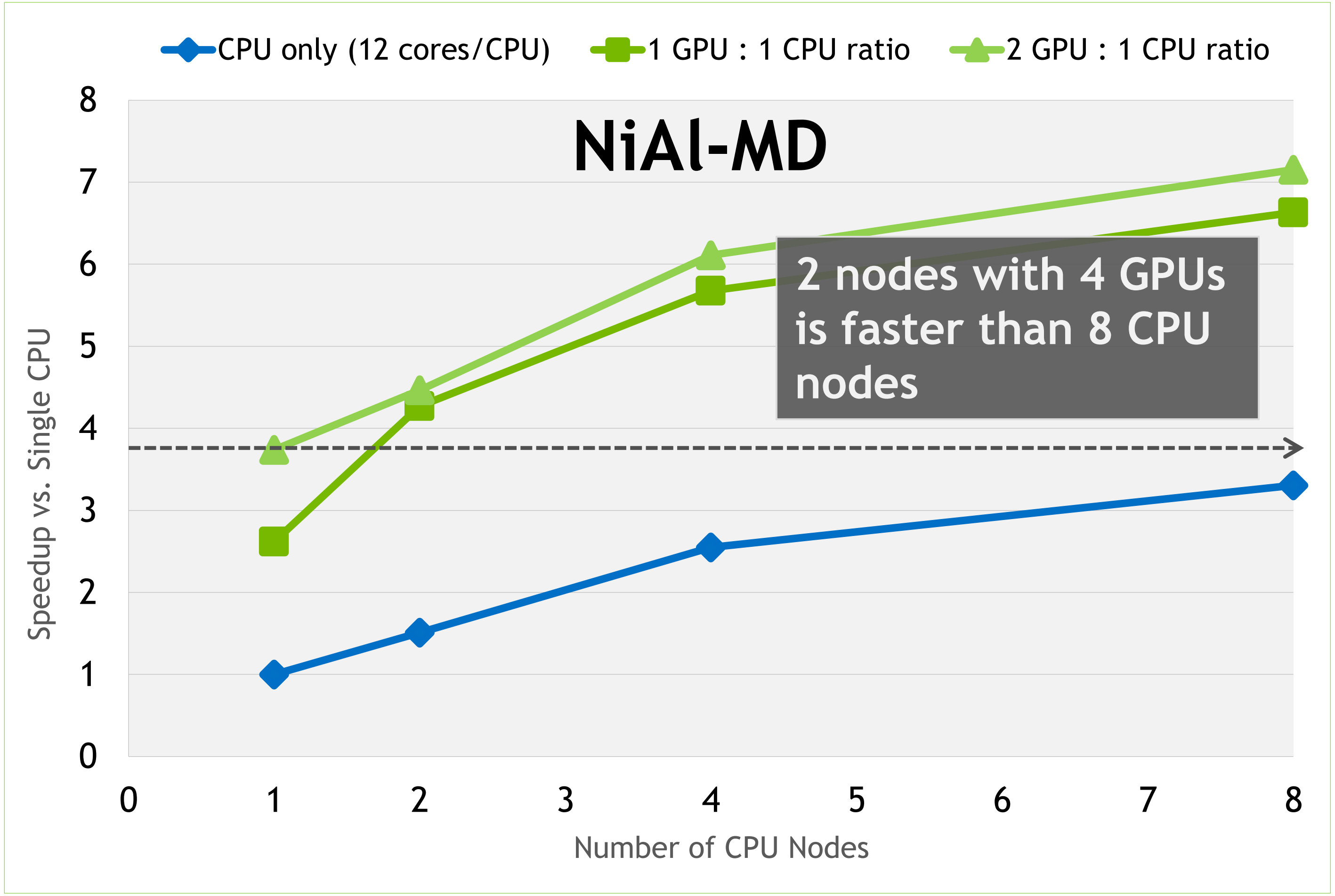
VASP NiAl-MD Benchmark



- Measured on Tesla K40 and dual-socket IvyBridge (12 cores per socket @2.7GHz)
- Default K40 clocks (GPU Autoboost off)
- FDR InfiniBand

➢ *Blocked Davidson + RMM-DIIS (ALGO=Fast)*

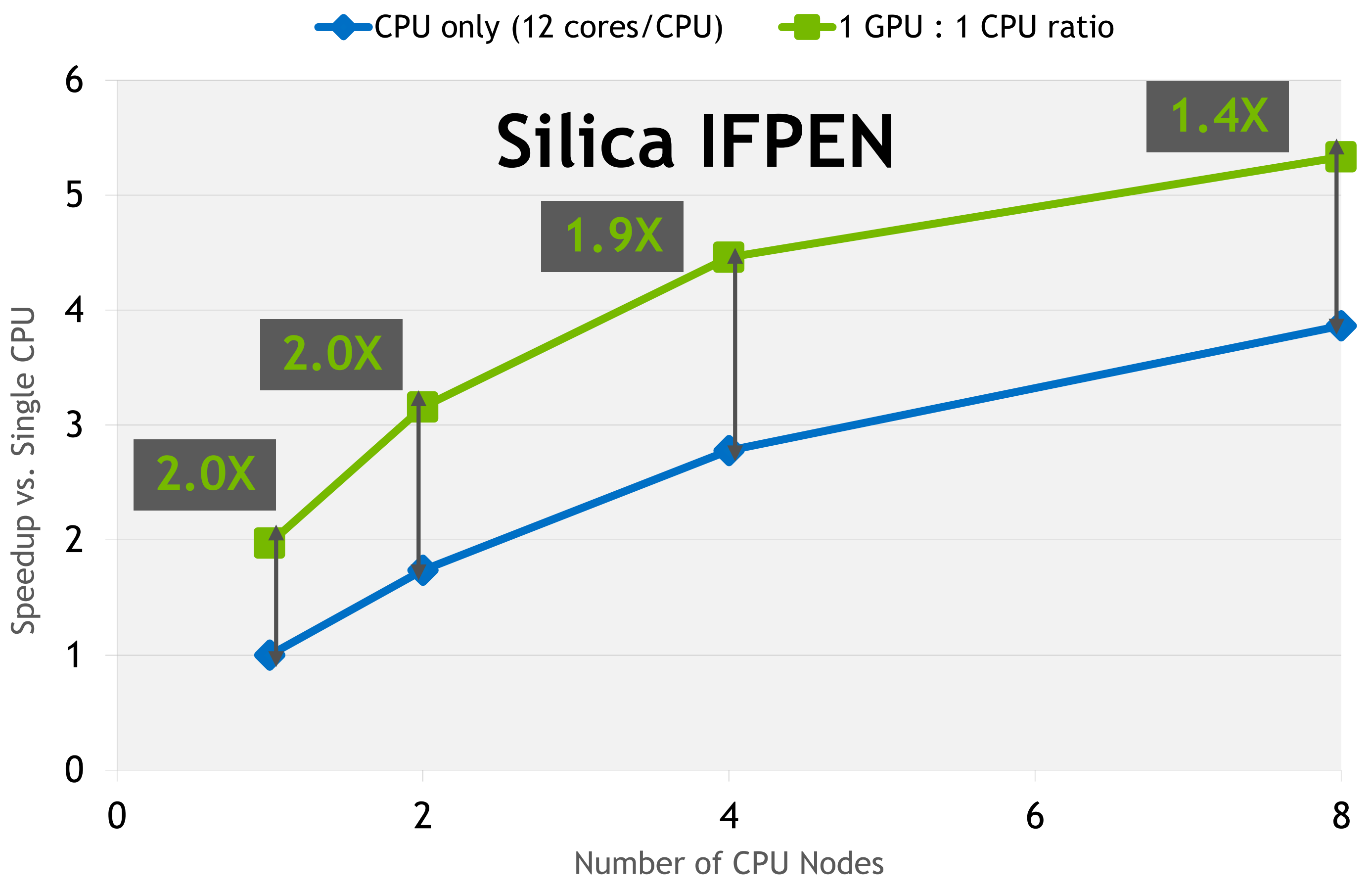
VASP NiAl-MD Benchmark



- Measured on Tesla K40 and dual-socket IvyBridge (12 cores per socket @2.7GHz)
- Default K40 clocks (GPU Autoboost off)
- FDR InfiniBand

➢ *Blocked Davidson + RMM-DIIS (ALGO=Fast)*

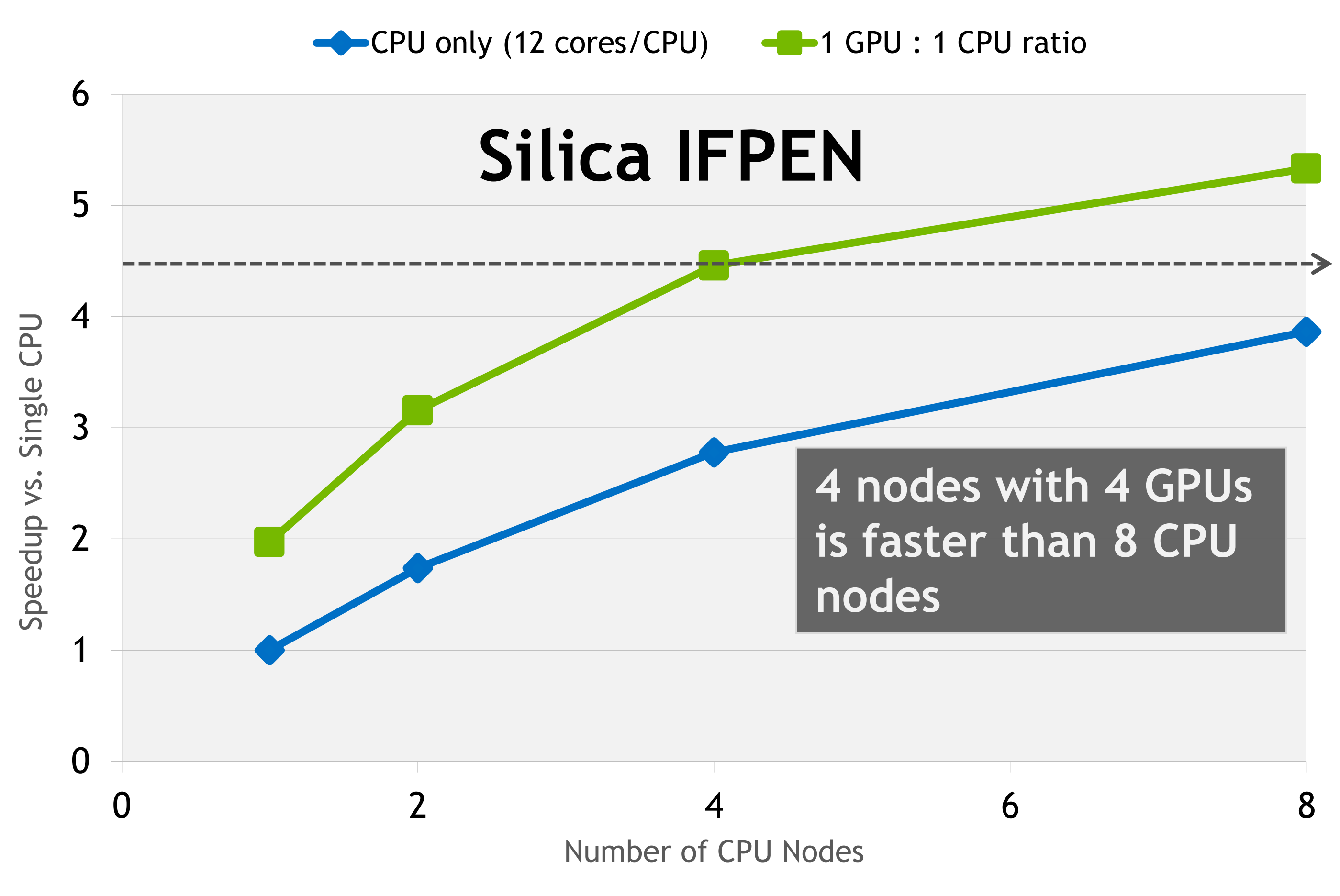
VASP Silica IFPEN Benchmark



- Measured on Tesla K40 and dual-socket IvyBridge (12 cores per socket @2.7GHz)
- Default K40 clocks (GPU Autoboost off)
- FDR InfiniBand

