Divide-and-Conquer Molecular Simulation: GROMACS, OpenMM & CUDA

ISC 2010-05-31

Erik Lindahl

lindahl@cbr.su.se

Center for Biomembrane Research
Stockholm University

These will soon be small computers

~2024: 1B 'cores'

2022: ~300M cores

2020: ~100M cores

2018: ~30M cores

2016: ~10M cores

2014: ~3M cores

2012: ~1M cores

2010: ~300,000 cores

How will YOU use a billion cores?

We're all doing Embarrassing Parallelism

But not the way you think.

We're investing huge efforts in parallelizing algorithms that only reach 50-75% scaling efficiency on large problems

Not a chance they will scale to 1B cores

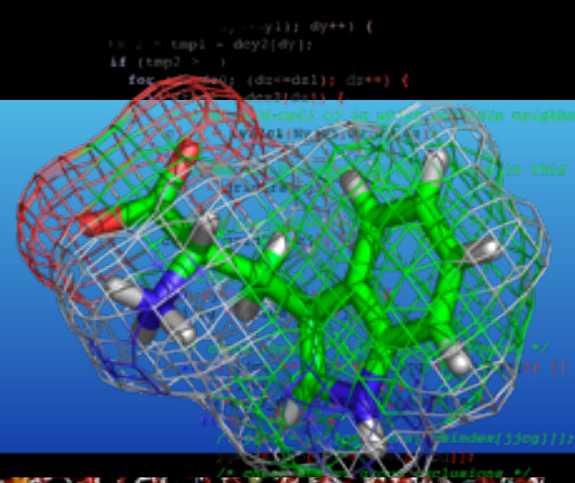
Close-to-useless for smaller problems of commercial interest

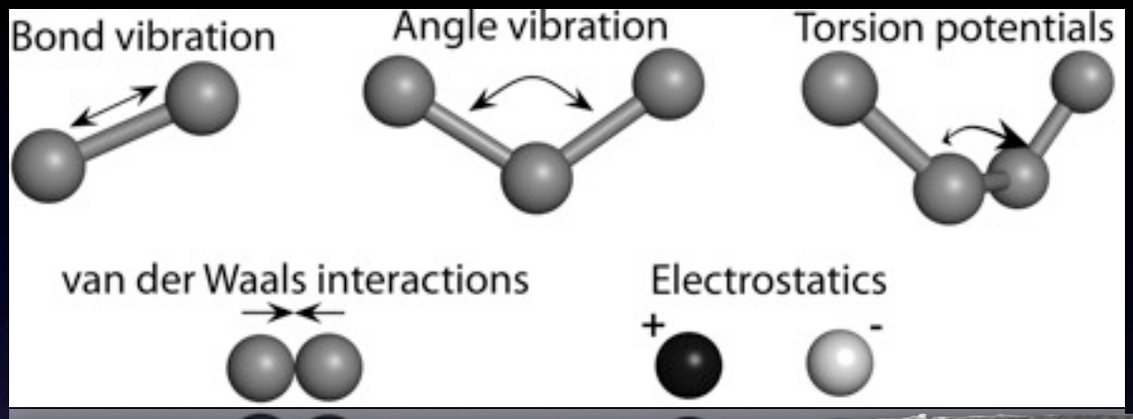
100% focus on programs, forget the problem we're solving

