



Adaptive Heterogeneous Computing with OpenCL: Harnessing hundreds of GPUs and CPUs

**Simon McIntosh-Smith simonm@cs.bris.ac.uk
Head of Microelectronics Research
University of Bristol, UK**



Collaborators

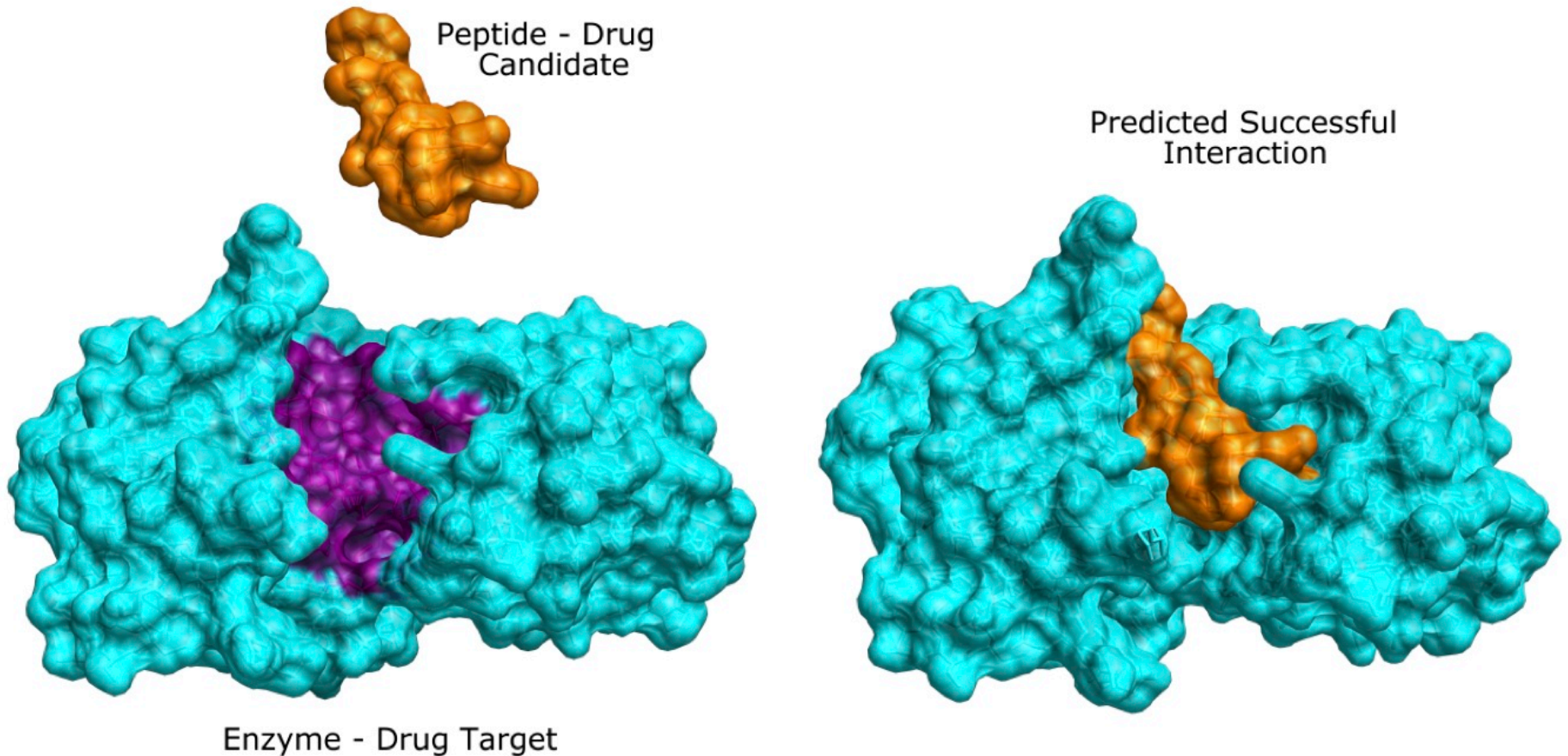
- Richard B. Sessions, Amaurys Avila Ibarra
 - University of Bristol, Biochemistry
 - Developers of the docking code BUDE
- James Price (port to OpenCL)
 - University of Bristol, Computer Science
- Tsuyoshi Hamada, Felipe Cruz (GPUs)
 - University of Nagasaki, Japan
 - Winners of the 2009 Gordon Bell price/performance prize