➢ *RMM-DIIS (ALGO=Veryfast)*

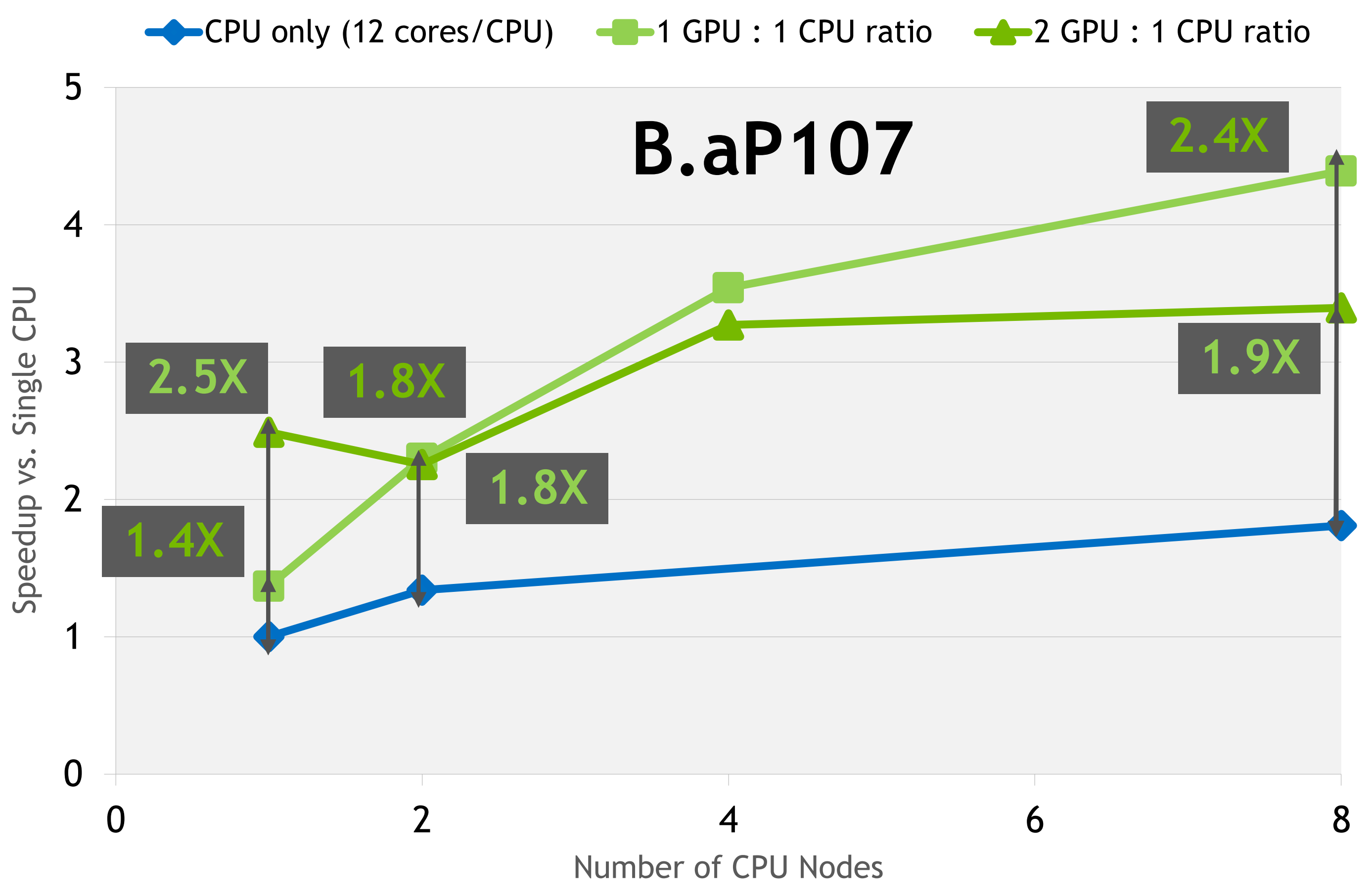
VASP Silica IFPEN Benchmark



- Measured on Tesla K40 and dual-socket IvyBridge (12 cores per socket @2.7GHz)
- Default K40 clocks (GPU Autoboost off)
- FDR InfiniBand

➢ *RMM-DIIS (ALGO=Veryfast)*

VASP B.aP107 Benchmark

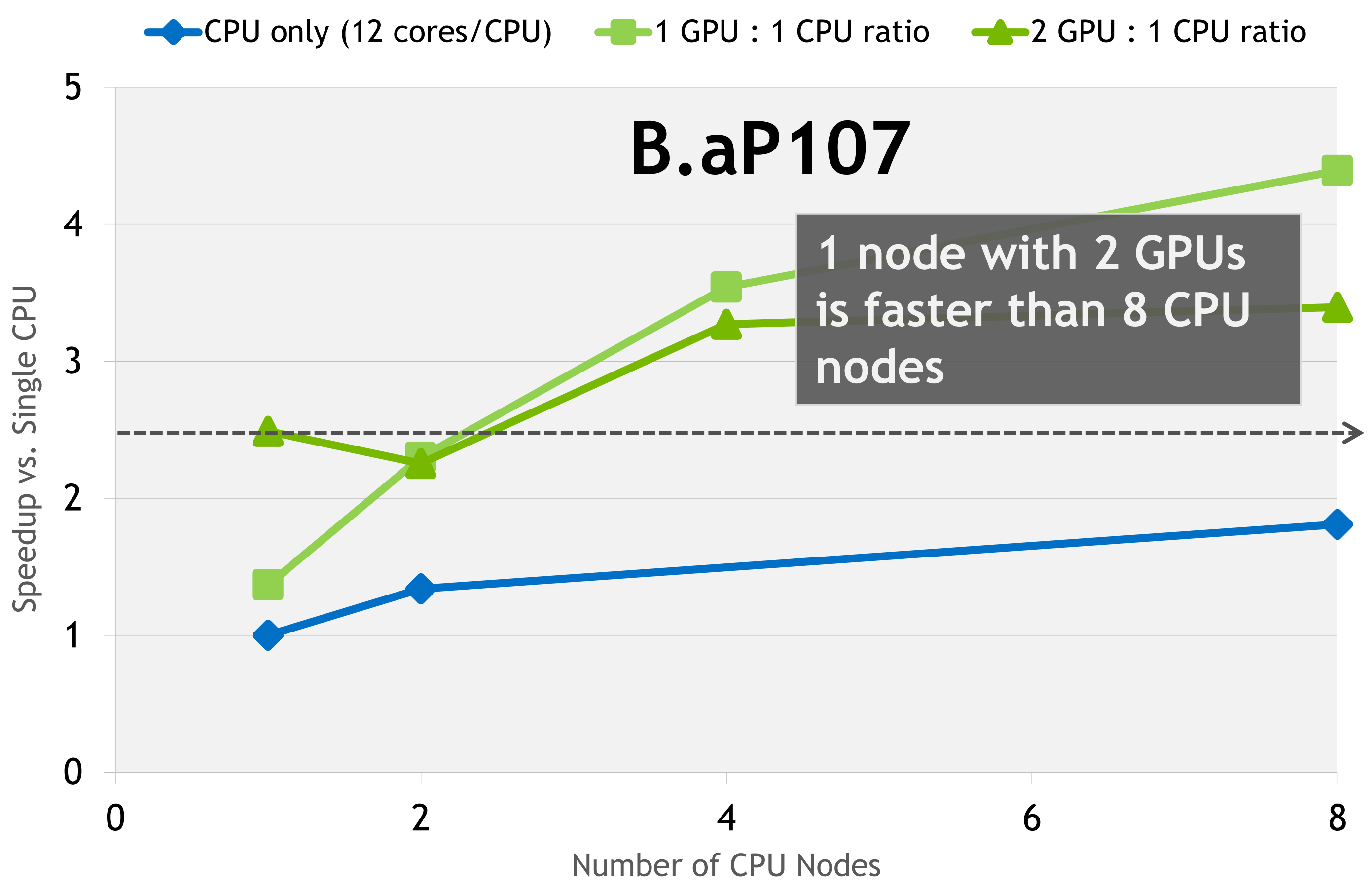


- Measured on Tesla K40 and dual-socket IvyBridge (12 cores per socket @2.7GHz)
- Default K40 clocks (GPU Autoboost off)
- FDR InfiniBand

➢ *Hybrid functional calculation (exact exchange) with blocked Davidson. No KPoint parallelization.*

➢ *Hybrid Functional with blocked Davicson (ALGO=Normal)*

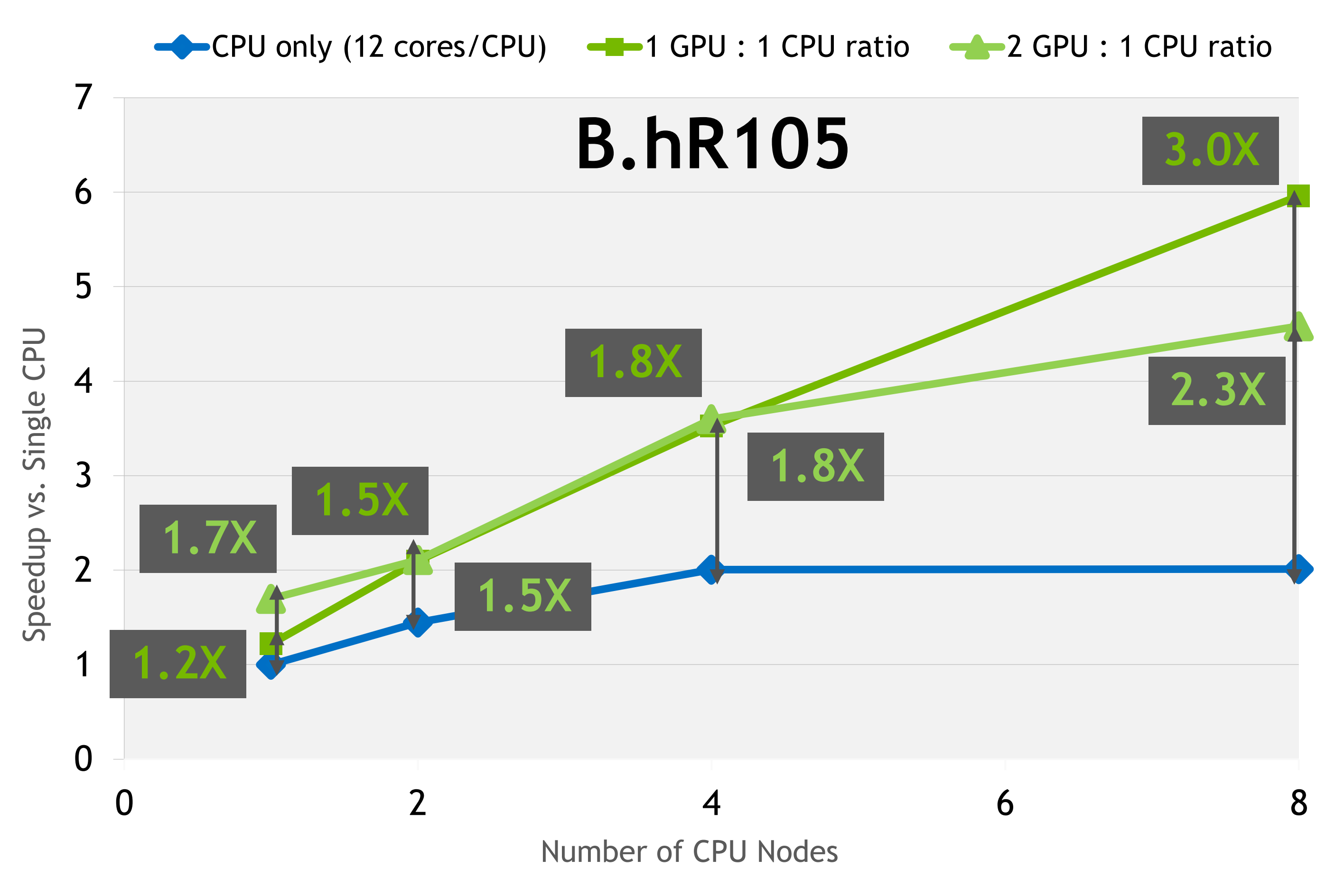
VASP B.aP107 Benchmark



- Measured on Tesla K40 and dual-socket IvyBridge (12 cores per socket @2.7GHz)
- Default K40 clocks (GPU Autoboost off)
- FDR InfiniBand

> *Hybrid functional calculation (exact exchange) with blocked Davidson. No KPoint parallelization.*
> *Hybrid Functional with blocked Davicson (ALGO=Normal)*