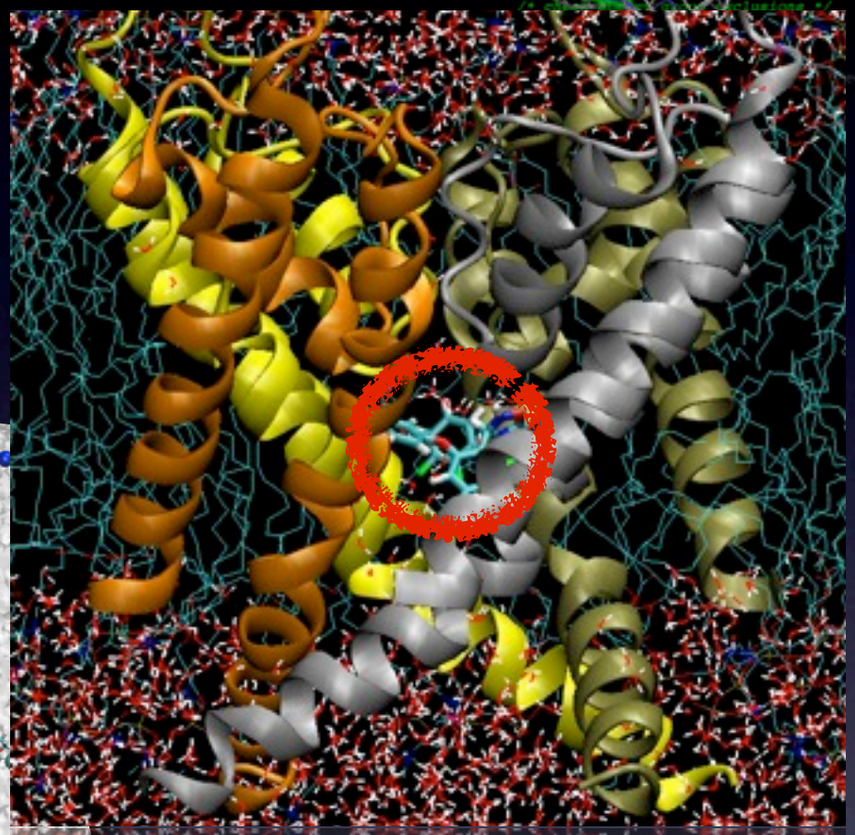
Ask taxpayers to foot the bill

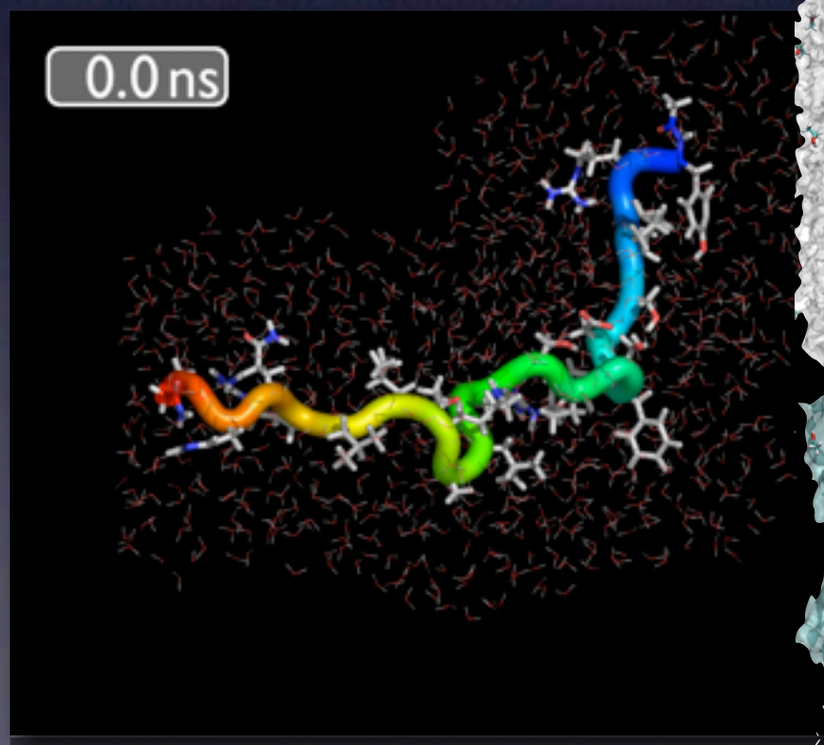
Pretty much the definition of 'embarrassing'?