Molecular docking

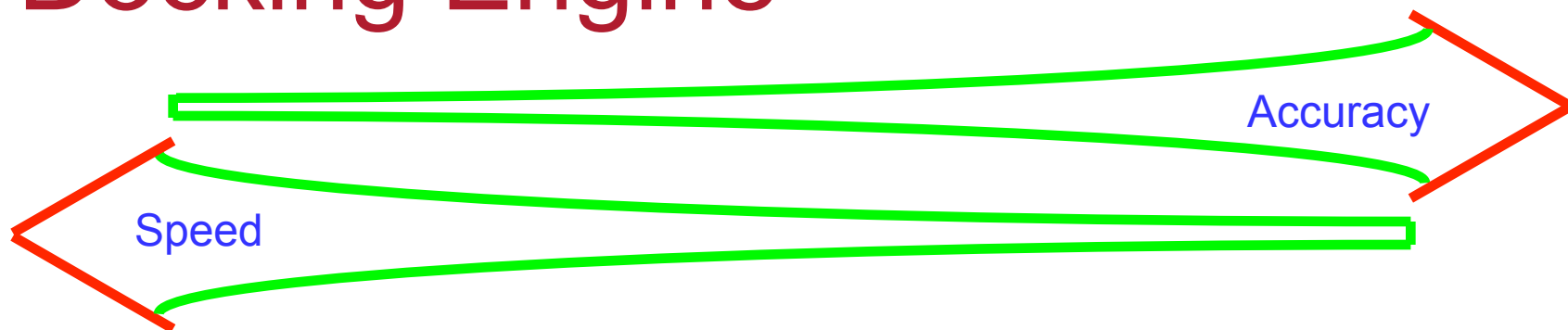


Molecular docking



Proteins typically $O(1,000)$ atoms
Ligands typically $O(100)$ atoms

BUDE: Bristol University Docking Engine



Typical docking scoring functions

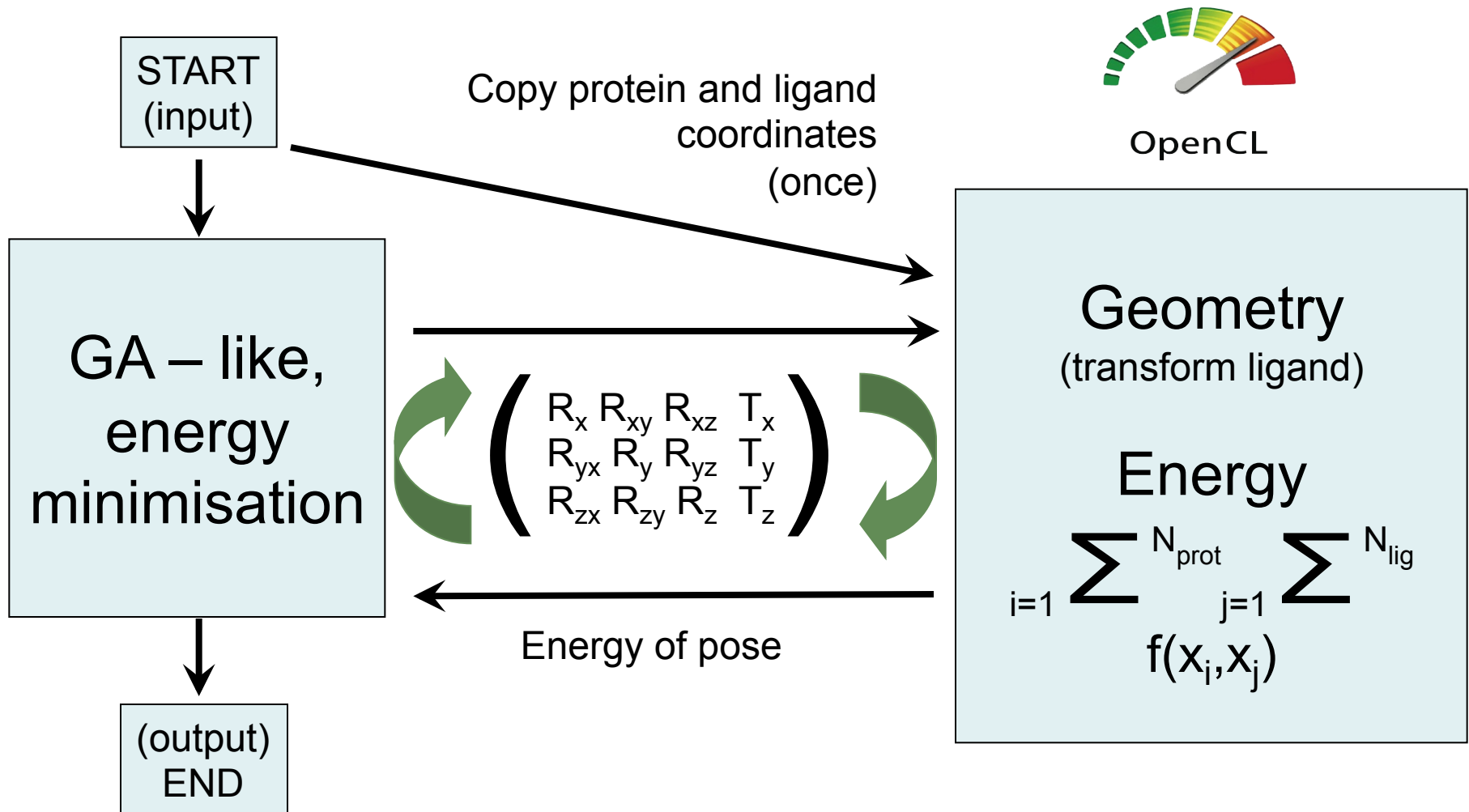
Empirical Free Energy Forcefield
BUDE

Free Energy calculations
MM^{1,2} QM/MM³

Entropy:

solvation	No	Yes	Yes
configurational	Approx	Approx	Yes
Electrostatics	?	Approx	Yes
All atom	No	Yes	Yes
Explicit solvent	No	No	Yes

BUDE Acceleration with OpenCL



Why OpenCL?

- Open standard with fast growing support
 - Evolving rapidly: C++, HSA, OpenCL 2.0, ...
- Platform portability
 - Have used dozens of different GPUs & CPUs
- Naturally supports heterogeneous systems
 - Harness CPUs and GPUs simultaneously

🔥 Multiple levels of parallelism

- $O(10^8)$ conformers from $O(10^7)$ ligands
- $O(10^5)$ poses per conformer (ligand)
- $O(10^3)$ atoms per protein
- $O(10^2)$ atoms per ligand (drug molecule)

- Conformers all independent
- Poses all independent, but there are benefits in grouping all poses of one conformer to one OpenCL device

Parallelism strategy

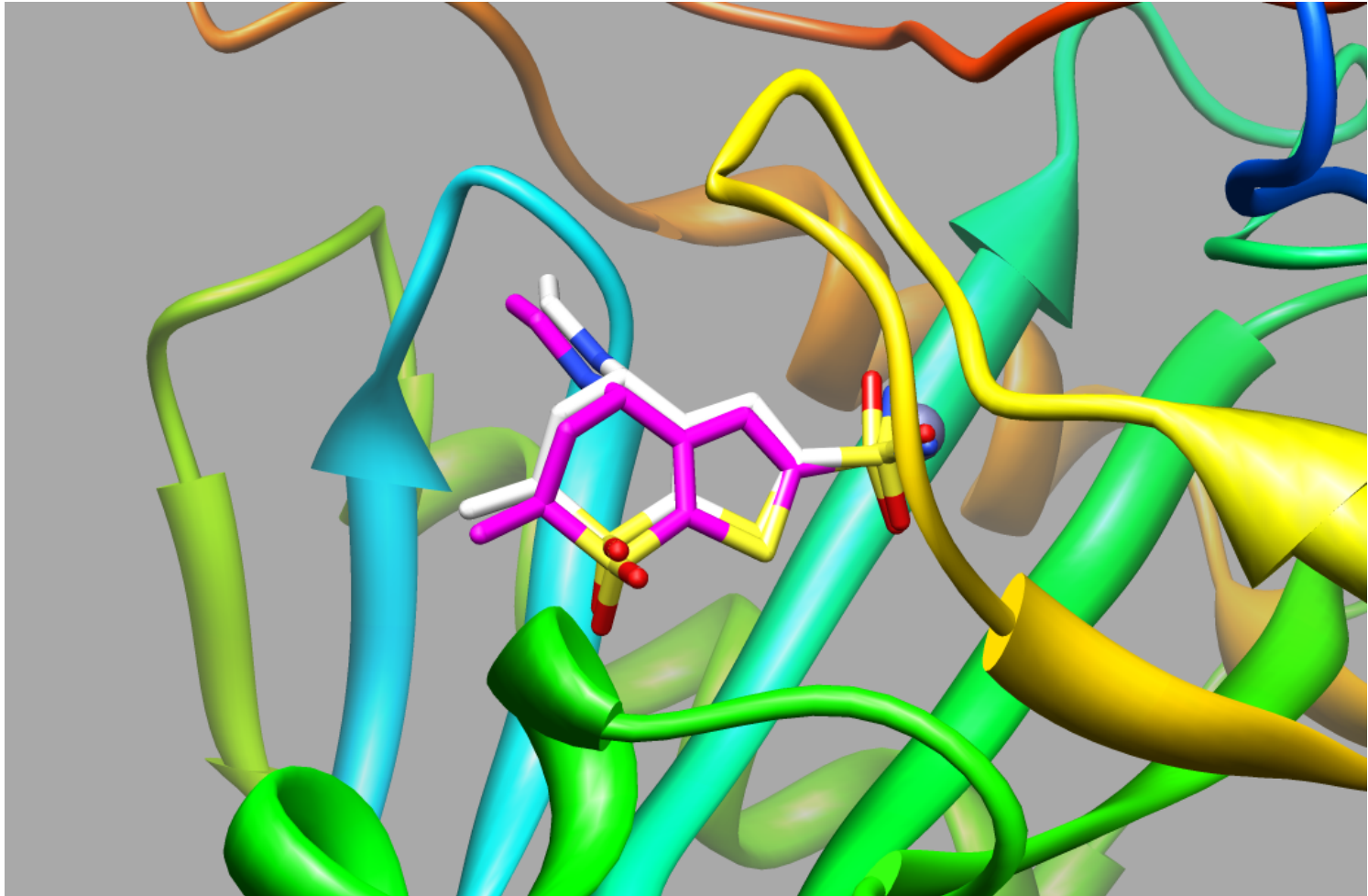
- Distribute ligands across nodes – 10^7 -way parallelism
- All the poses of one conformer distributed across all the OpenCL devices in a node – 10^3 -way parallelism
- Each Work-Item (thread) performs an entire conformer-protein docking – 10^5 atom-atom force calculations