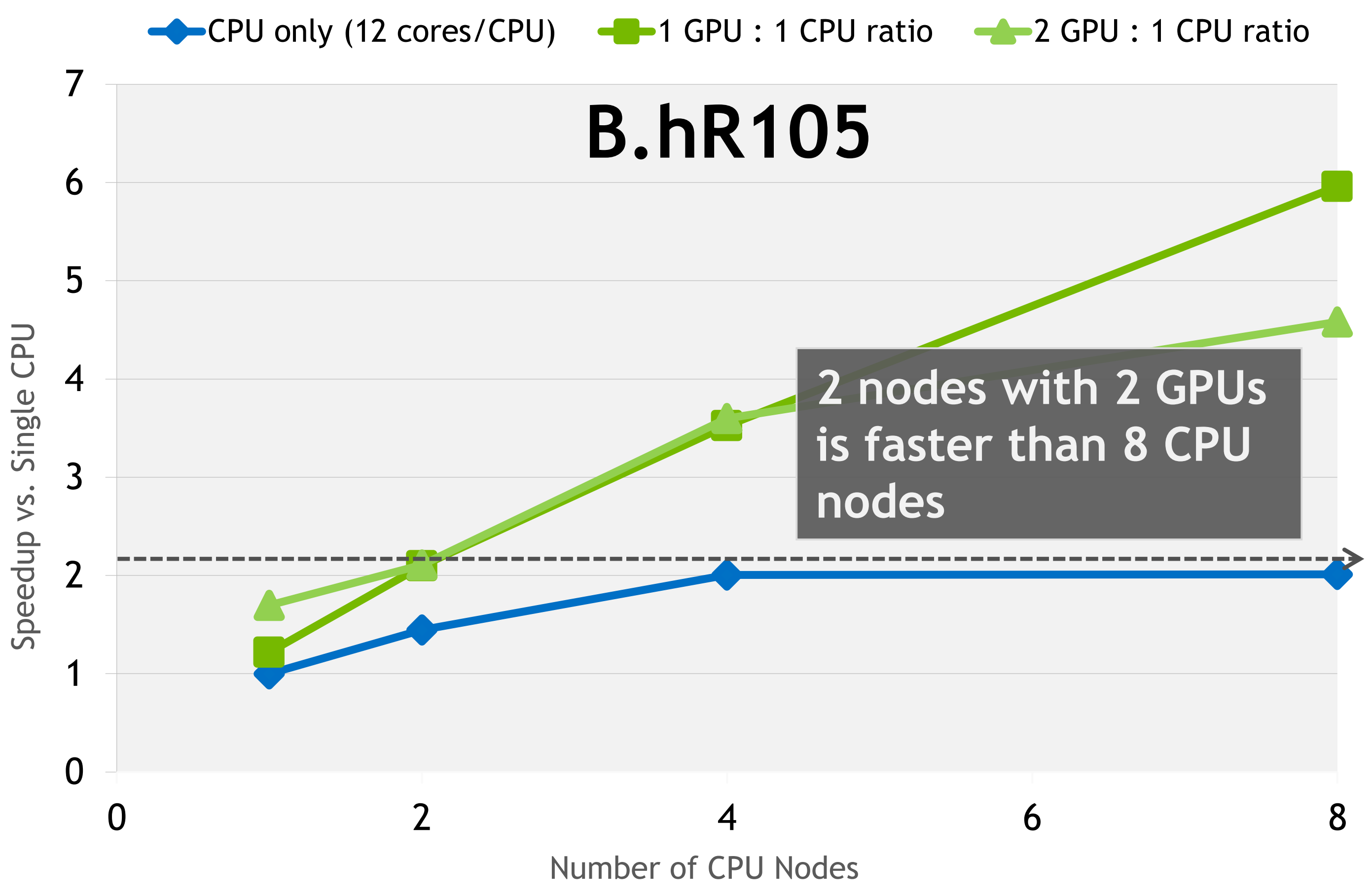
VASP B.hR105 Benchmark



- Measured on Tesla K40 and dual-socket IvyBridge (12 cores per socket @2.7GHz)
- Default K40 clocks (GPU Autoboost off)
- FDR InfiniBand

➢ *Hybrid Functional with blocked Davicson (ALGO=Normal)*

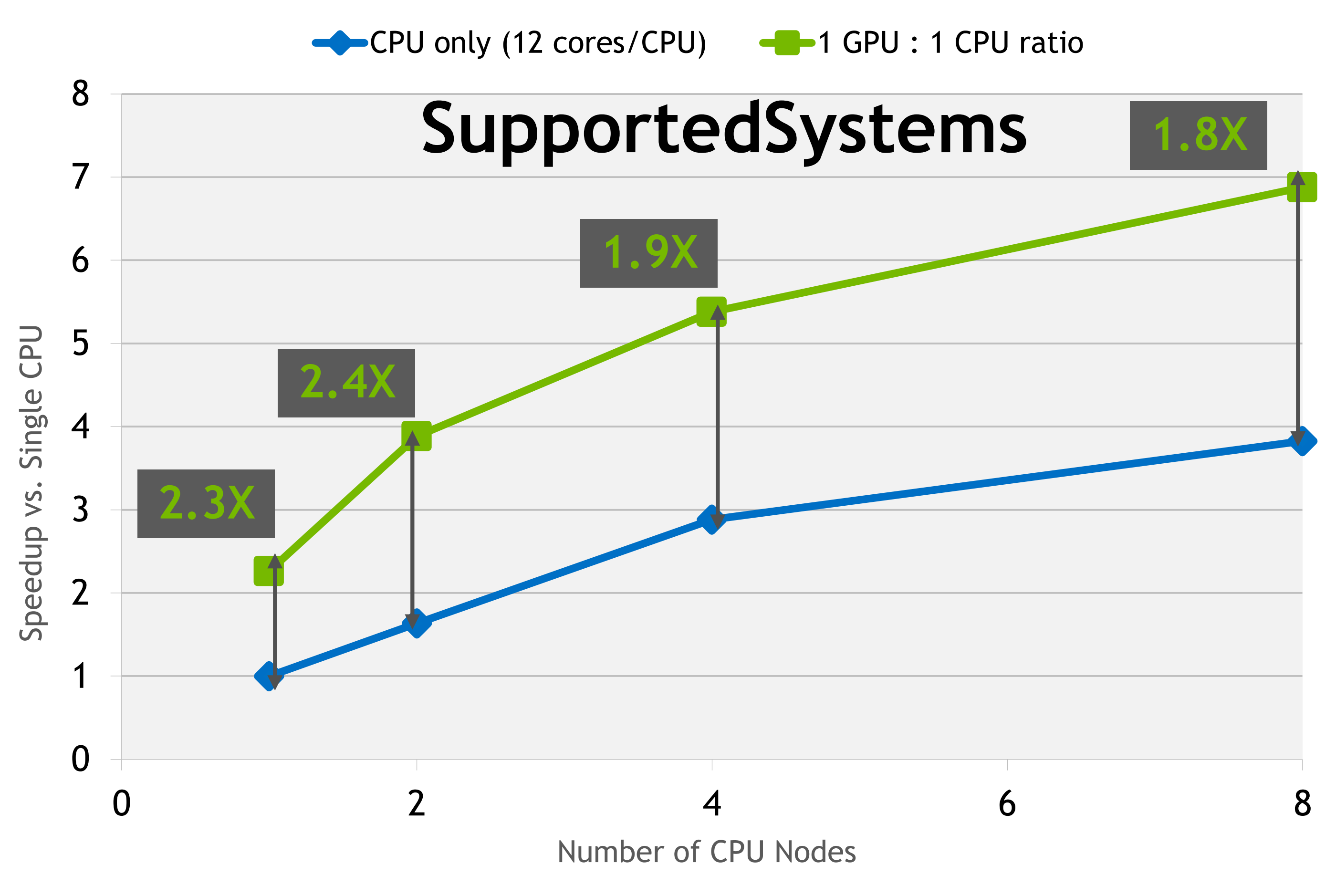
VASP B.hR105 Benchmark



- Measured on Tesla K40 and dual-socket IvyBridge (12 cores per socket @2.7GHz)
- Default K40 clocks (GPU Autoboost off)
- FDR InfiniBand

➢ *Hybrid Functional with blocked Davicson (ALGO=Normal)*

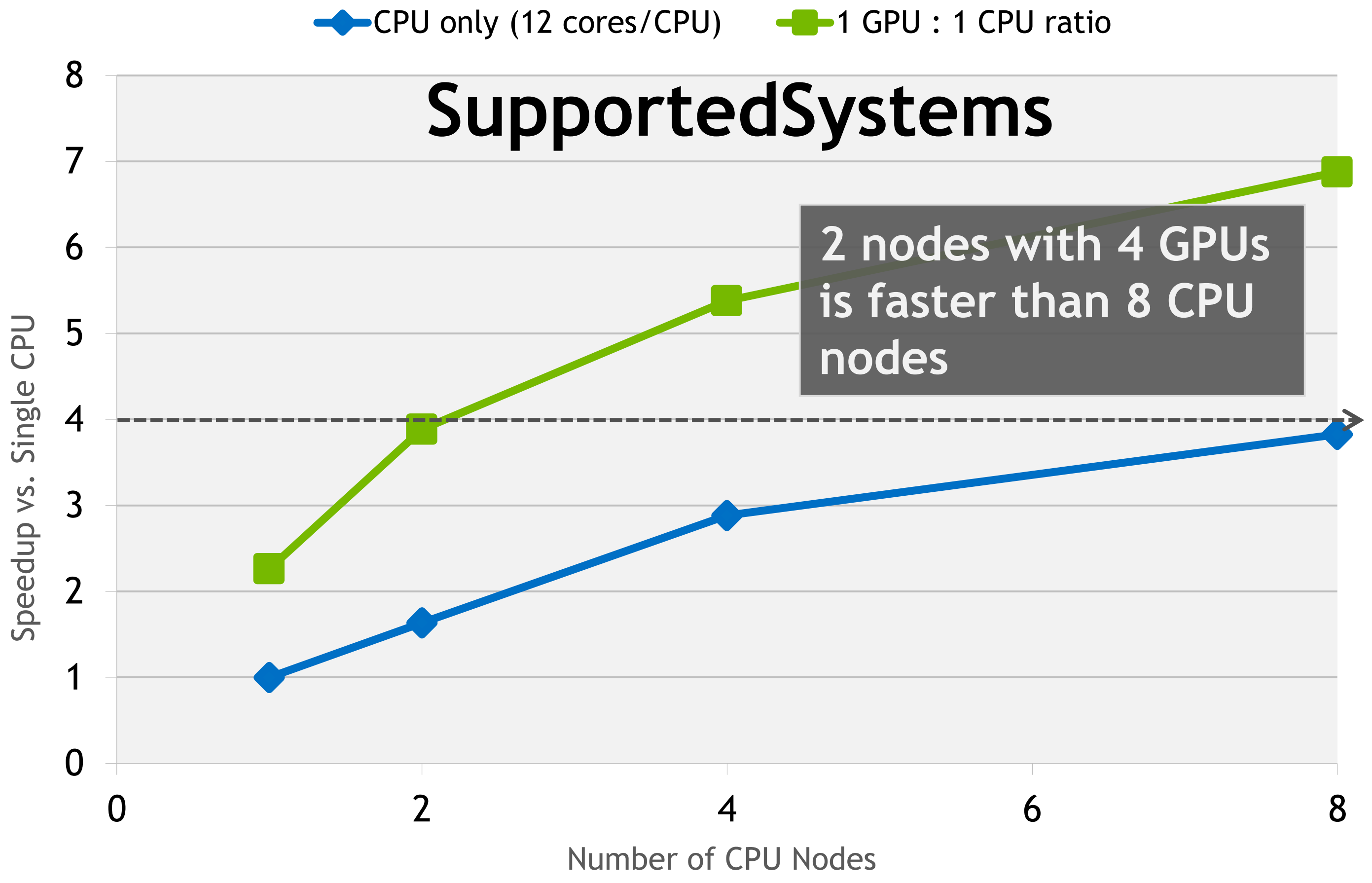
VASP SupportedSystems Benchmark



- Measured on Tesla K40 and dual-socket IvyBridge (12 cores per socket @2.7GHz)
- Default K40 clocks (GPU Autoboost off)
- FDR InfiniBand

➢ *Blocked Davidson + RMM-DIIS (ALGO=Fast)*

VASP SupportedSystems Benchmark



- Measured on Tesla K40 and dual-socket IvyBridge (12 cores per socket @2.7GHz)
- Default K40 clocks (GPU Autoboost off)
- FDR InfiniBand