Molecular Dynamics

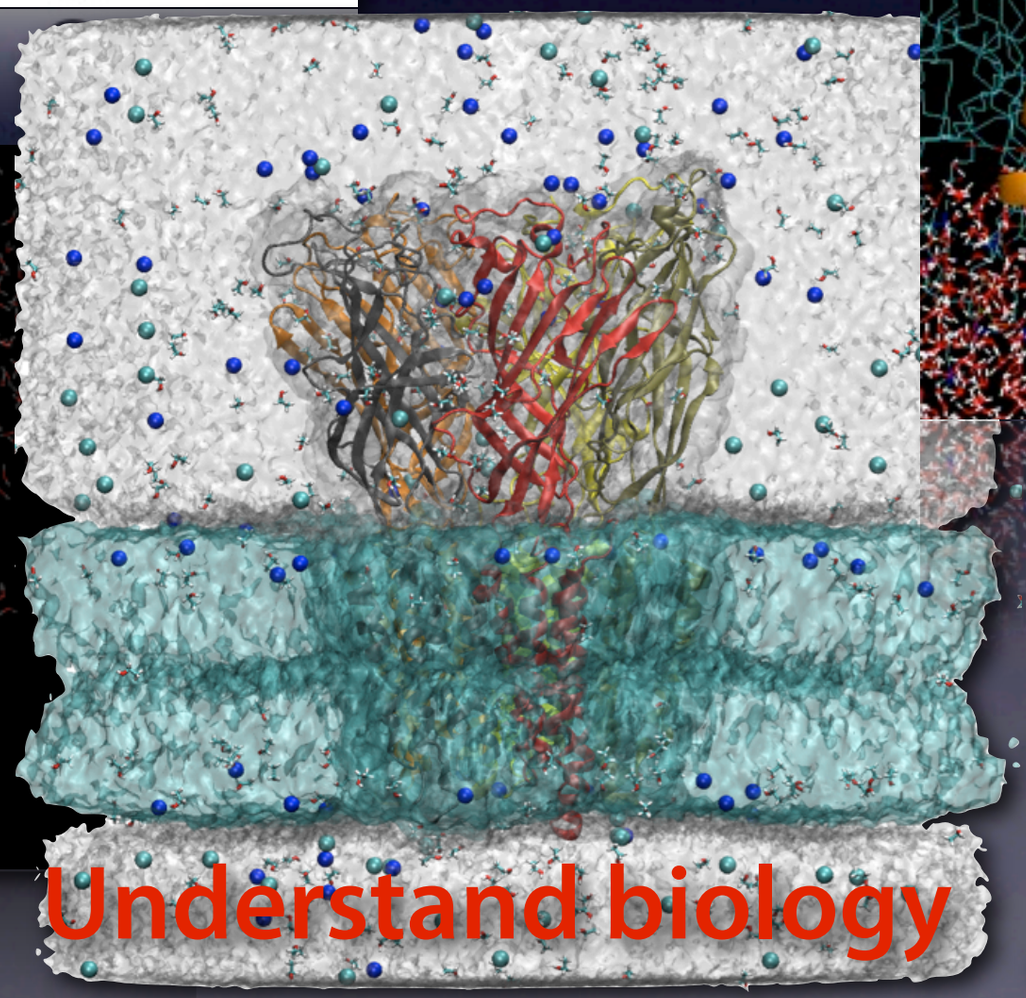








Protein Folding

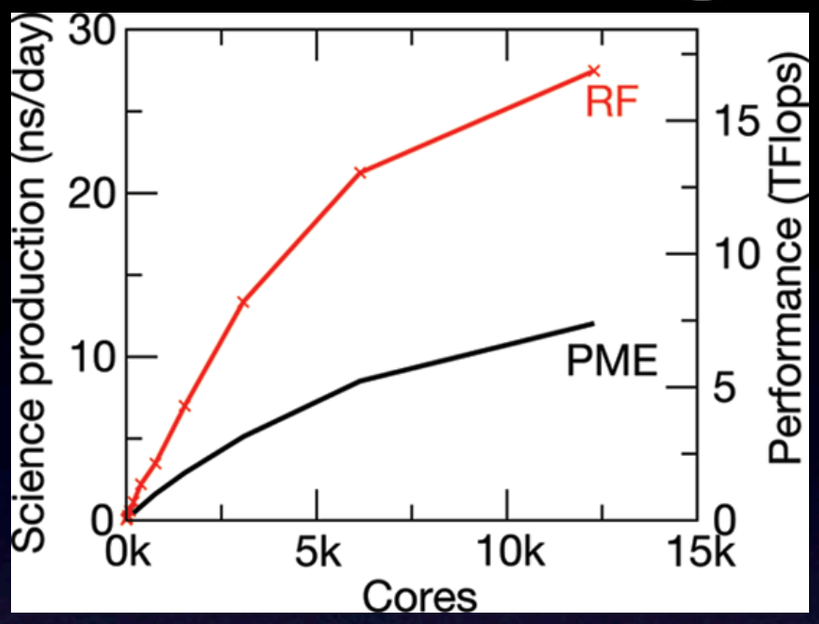


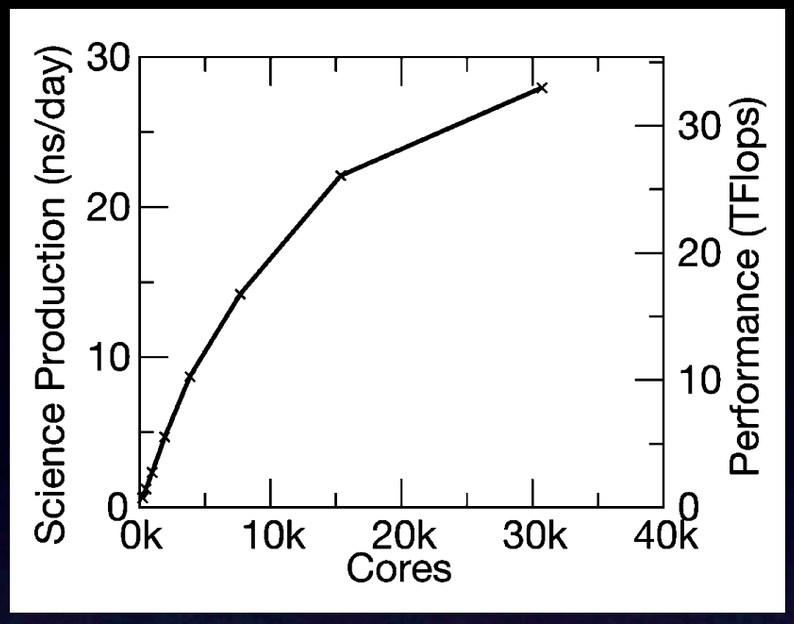
Understand biology

Free Energy &
Drug Design

GROMACS
www.gromacs.org

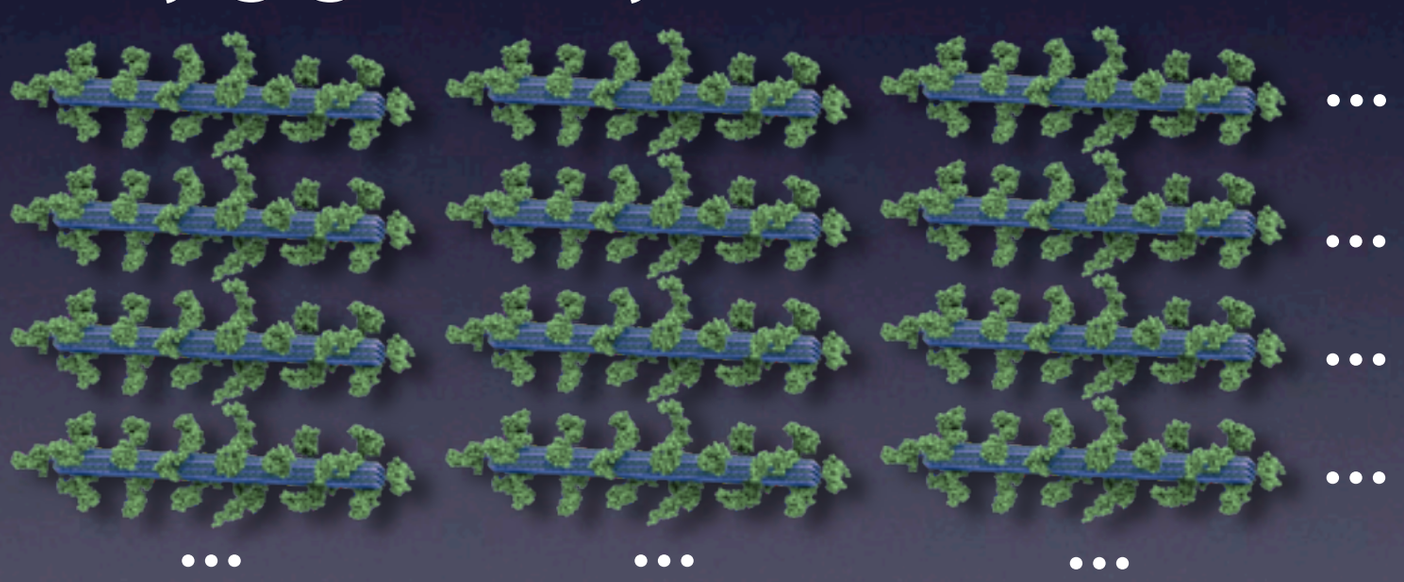
Scaling as an Obsession?