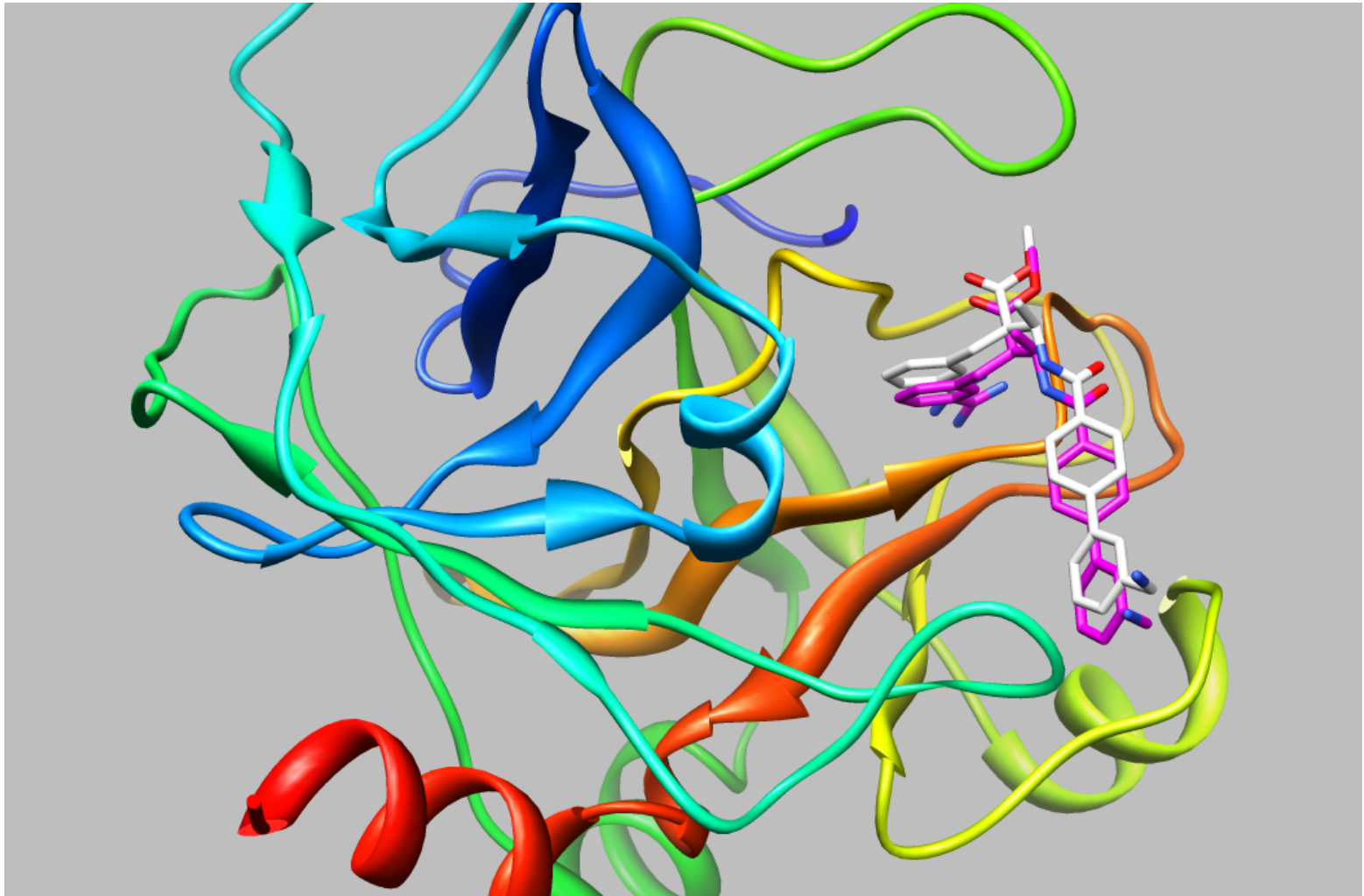
Experimental results



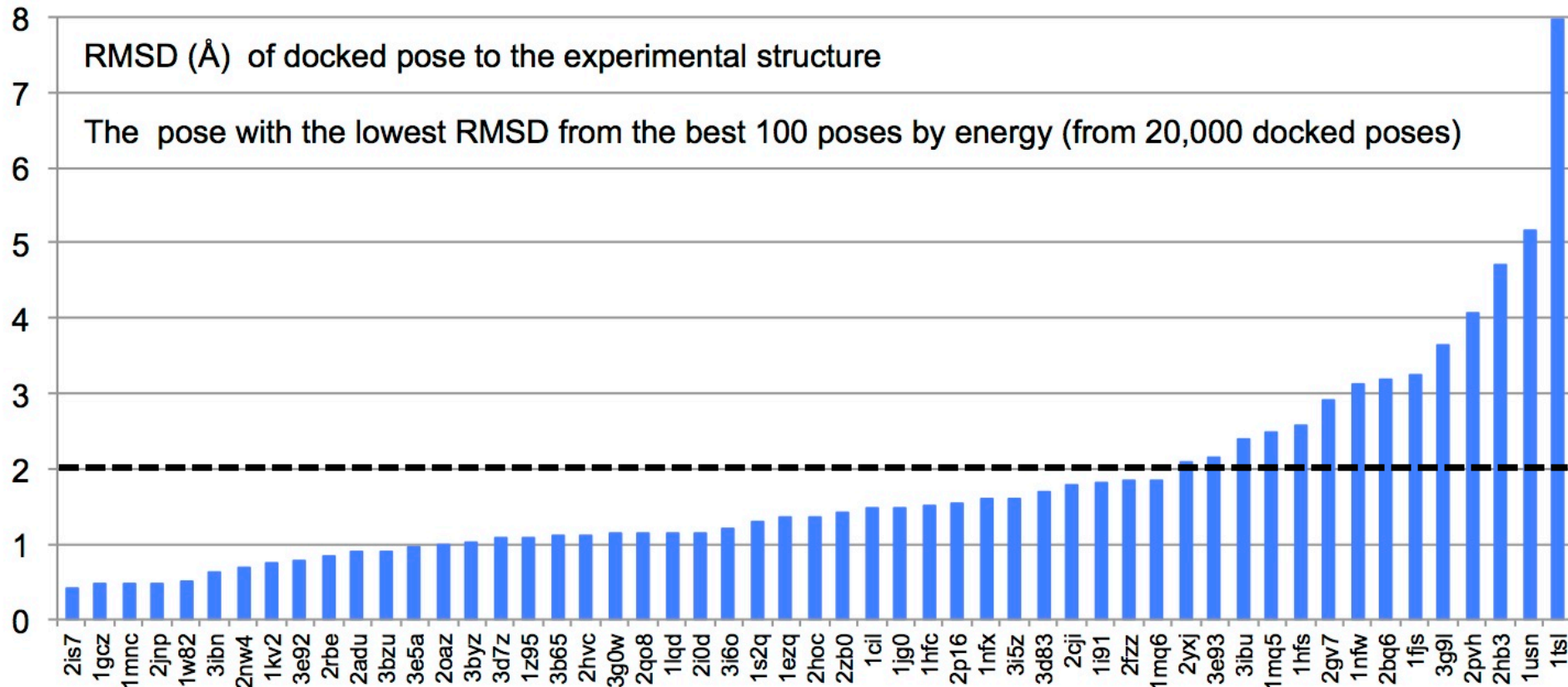
🔥 Redocking into Xray Structure



🌿 Another example



The science is working well



53 Experimental results from the Ligand Binding Database

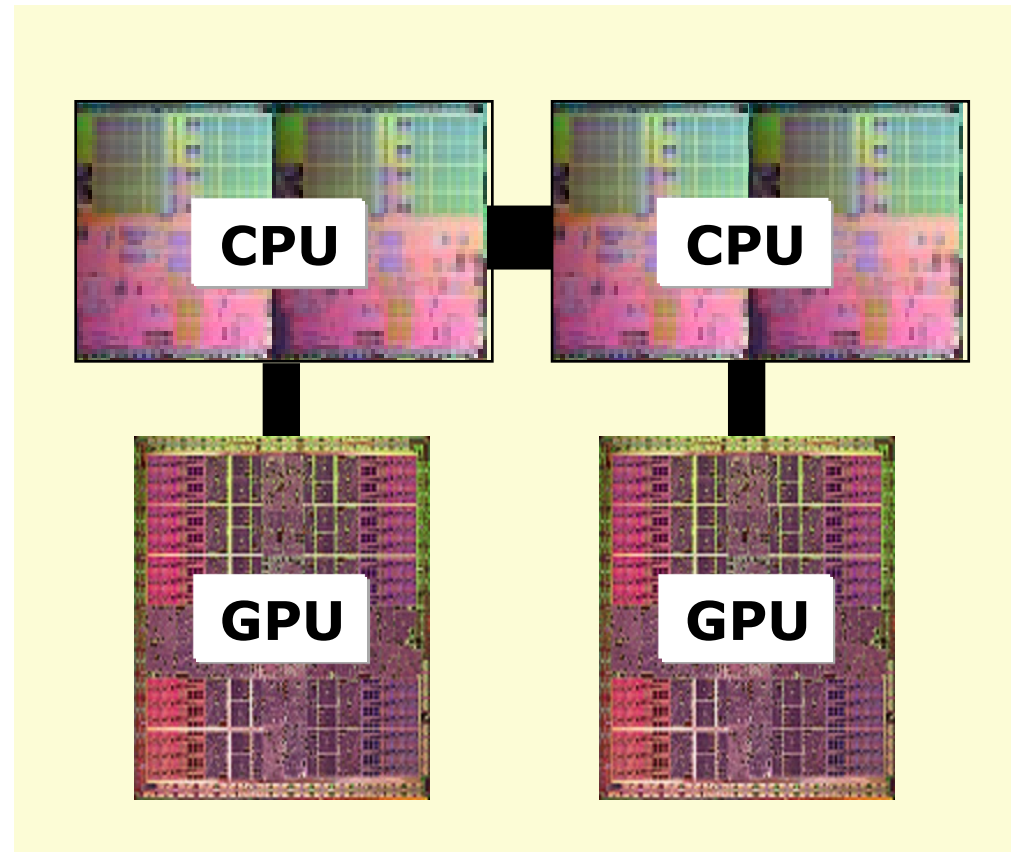
Heterogeneous Systems



🔥 OpenCL for heterogeneous computing

A modern computer includes:

- One or more CPUs
- One or more GPUs



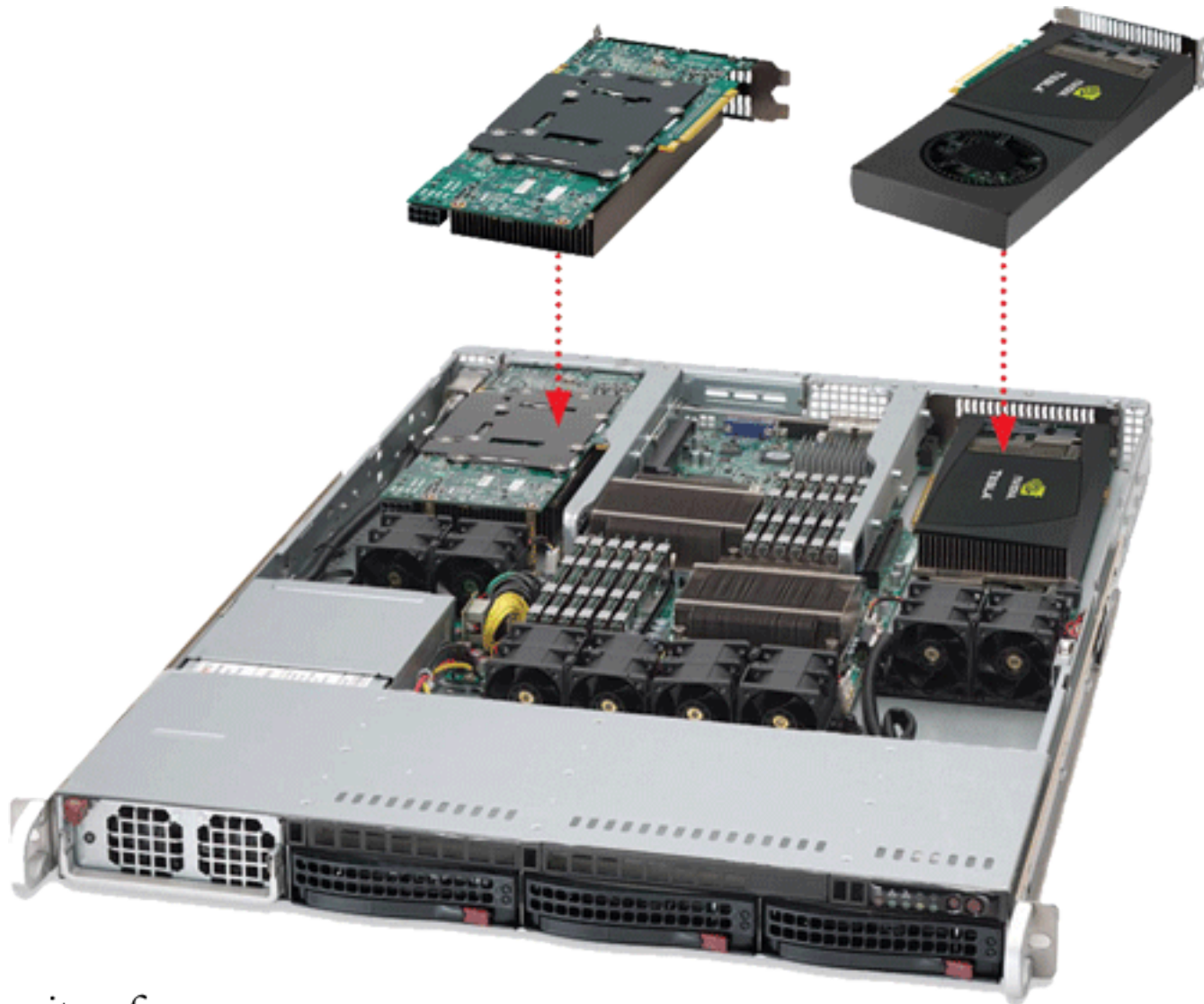
OpenCL (Open Compute Language) lets programmers write a single portable program that uses ALL resources in the heterogeneous platform