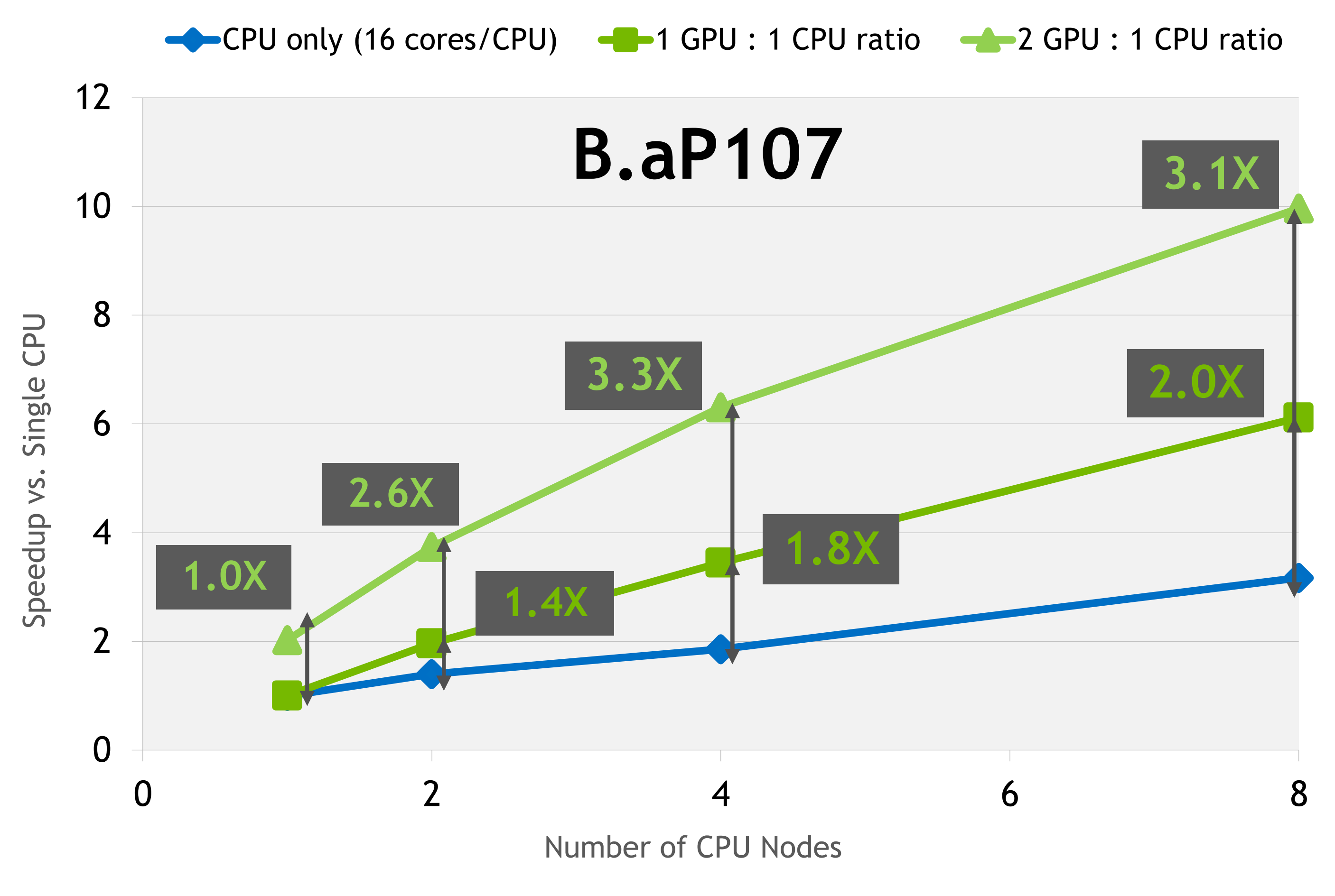
➢ *Blocked Davidson + RMM-DIIS (ALGO=Fast)*

VASP (Haswell & K80)

November 2015

NVIDIA

VASP B.aP107 Benchmark

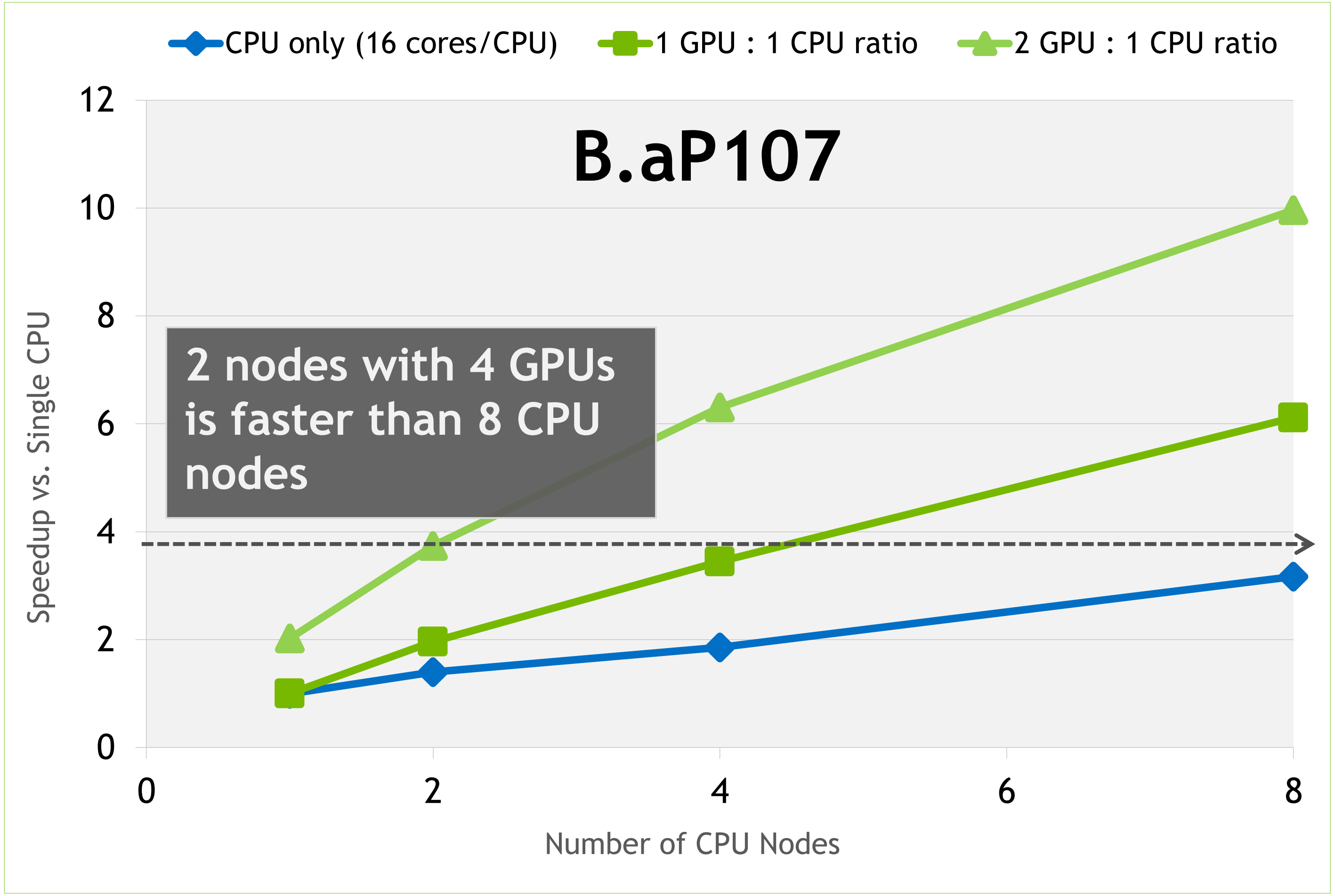


- Measured on Tesla K80 and dual-socket Haswell (16 cores per socket @2.3GHz)
- Default K80 clocks (GPU Autoboost on)
- FDR InfiniBand

➢ *Hybrid functional calculation (exact exchange) with blocked Davidson. No KPoint parallelization.*

➢ *Hybrid Functional with blocked Davicson (ALGO=Normal)*

VASP B.aP107 Benchmark

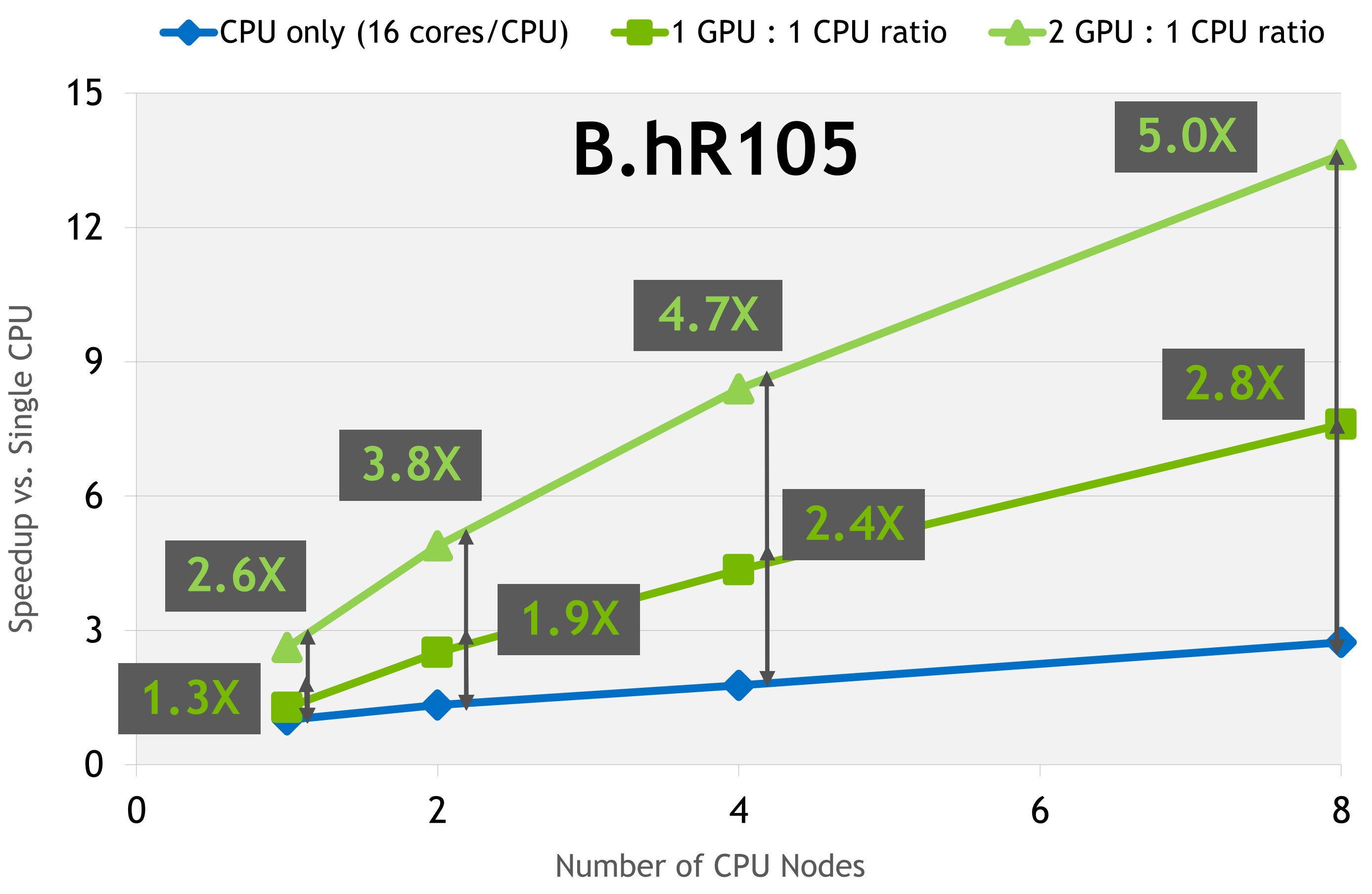


- Measured on Tesla K80 and dual-socket Haswell (16 cores per socket @2.3GHz)
- Default K80 clocks (GPU Autoboost on)
- FDR InfiniBand

➢ *Hybrid functional calculation (exact exchange) with blocked Davidson. No KPoint parallelization.*