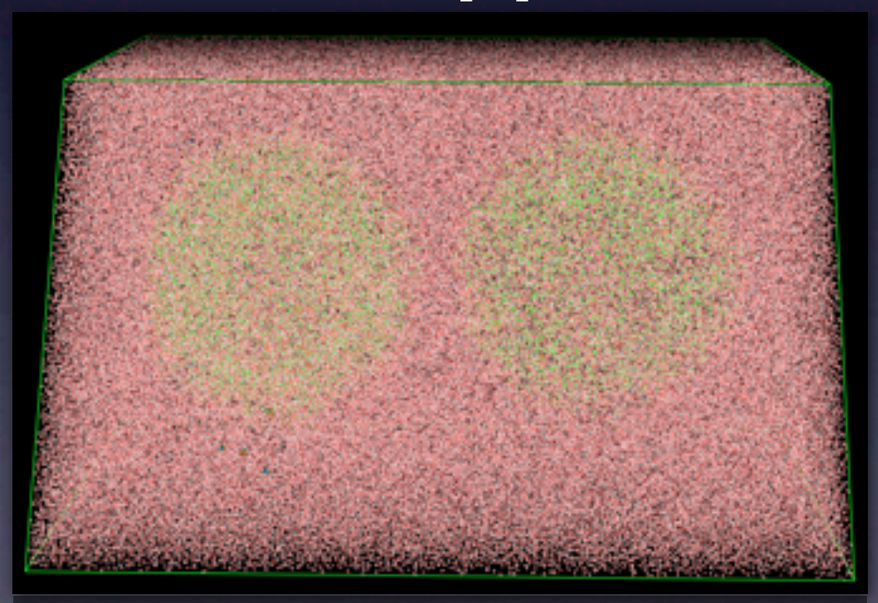


Gromacs has scaled to 150k cores on Jaguar @ ORNL

Only gigantic systems scale - limited number of applications



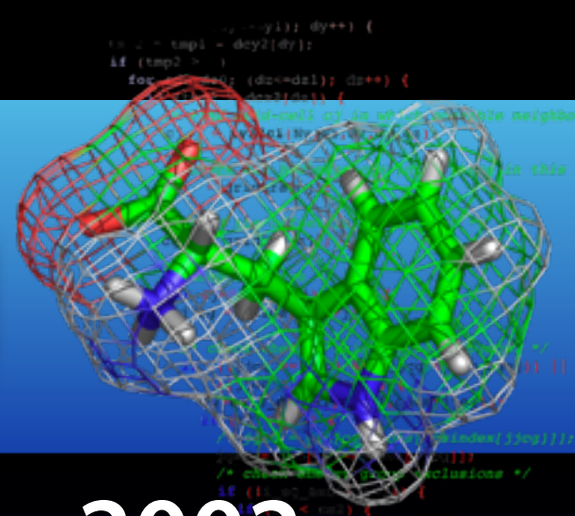
1M-100M atoms



But: <u>Small</u> systems won't scale to large numbers of cores!

How shall we break this impasse?