🔥 Kinds of Heterogeneous Systems

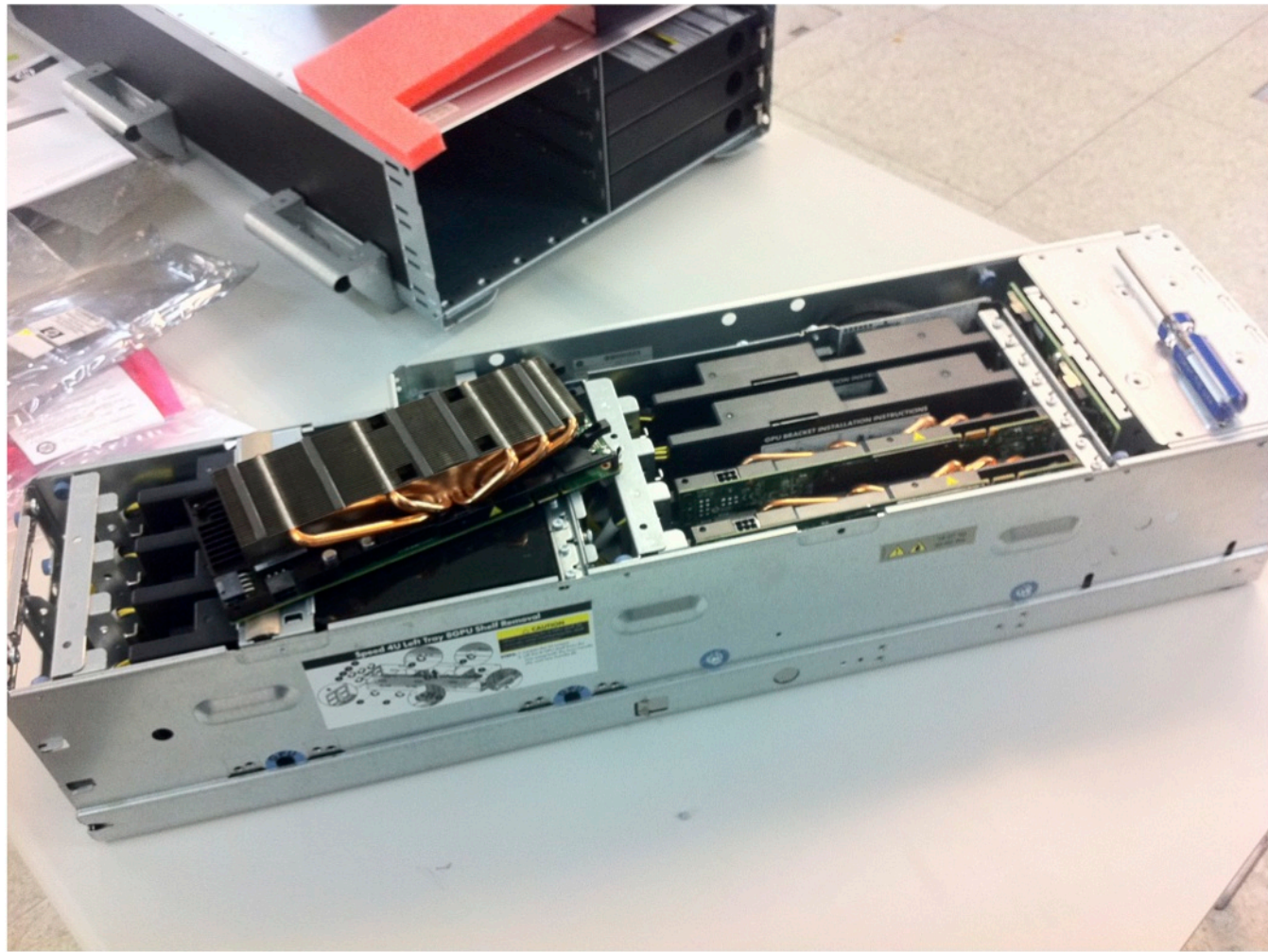


SuperMicro 4 GPU workstation

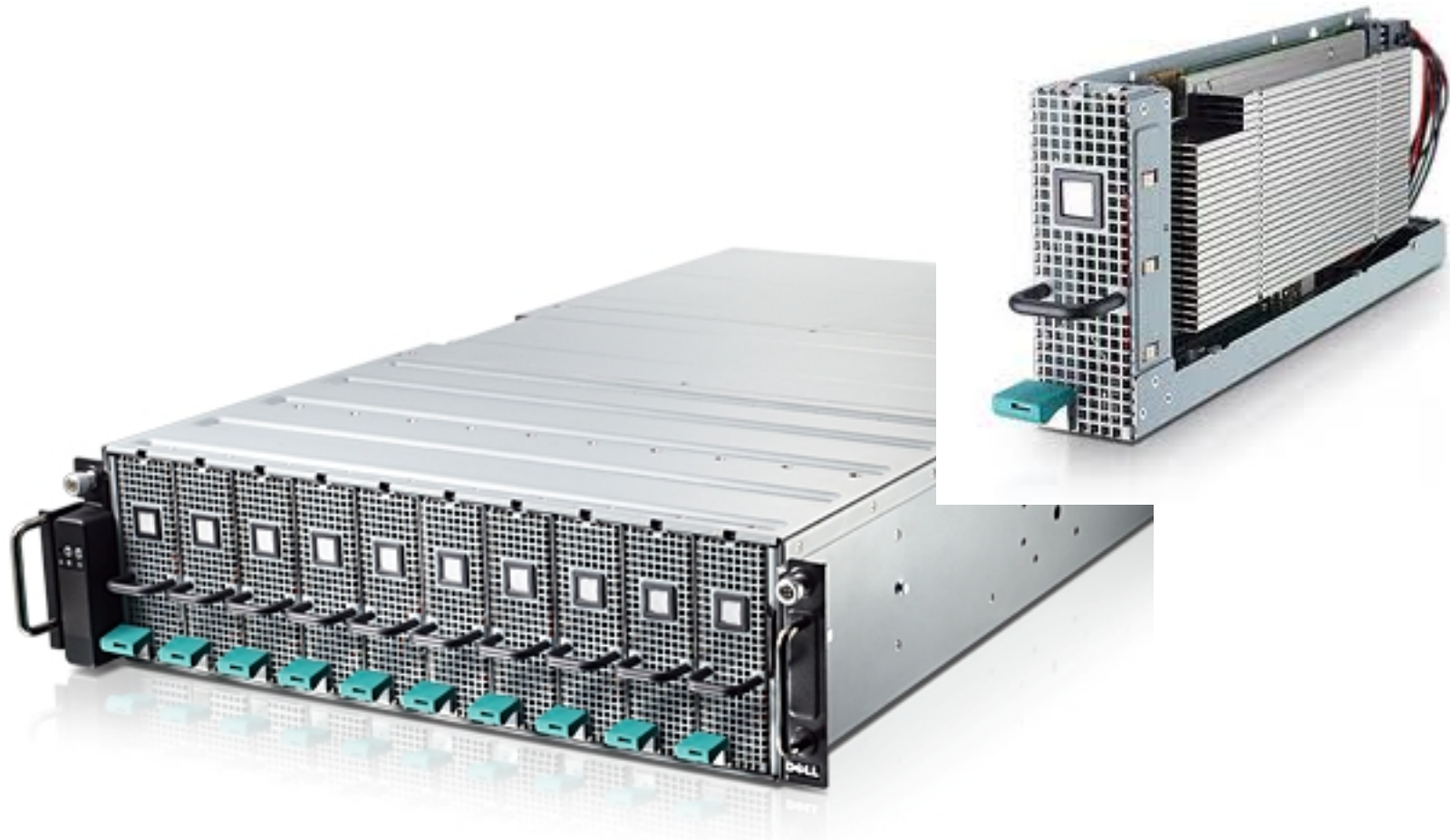
🔥 Kinds of Heterogeneous Systems



🔥 Kinds of Heterogeneous Systems

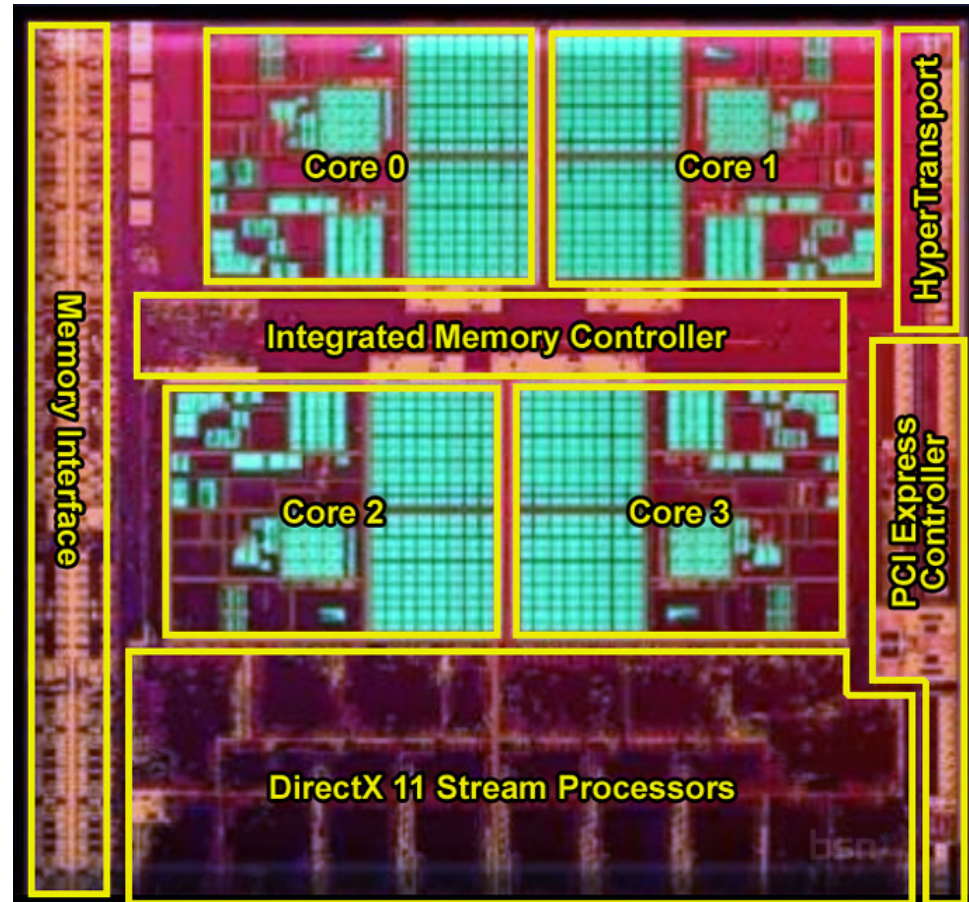


🔥 Kinds of Heterogeneous Systems



Dell PowerEdge C410x – 16 GPUs in 4U

🔥 Kinds of Heterogeneous Systems



AMD Llano Fusion APUs

🔥 Kinds of Heterogeneous Systems



Intel Core2 Duo CPU P8600 @ 2.40GHz,
NVIDIA GeForce 9400M integrated GPU, NVIDIA GeForce 9600M GT discrete GPU



Benchmark results



🔥 BUDE's heterogeneous approach

Within a node:

1. Discover all OpenCL platforms/devices, including CPUs *and* GPUs
2. Run a [micro benchmark](#) on each device, actually a short piece of real work
3. Load balance using micro benchmark results
4. Re-run micro benchmark at regular intervals to adapt to load changes

🔥 BUDE's heterogeneous approach

Between nodes:

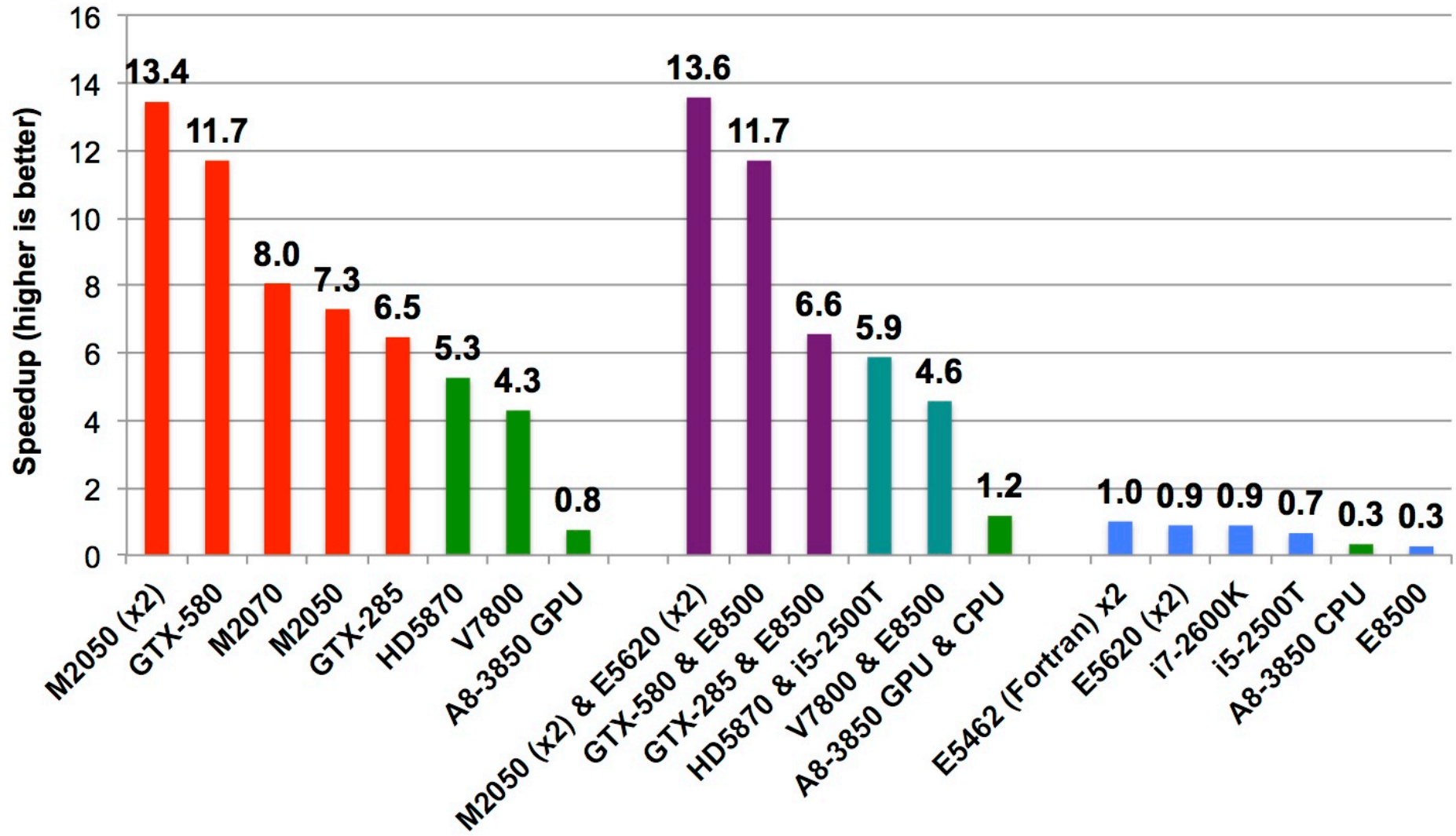
1. Partition entire ligand database (~160M) into subsets that will take a few minutes to process on one node
2. Distribute initial subsets across nodes
3. Work stealing scheduler to load balance across nodes of different performance
4. Time-outs trigger reallocation of MIA subsets to working nodes

🔥 Benchmarking methodology

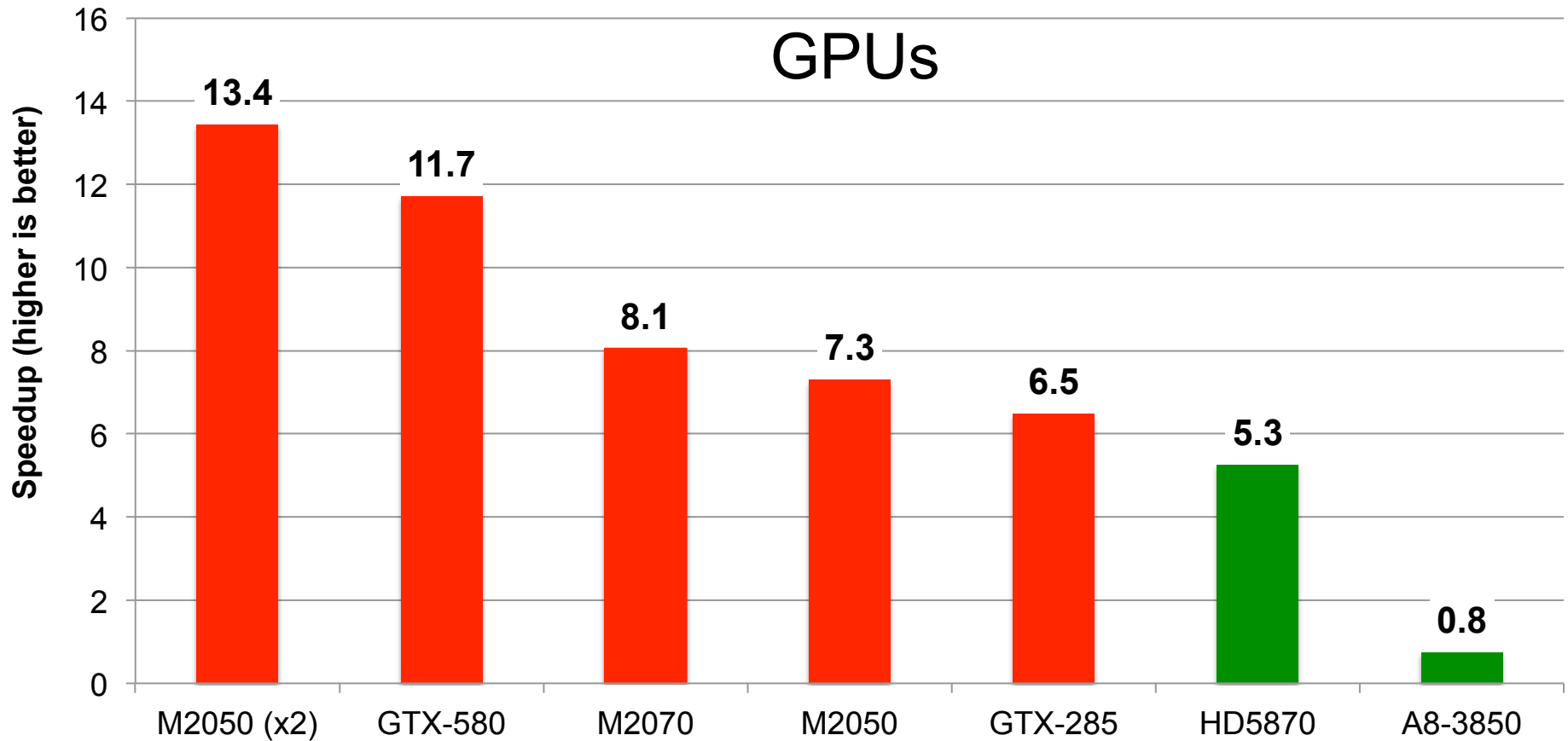
- Use the same power measurement equipment for all the systems under test
- Watts Up? Pro meter
- Measures complete system power 'at the wall'
- Run as fast as possible on all available resources (i.e. all cores or all GPUs simultaneously)