➢ *Hybrid Functional with blocked Davicson (ALGO=Normal)*

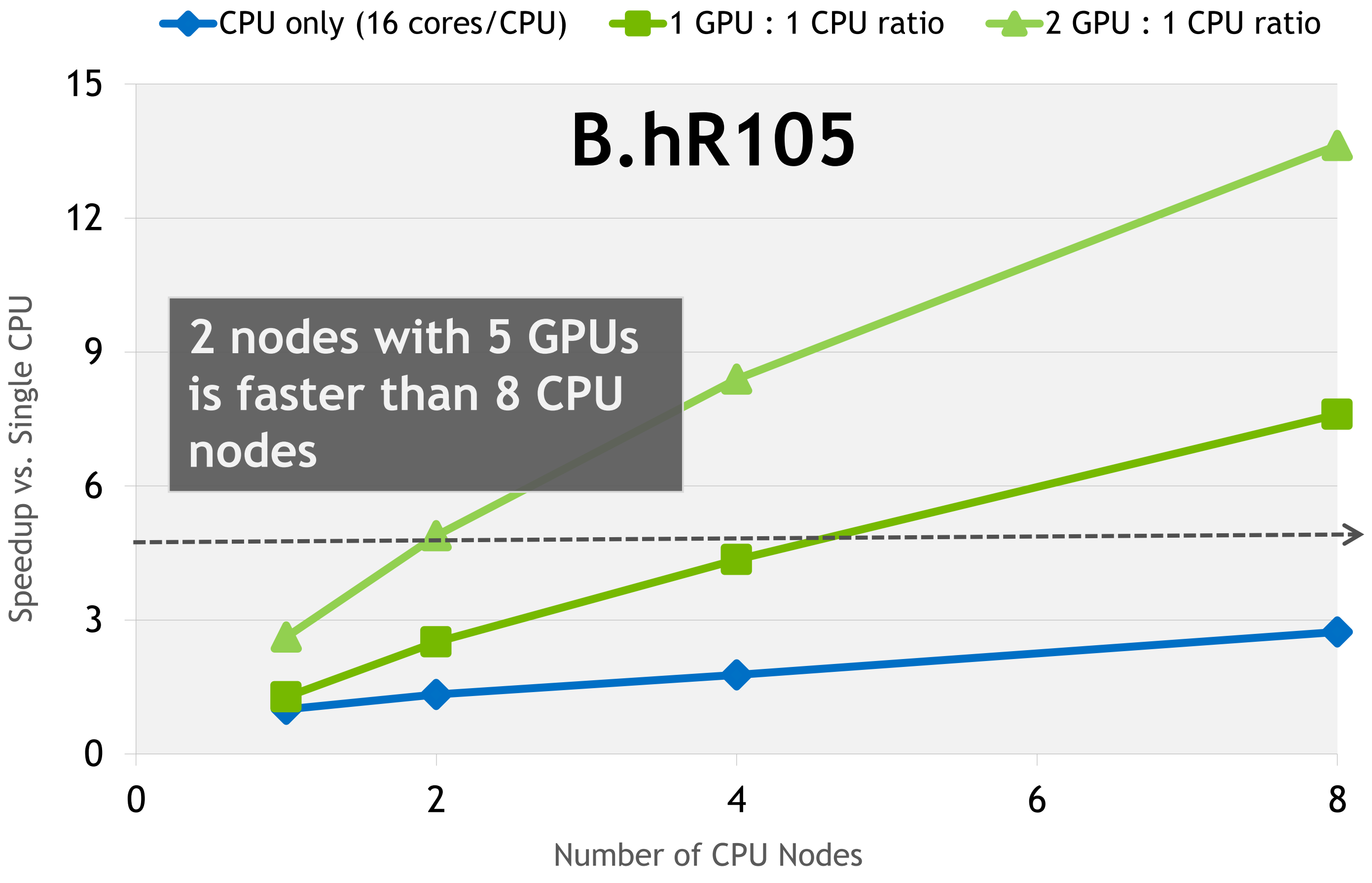
VASP B.hR105 Benchmark



- Measured on Tesla K80 and dual-socket Haswell (16 cores per socket @2.3GHz)
- Default K80 clocks (GPU Autoboost on)
- FDR InfiniBand

➢ *Hybrid Functional with blocked Davicson (ALGO=Normal)*

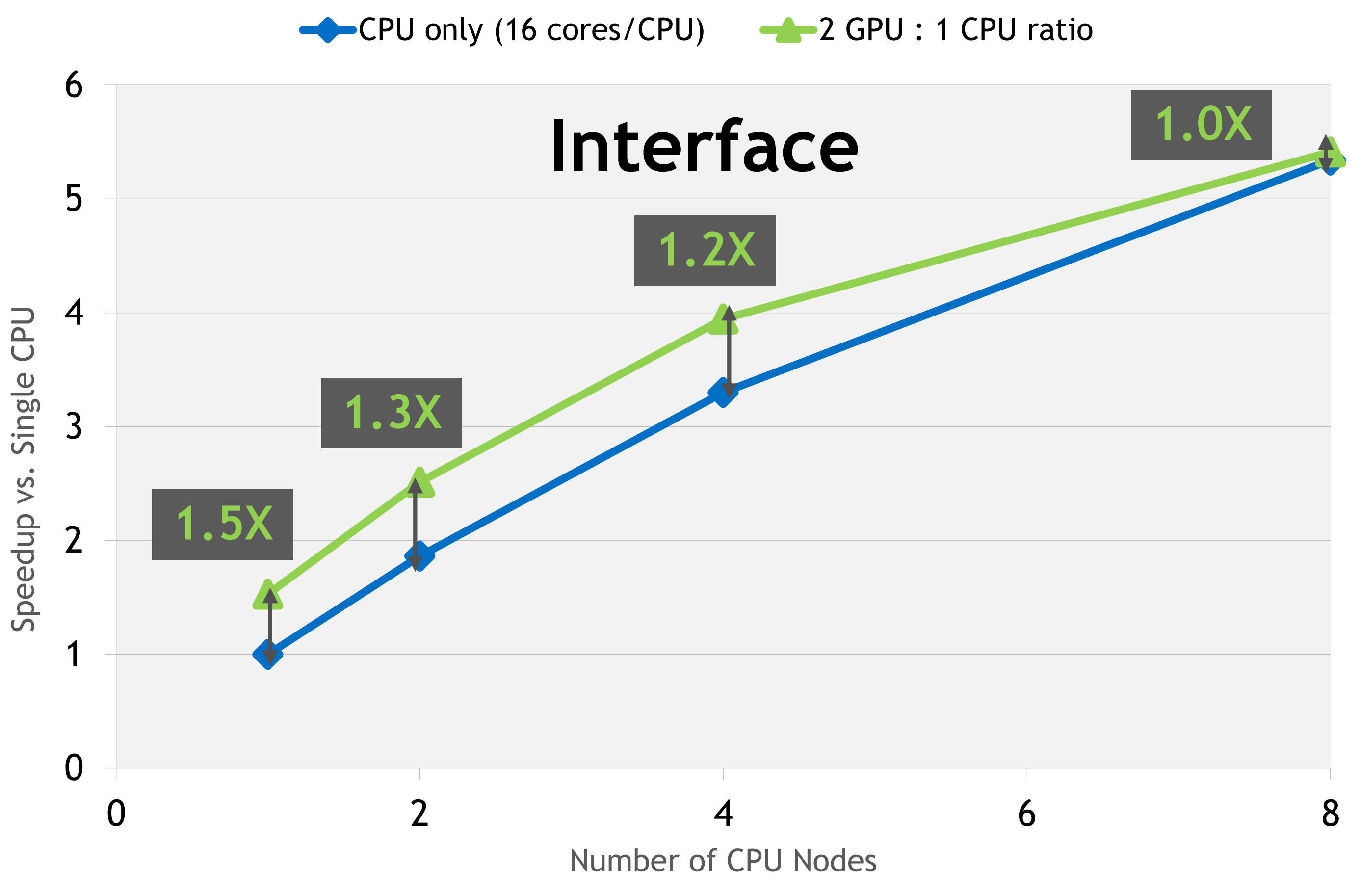
VASP B.hR105 Benchmark



- Measured on Tesla K80 and dual-socket Haswell (16 cores per socket @2.3GHz)
- Default K80 clocks (GPU Autoboost on)
- FDR InfiniBand

➢ *Hybrid Functional with blocked Davicson (ALGO=Normal)*

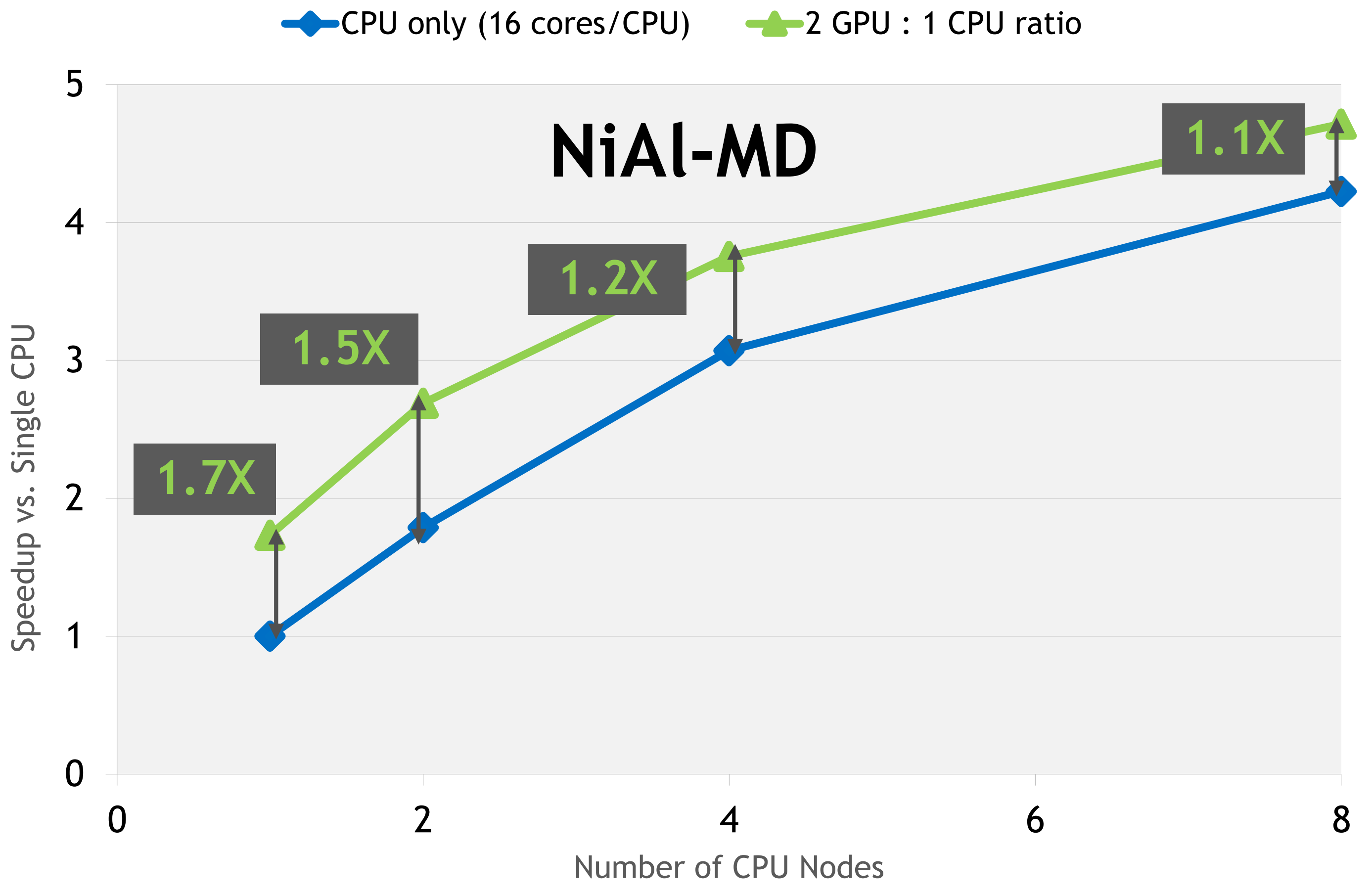
VASP Interface Benchmark



- Measured on Tesla K80 and dual-socket Haswell (16 cores per socket @2.3GHz)
- Default K80 clocks (GPU Autoboost on)
- FDR InfiniBand