Stream computing

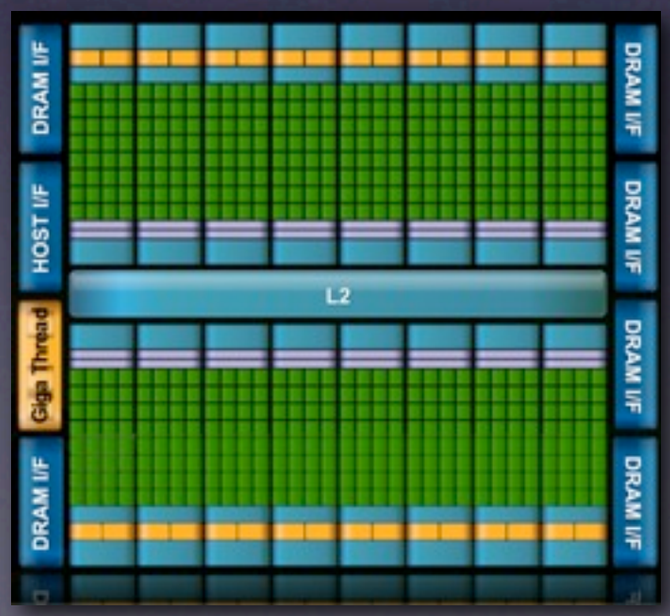


First Gromacs GPU project in 2002 with Ian Buck & Pat Hanrahan, Stanford

Promise of theoretical high FP performance on GeForce4

Severe limitations in practice...

More general than mere hardware acceleration!



Fermi: 448 cores

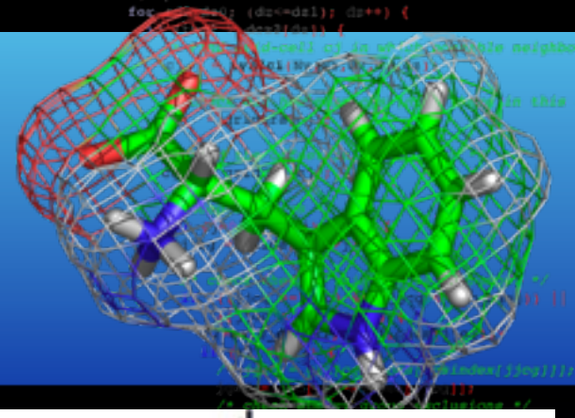
TFLOP-desktops!

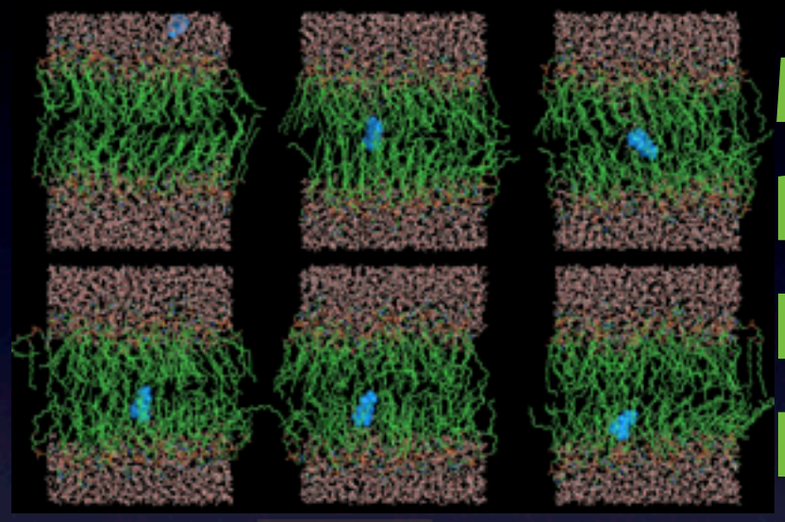
Think in Parallel Problems!