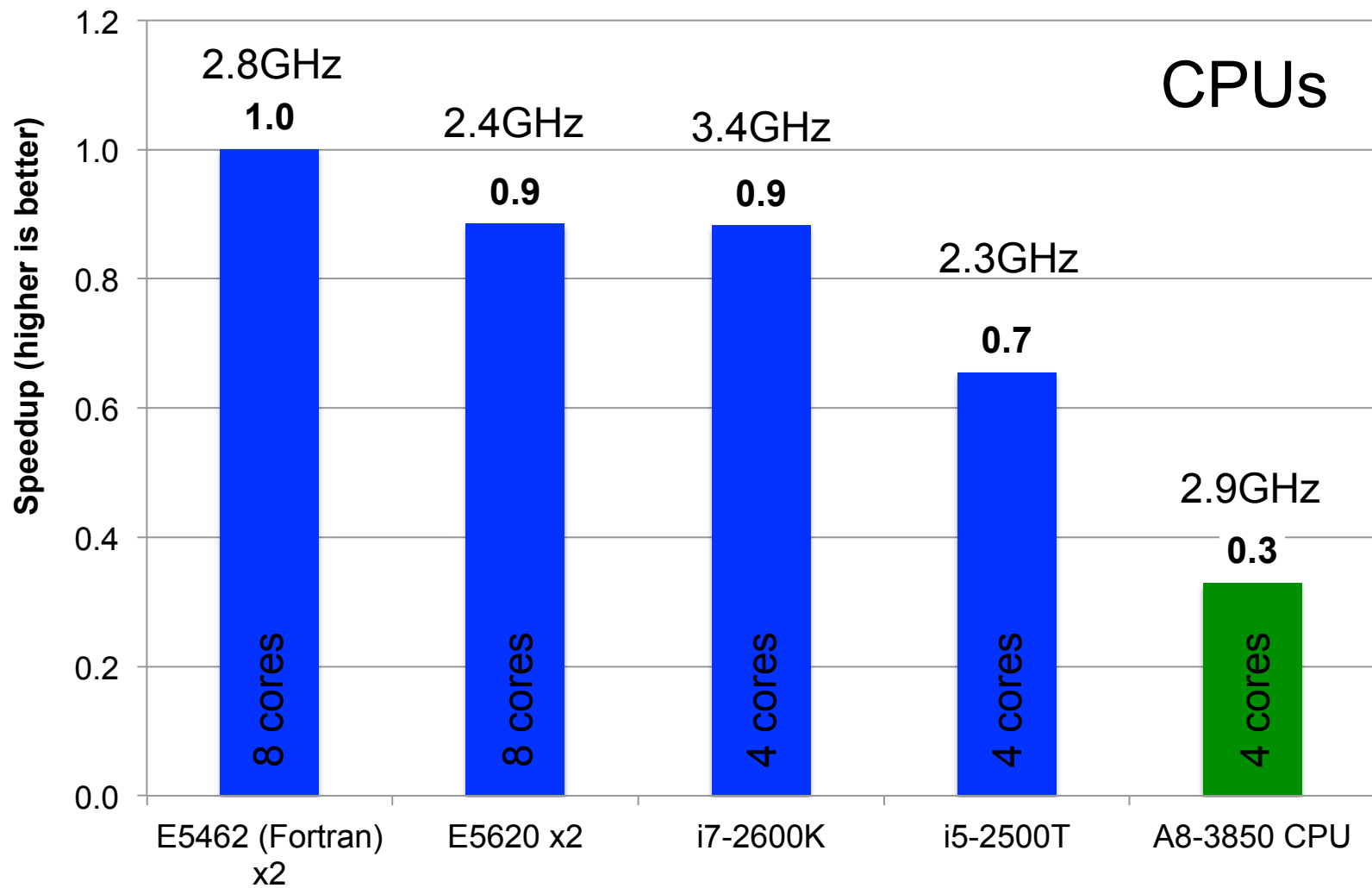
Benchmark results



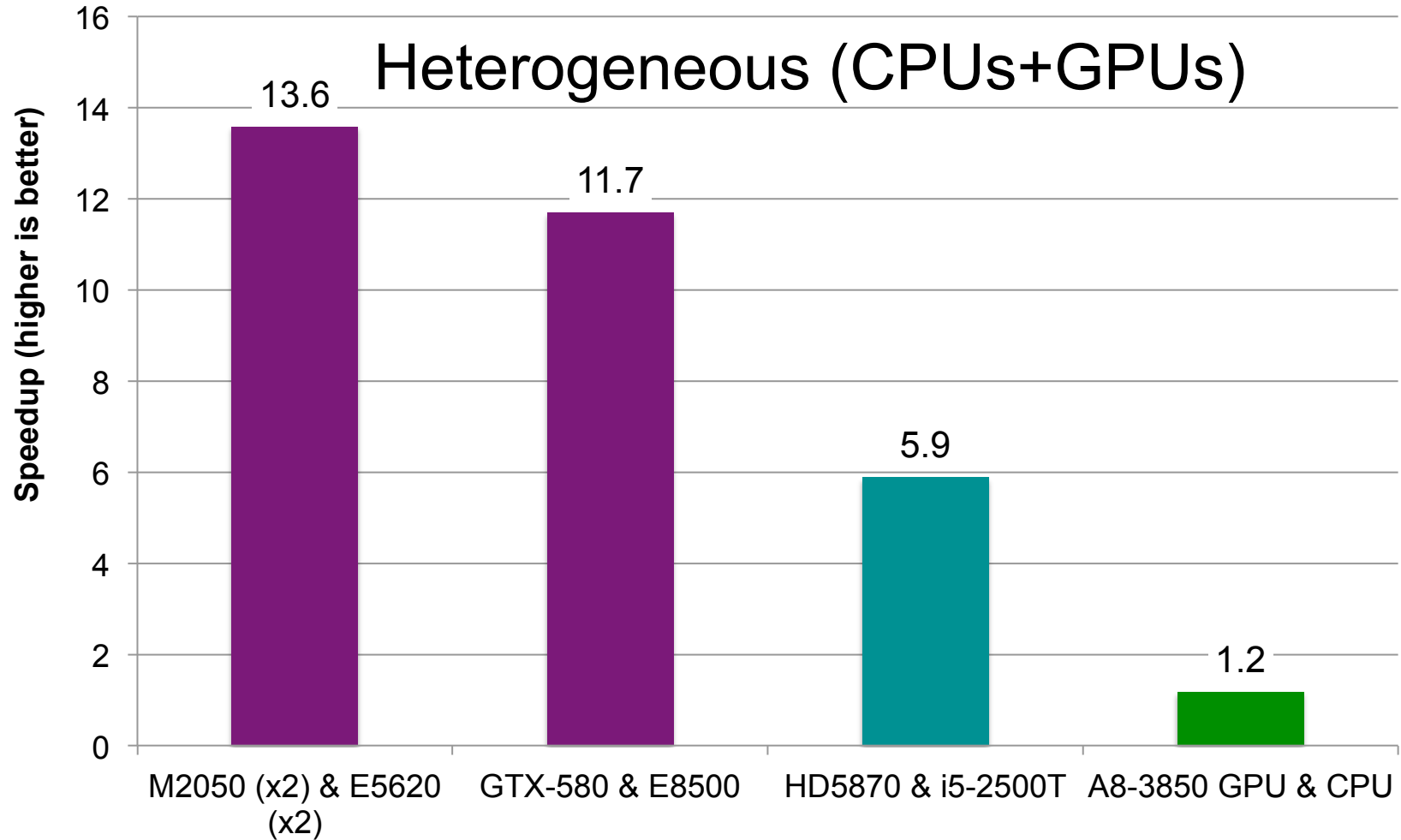
🔥 Performance results



Performance results

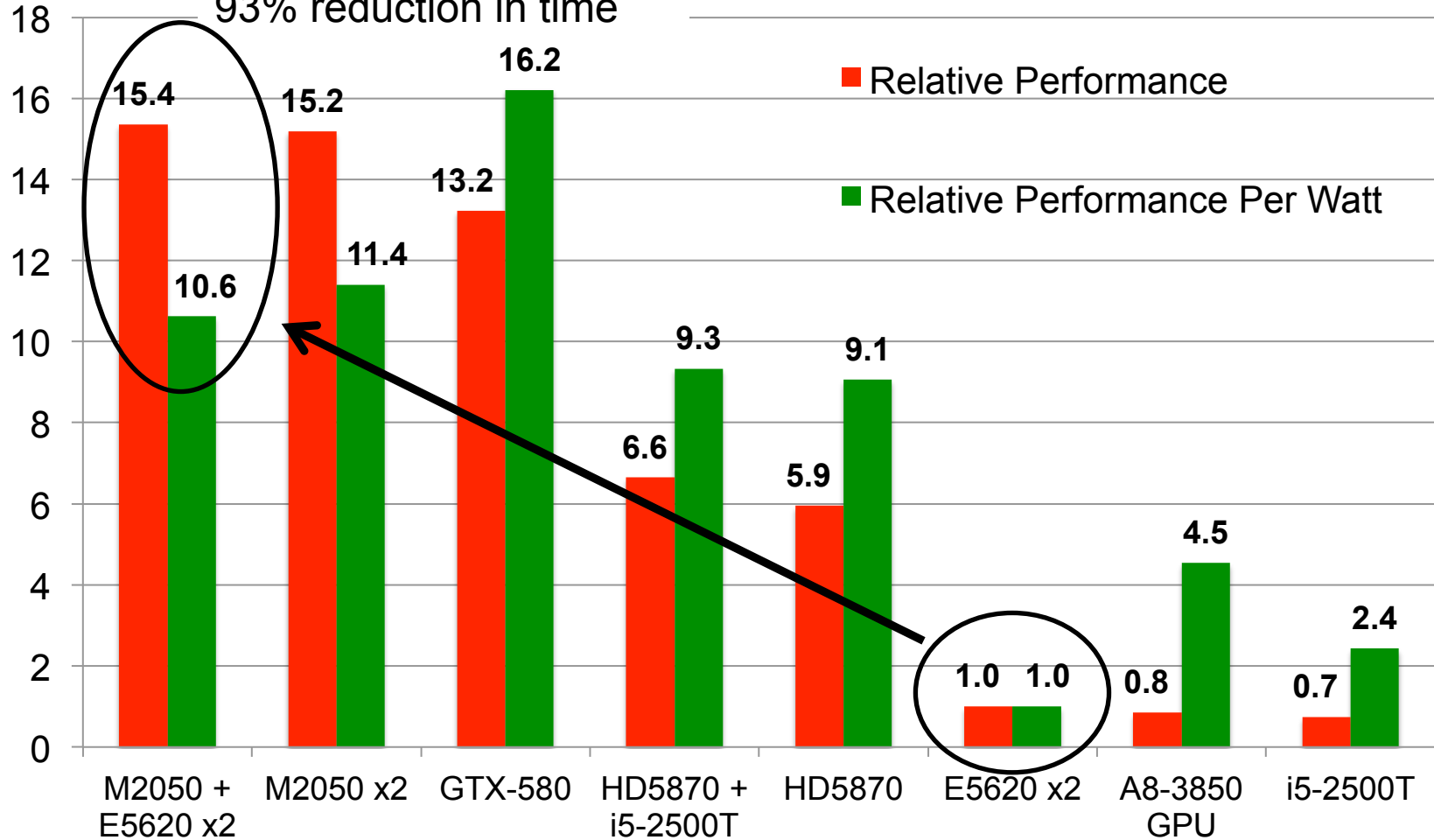


🌿 Performance results



Relative energy and run-time

88% reduction in energy
93% reduction in time





What does this let us do?