➢ *Blocked Davidson + RMM-DIIS (ALGO=Fast)*

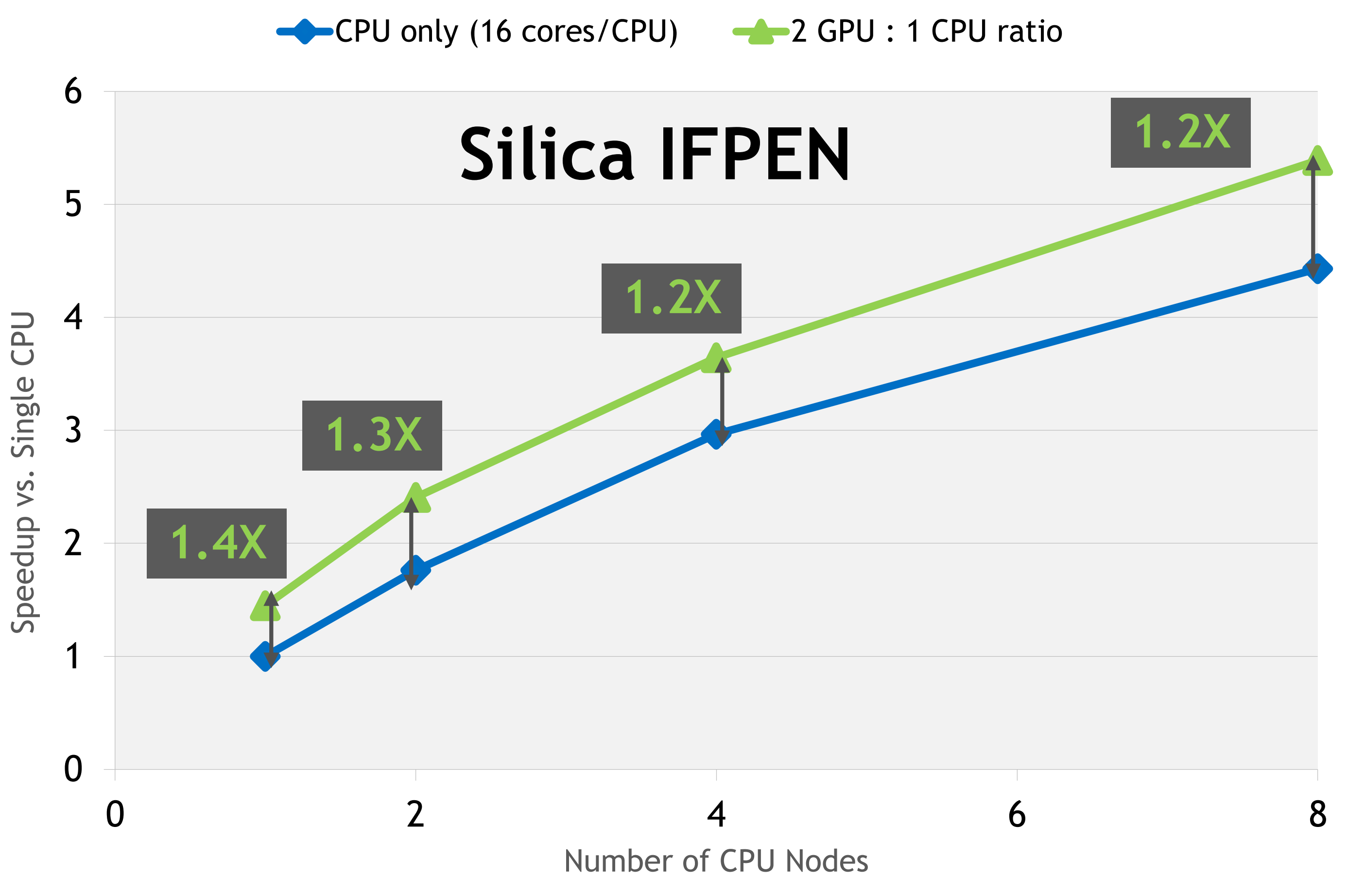
VASP NiAl-MD Benchmark



- Measured on Tesla K80 and dual-socket Haswell (16 cores per socket @2.3GHz)
- Default K80 clocks (GPU Autoboost on)
- FDR InfiniBand

➢ *Blocked Davidson + RMM-DIIS (ALGO=Fast)*

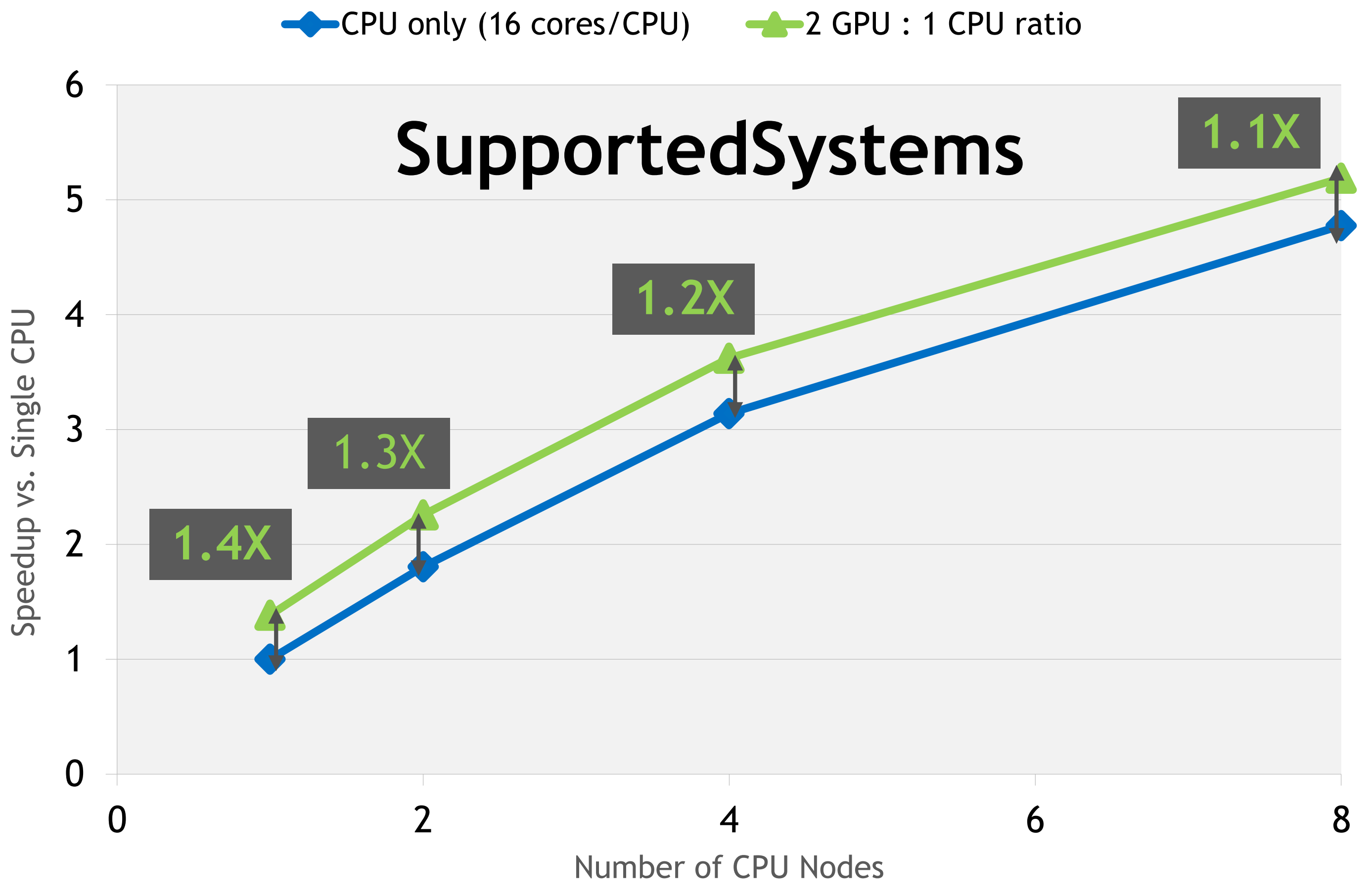
VASP Silica IFPEN Benchmark



- Measured on Tesla K80 and dual-socket Haswell (16 cores per socket @2.3GHz)
- Default K80 clocks (GPU Autoboost on)
- FDR InfiniBand

➢ *RMM-DIIS (ALGO=Veryfast)*

VASP SupportedSystems Benchmark



- Measured on Tesla K80 and dual-socket Haswell (16 cores per socket @2.3GHz)
- Default K80 clocks (GPU Autoboost on)
- FDR InfiniBand

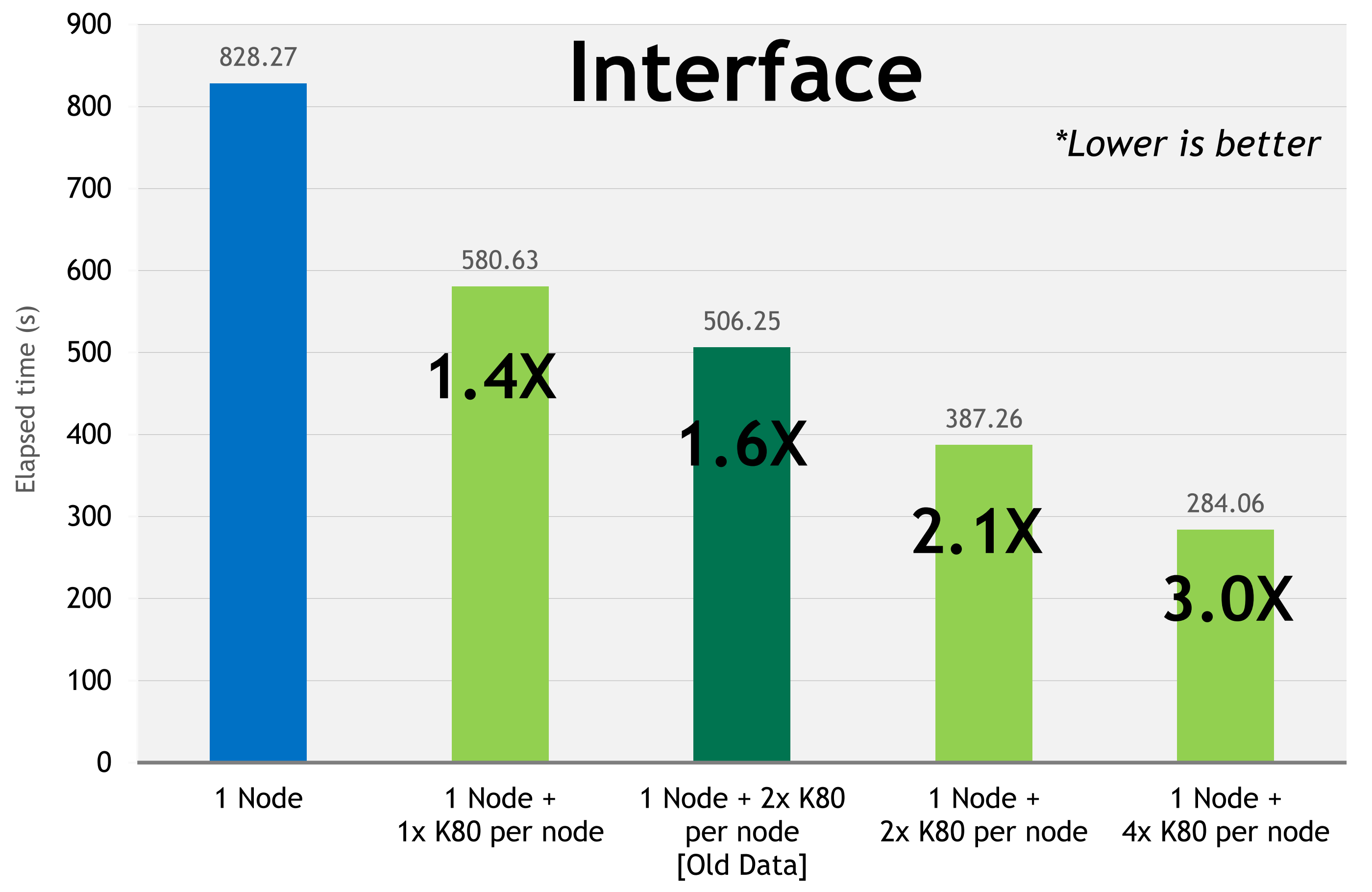
➢ *Blocked Davidson + RMM-DIIS (ALGO=Fast)*