ScalaLife

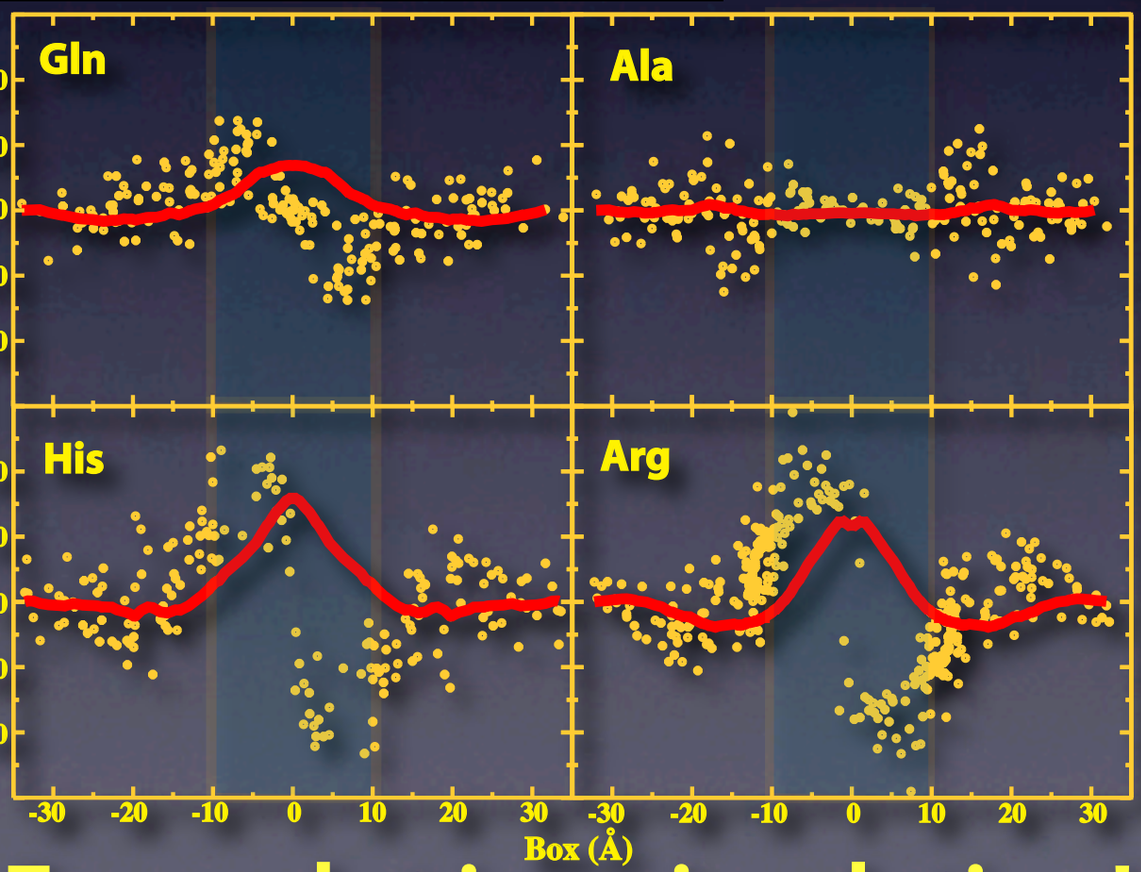
Life Science on next-gen hardware

Ensemble Simulation

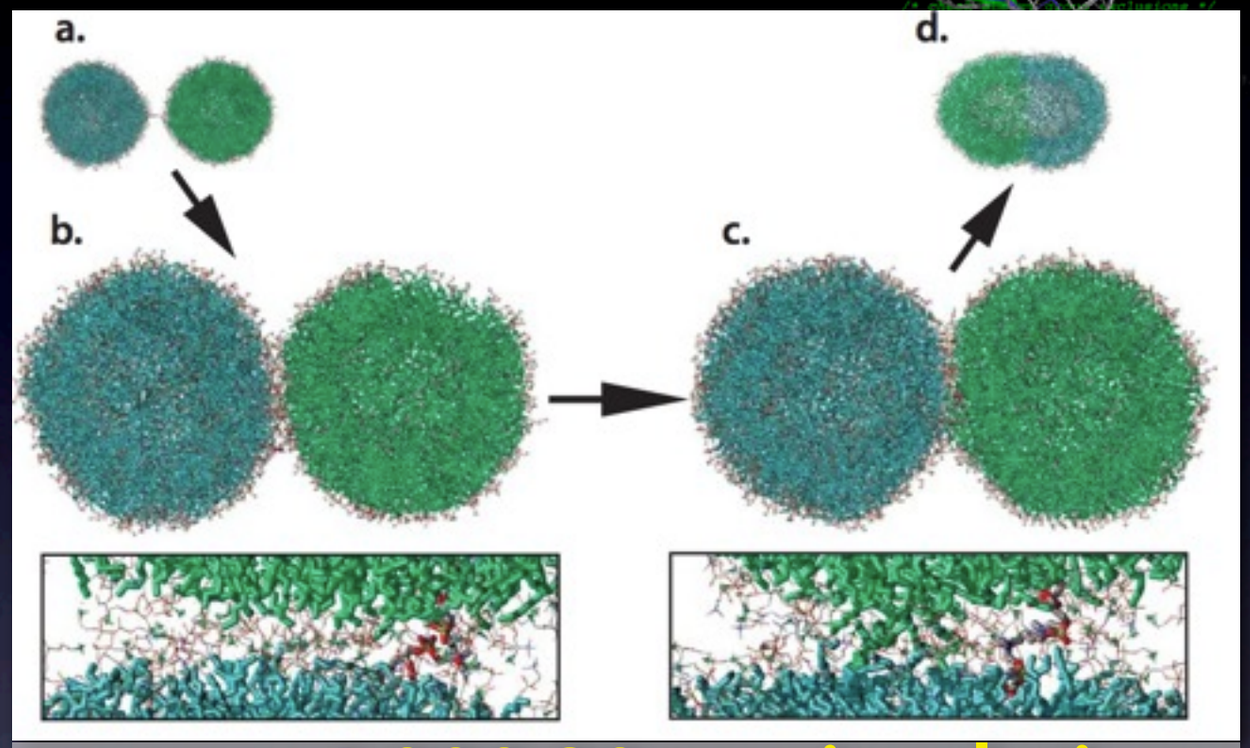