🔥 Potentially save lives

New Delhi Metallo-beta-lactamase-1 (NDM-1) is an enzyme that makes bacteria resistant to antibiotics, giving rise to “superbugs”

<http://news.discovery.com/human/superbug-found-in-japan.html>

Discovery News > Human News > Japan Detects Antibiotic-Resistant Superbug

JAPAN DETECTS ANTIBIOTIC-RESISTANT SUPERBUG

The first case of an antibiotic-resistant "superbug" in South Asia has triggered a global health alert.

Tue Sep 7, 2010 01:03 PM ET

Content provided by AFP

(1) Comments | [Leave a Comment](#)

Print

Email

Like

168 people like this. Be the first of your friends.

Share

83

Tweet 51



The New Delhi metallo-lactamase-1 (NDM-1) gene has created a drug-resistant superbug. [Click to enlarge this image.](#)

iStockphoto

THE GIST

- A hospital north of Tokyo detected a drug-resistant "superbug."
- The bacterium has been identified as carrying the New Delhi metallo-lactamase-1 (NDM-1) gene.
- In August a Belgian man became the first such known fatality.

Japan on Monday said it has detected its first case of an antibiotic-resistant "superbug" that surfaced in South Asia and has triggered a global health alert.

A hospital linked to the Dokkyo Medical University in Tochigi prefecture north of Tokyo detected a drug-resistant "superbug," a bacterium carrying the New Delhi metallo-lactamase-1 (NDM-1) gene, in a patient last year, a hospital official told AFP.

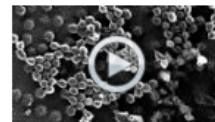
The case follows a warning from the World Health Organization (WHO) last month calling on global health

authorities to monitor the drug-resistant superbug that is believed to have spread from India.

"A patient who came home from India, in his 50s, had fever symptoms while he was hospitalized in May last year, and after a blood test the hospital detected an antibiotics-resistant bacterium," the official said, adding that the patient fully recovered.

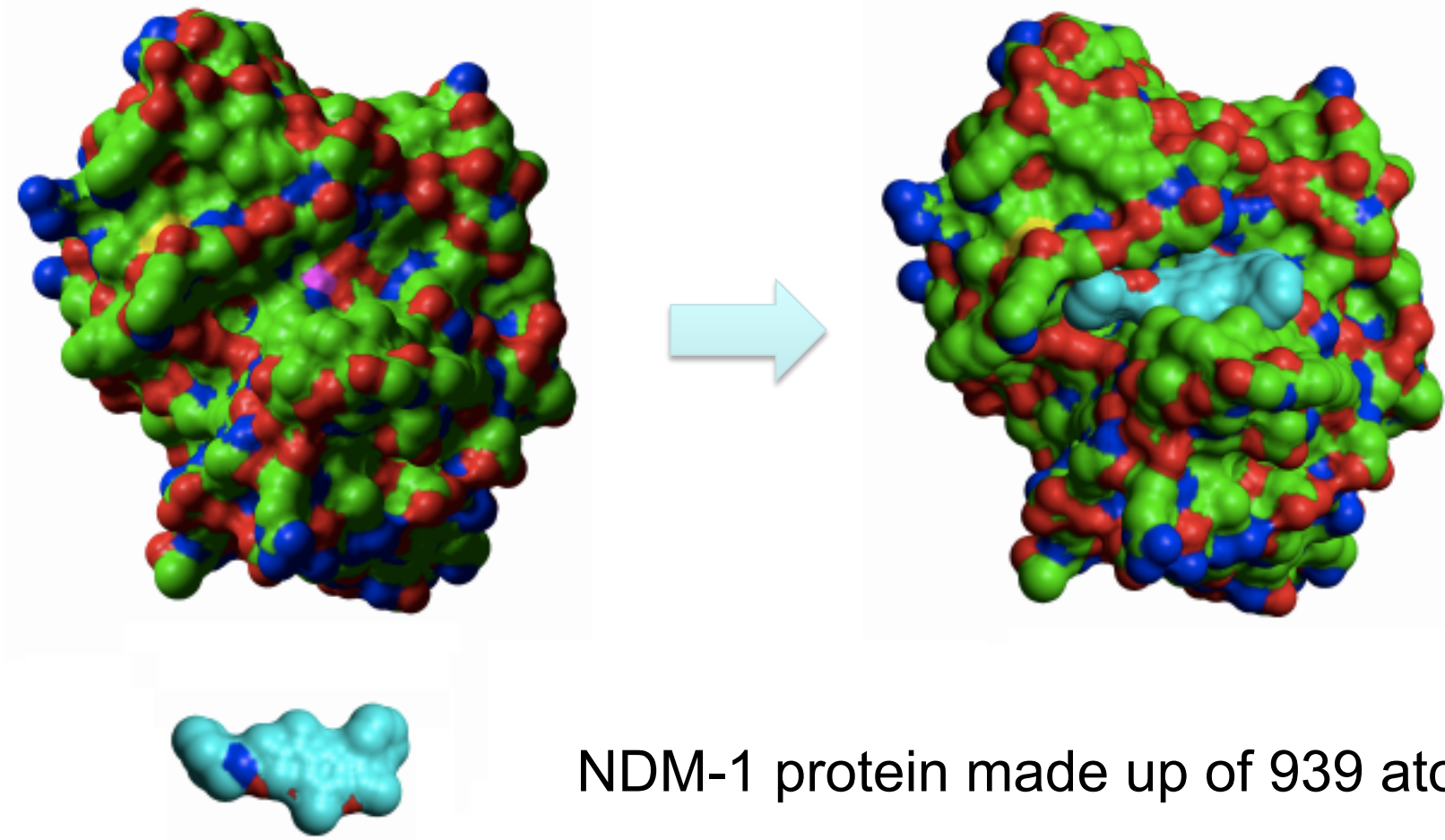
After examination doctors found that the bacterium contained the NDM-1 gene,

RELATED CONTENT



[Take a look at how the swine flu originated and what happens once it](#)

🌟 NDM-1 as a docking target



GPU-system DEGIMA



- Used 222 GPUs in parallel for drug docking simulations
 - ATI Radeon HD5870 (2.72 TFLOPS) & Intel i5-2500T
- ~600 TFLOPS single precision peak performance
- Courtesy of Tsuyoshi Hamada and Felipe Cruz, Nagasaki

NDM-1 experiment

- 7.65 million candidate drug molecules, 21.8 conformers each $\rightarrow 166.7 \times 10^6$ dockings
- 4.168×10^{12} poses calculated
- ~98 hours actual wall-time
- Top 300 “hits” being analysed, down selecting to 10 compounds for further investigation in the lab



Future work



Future work

- Run on new, 372 GPU “Emerald” cluster
 - HP SL390 system, Nvidia M2090s GPUs
 - Also Gnodal fast Ethernet interconnect
- Further improve forcefield (underway)
 - RMSD already under 2Å, down from >4Å
- Port to emerging, OpenCL systems
 - E.g. ARM-based, Imagination, ...
- Benchmark on Kepler, Graphics Core Next
- Explore fault tolerance within a node

Conclusions

- OpenCL enables **truly heterogeneous computing**, harnessing all hardware resources in a system
- GPUs can yield **significant savings in energy costs** (and equipment costs) as well as performance
- OpenCL can work well for multi-core CPUs as well as for GPUs

For the first time we can screen libraries of millions of molecules against complex targets using highly accurate, computationally-expensive methods in hours using equipment costing $O(\$100K)$

References

- S. McIntosh-Smith, T. Wilson, A.A. Ibarra, J. Crisp and R.B. Sessions, "Benchmarking energy efficiency, power costs and carbon emissions on heterogeneous systems", The Computer Journal, September 12th 2011. DOI: 10.1093/comjnl/bxr091
- N. Gibbs, A.R. Clarke & R.B. Sessions, "Ab-initio Protein Folding using Physicochemical Potentials and a Simplified Off-Lattice Model", Proteins 43:186-202,200