VASP 5.4.1 w/ Patch#1

March 2016

NVIDIA

VASP Interface Benchmark



Running VASP version 5.4.1

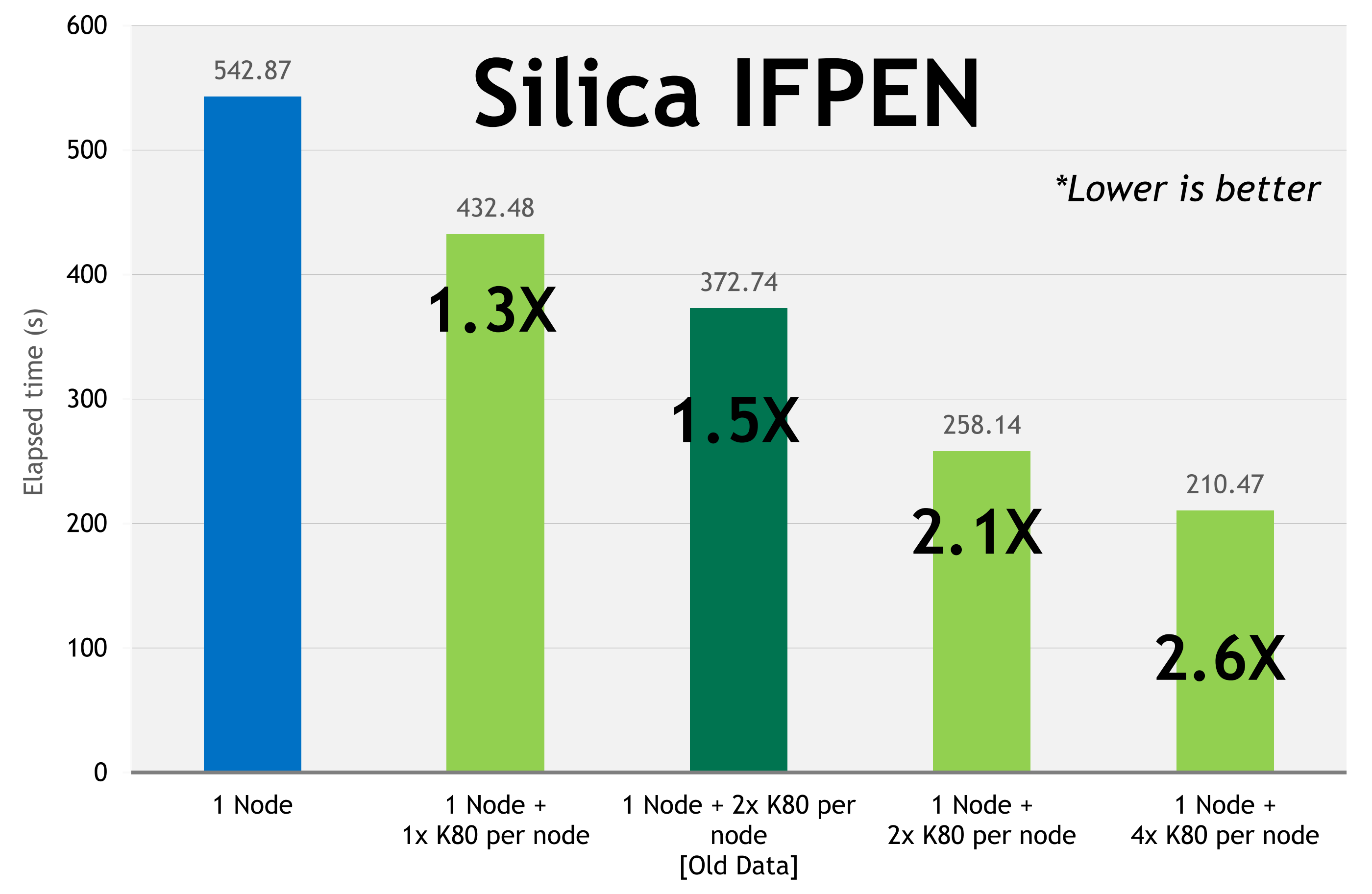
The blue node contains Dual Intel Xeon E5-2698 v3@2.3GHz (Haswell) CPUs

The green nodes contain Dual Intel Xeon E5-2698 v3@2.3GHz (Haswell) CPUs + Tesla K80 (autoboost) GPUs

"[Old Data]" = pre-Bugfix: patch #1 for vasp.5.4.1.05Feb15 which yield up to 2.7X faster calculations. (patch #1 available at VASP.at)

Blocked Davidson + RMM-DIIS (ALGO=Fast)

VASP Silica IFPEN Benchmark



Running VASP version 5.4.1

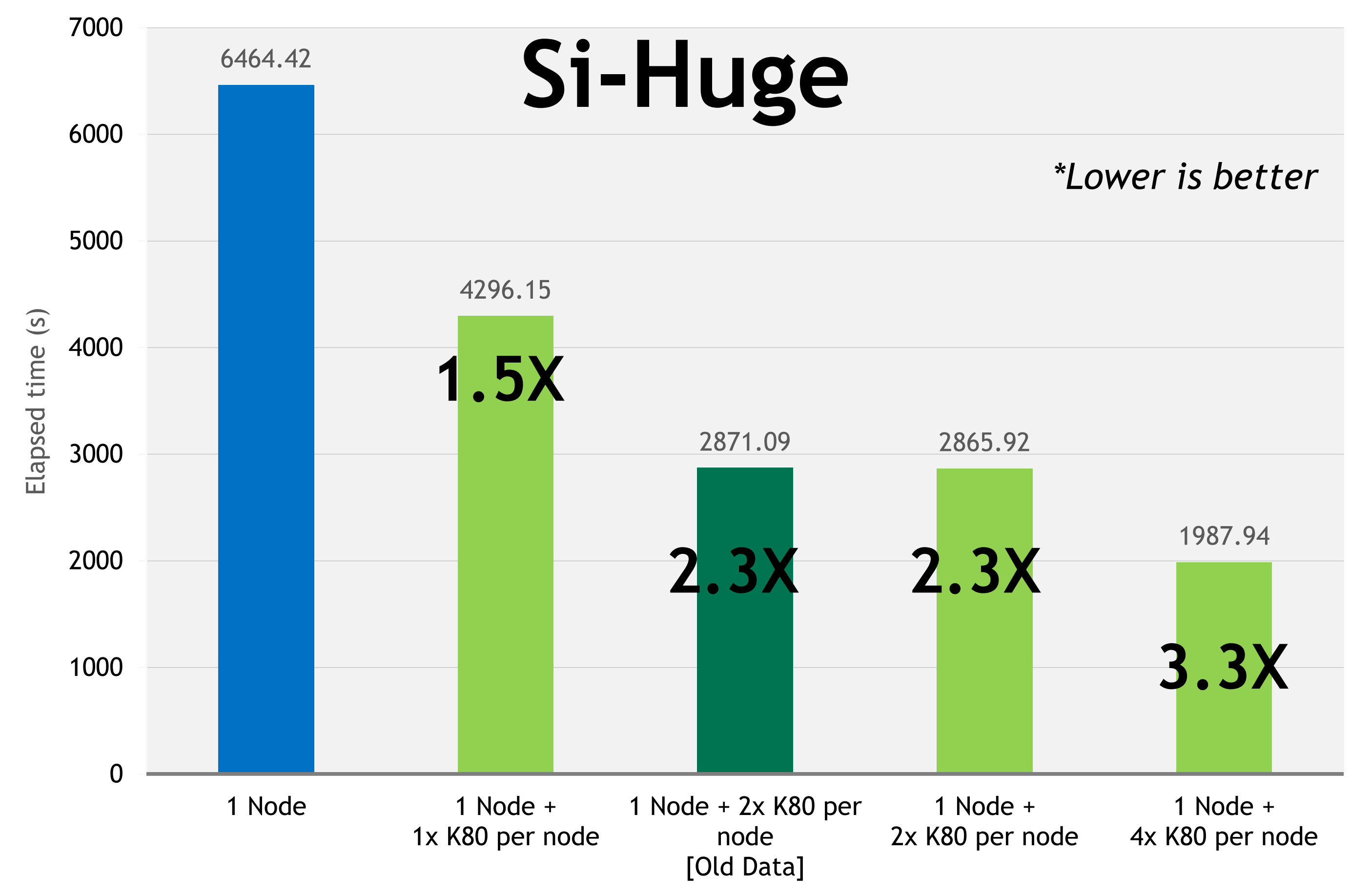
The blue node contains Dual Intel Xeon E5-2698 v3@2.3GHz (Haswell) CPUs

The green nodes contain Dual Intel Xeon E5-2698 v3@2.3GHz (Haswell) CPUs + Tesla K80 (autoboost) GPUs

“[Old Data]” = pre-Bugfix: patch #1 for vasp.5.4.1.05Feb15 which yield up to 2.7X faster calculations. (patch #1 available at VASP.at)

RMM-DIIS (ALGO=Veryfast)

VASP Si-Huge Benchmark



Running VASP version 5.4.1

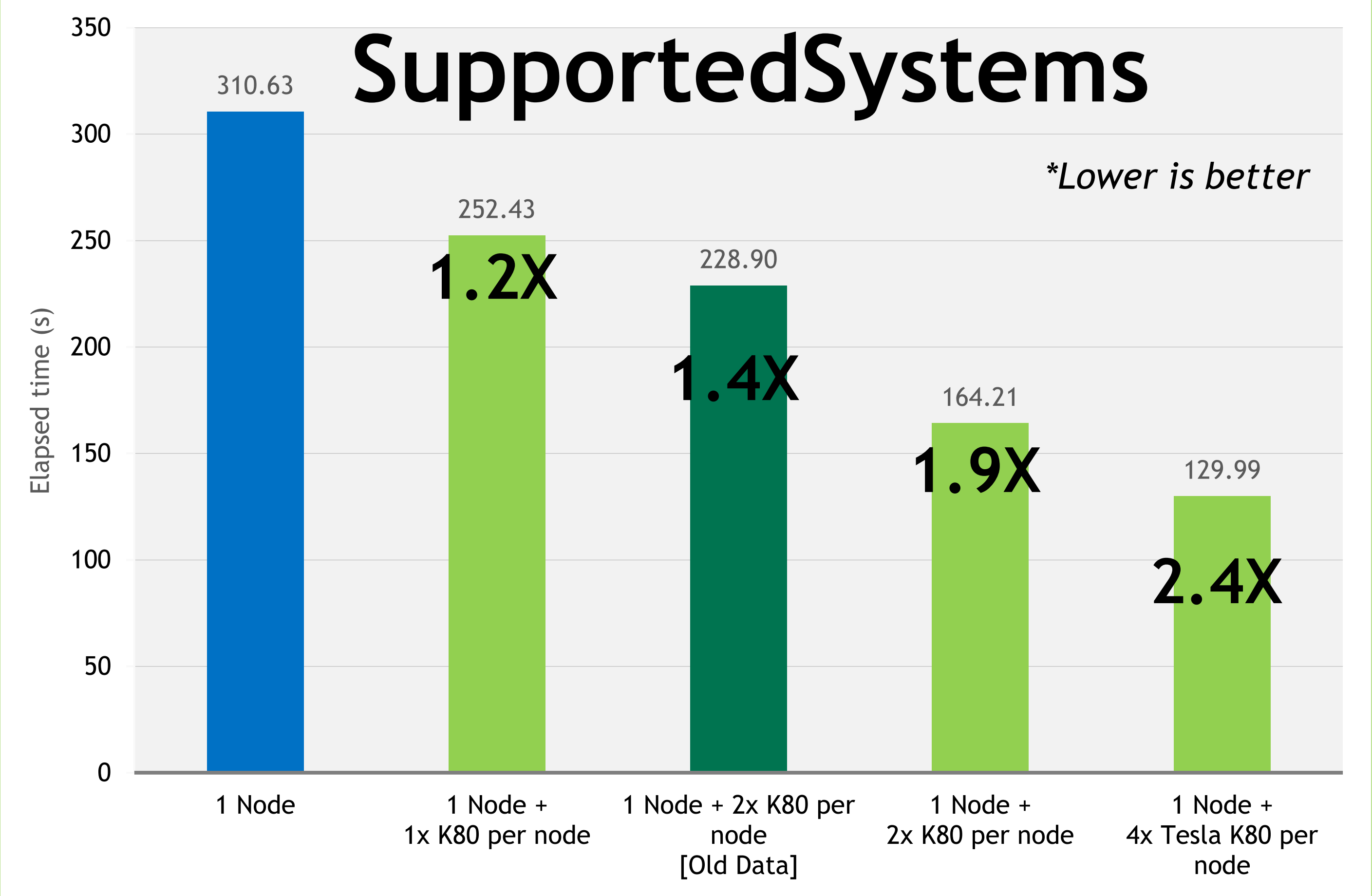
The blue node contains Dual Intel Xeon E5-2698 v3@2.3GHz (Haswell) CPUs

The green nodes contain Dual Intel Xeon E5-2698 v3@2.3GHz (Haswell) CPUs + Tesla K80 (autoboost) GPUs

"[Old Data]" = pre-Bugfix: patch #1 for vasp.5.4.1.05Feb15 which yield up to 2.7X faster calculations. (patch #1 available at VASP.at)

Blocked Davidson + RMM-DIIS (ALGO=Fast)