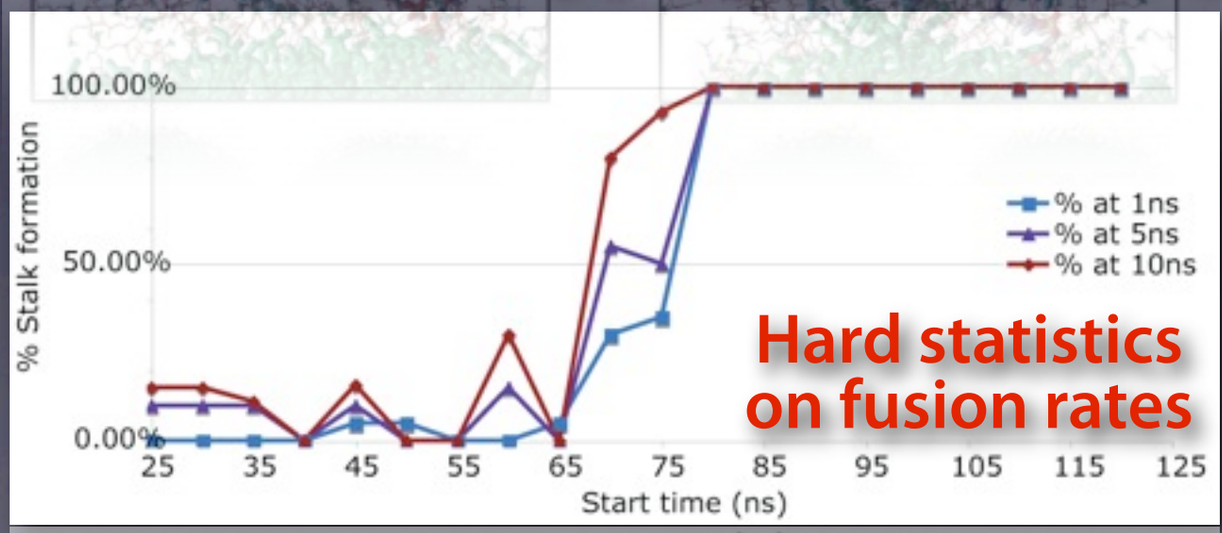
Membrane
Protein
Insertion
Free Energy



Every dot is a simulation!