VASP SupportedSystems Benchmark



Running VASP version 5.4.1

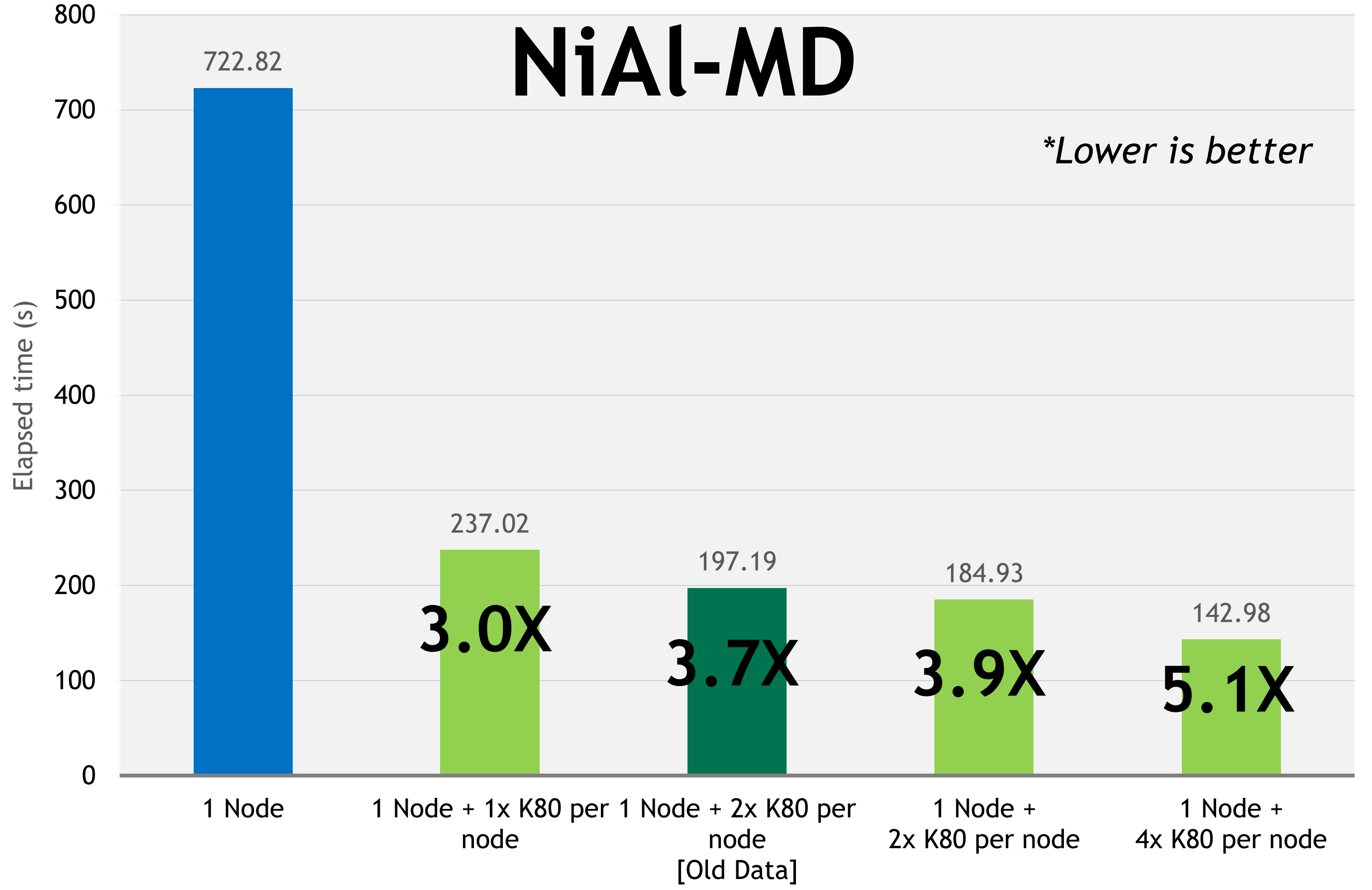
The blue node contains Dual Intel Xeon E5-2698 v3@2.3GHz (Haswell) CPUs

The green nodes contain Dual Intel Xeon E5-2698 v3@2.3GHz (Haswell) CPUs + Tesla K80 (autoboost) GPUs

"[Old Data]" = pre-Bugfix: patch #1 for vasp.5.4.1.05Feb15 which yield up to 2.7X faster calculations. (patch #1 available at VASP.at)

Blocked Davidson + RMM-DIIS (ALGO=Fast)

VASP NiAl-MD Benchmark



Running VASP version 5.4.1

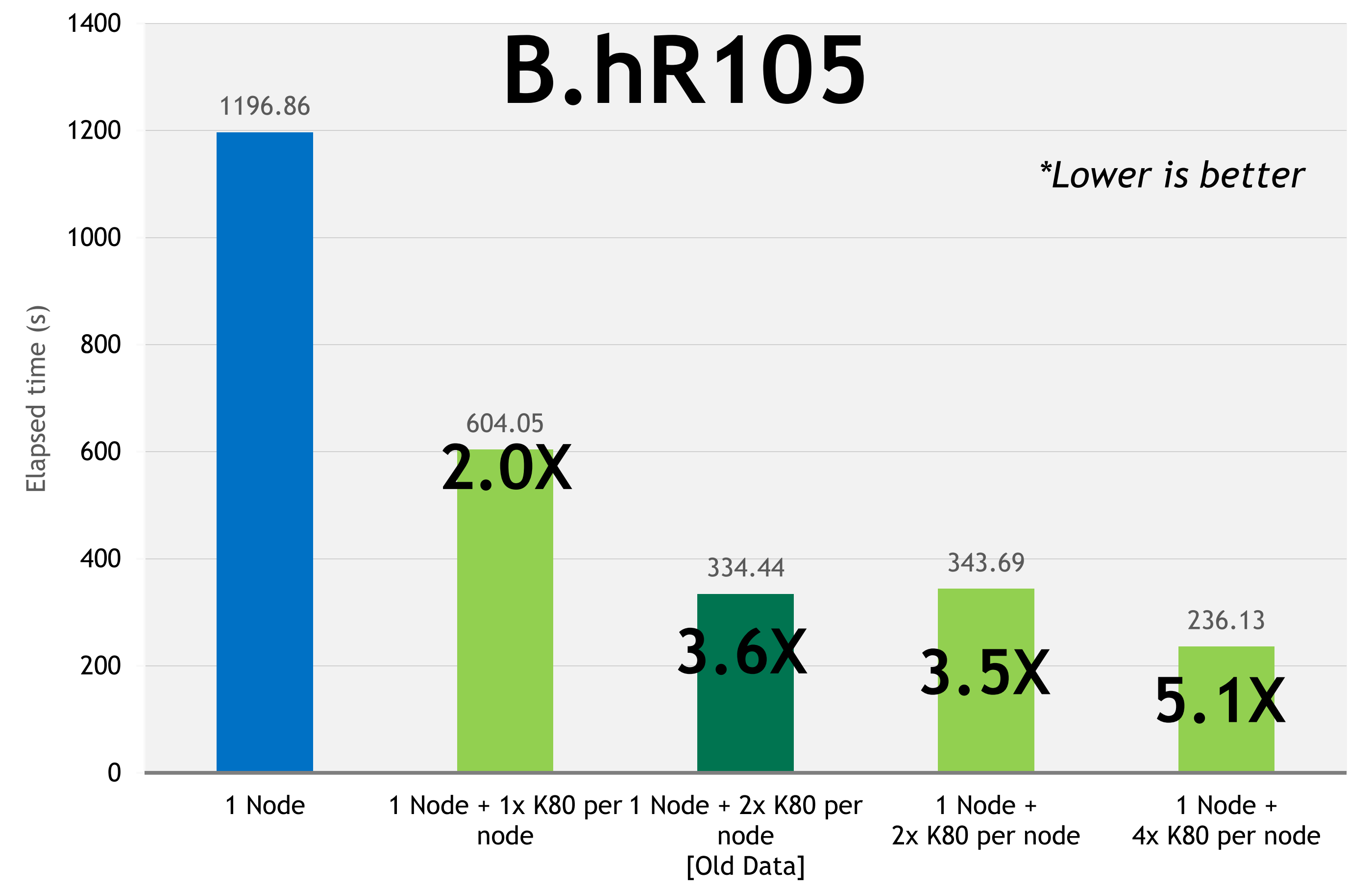
The blue node contains Dual Intel Xeon E5-2698 v3@2.3GHz (Haswell) CPUs

The green nodes contain Dual Intel Xeon E5-2698 v3@2.3GHz (Haswell) CPUs + Tesla K80 (autoboost) GPUs

“[Old Data]” = pre-Bugfix: patch #1 for vasp.5.4.1.05Feb15 which yield up to 2.7X faster calculations. (patch #1 available at VASP.at)

Blocked Davidson + RMM-DIIS (ALGO=Fast)

VASP B.hR105 Benchmark



Running VASP version 5.4.1

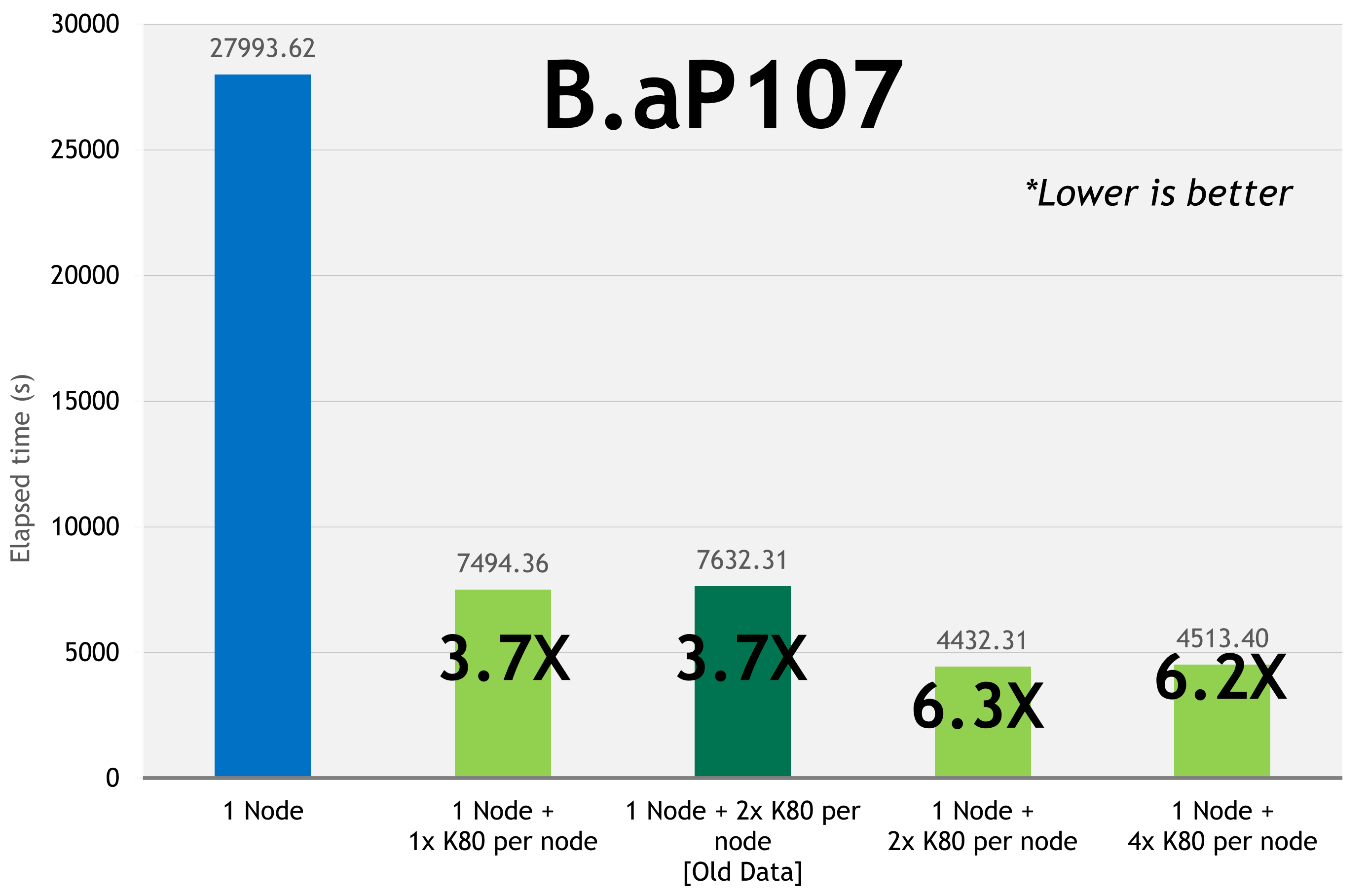
The blue node contains Dual Intel Xeon E5-2698 v3@2.3GHz (Haswell) CPUs

The green nodes contain Dual Intel Xeon E5-2698 v3@2.3GHz (Haswell) CPUs + Tesla K80 (autoboost) GPUs

"[Old Data]" = pre-Bugfix: patch #1 for vasp.5.4.1.05Feb15 which yield up to 2.7X faster calculations. (patch #1 available at VASP.at)

Hybrid Functional with blocked Davicson (ALGO=Normal)

VASP B.aP107 Benchmark



Running VASP version 5.4.1

The blue node contains Dual Intel Xeon E5-2698 v3@2.3GHz (Haswell) CPUs

The green nodes contain Dual Intel Xeon E5-2698 v3@2.3GHz (Haswell) CPUs + Tesla K80 (autoboost) GPUs

"[Old Data]" = pre-Bugfix: patch #1 for vasp.5.4.1.05Feb15 which yield up to 2.7X faster calculations. (patch #1 available at VASP.at)

Hybrid functional calculation (exact exchange) with blocked Davidson. No KPoint parallelization.

Hybrid Functional with blocked Davicson (ALGO=Normal)

Quantum Chemistry (QC) on GPUs

February 11, 2016

NVIDIA