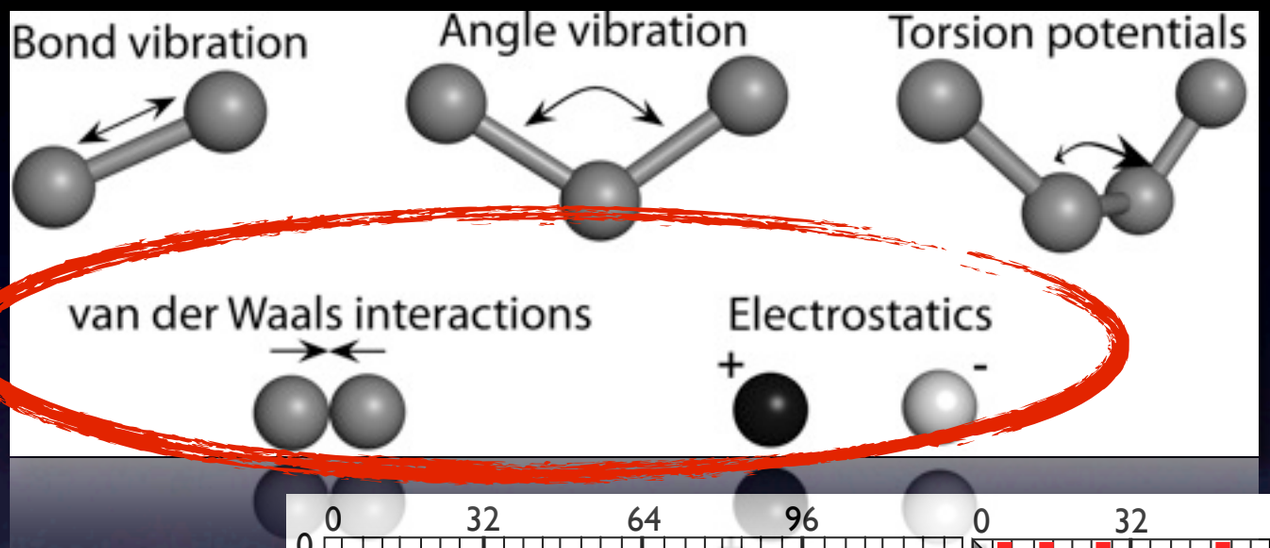
4,000 20-ns simulations



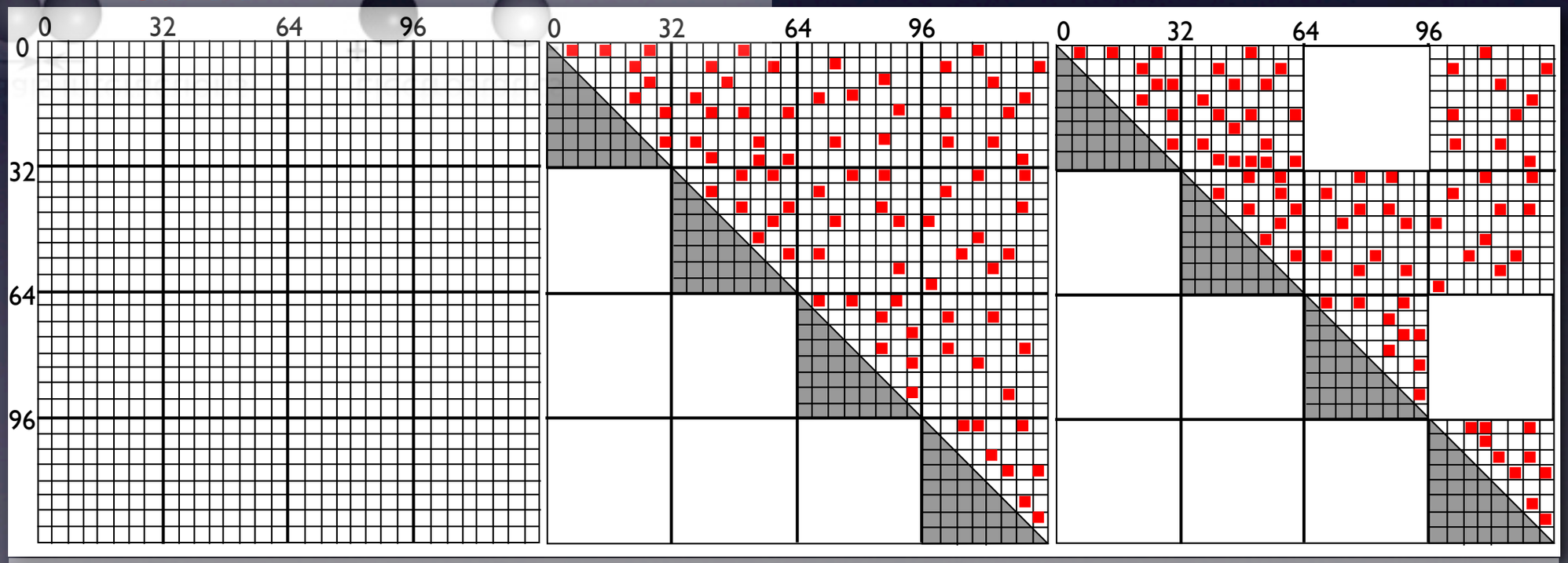
Hard statistics
on fusion rates

Anna Johansson, Peter Kasson

Molecular Dynamics with CUDA



Absolute performance critical, not speedup relative to a reference implementation!

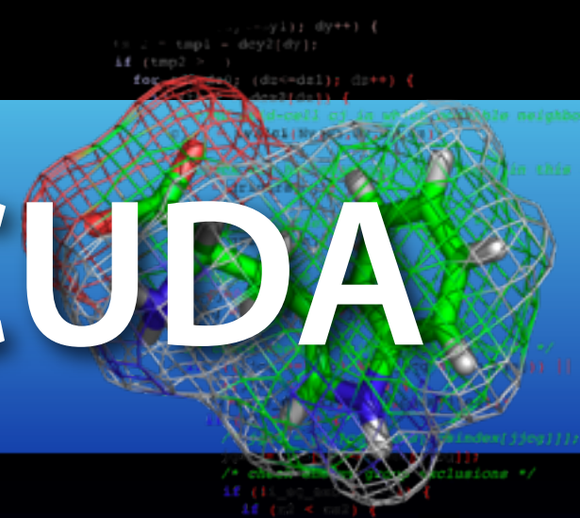


All-vs-all (CUDA book) **Newton's 3rd law** **Sort atoms in tiles**

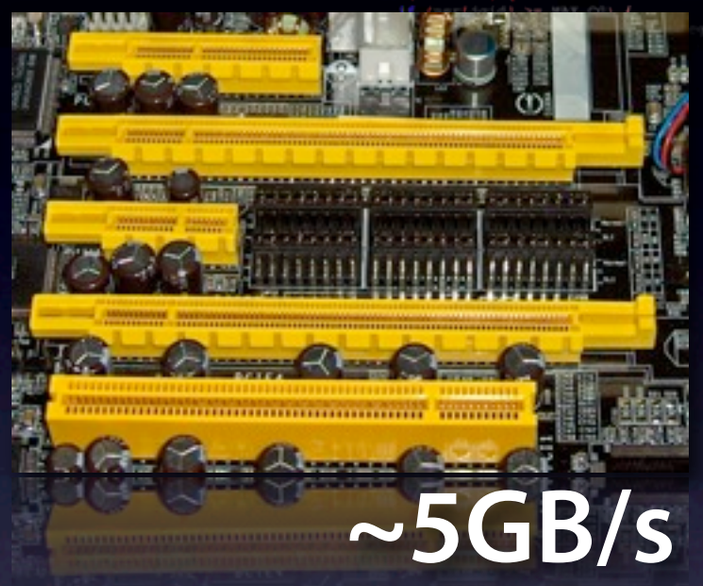
N^2 **(N^2)/2** **N log N**

Scott LeGrand, Peter Eastman

MD Performance in CUDA



Not sufficient to accelerate nonbonded interactions - need to send to/from GPU



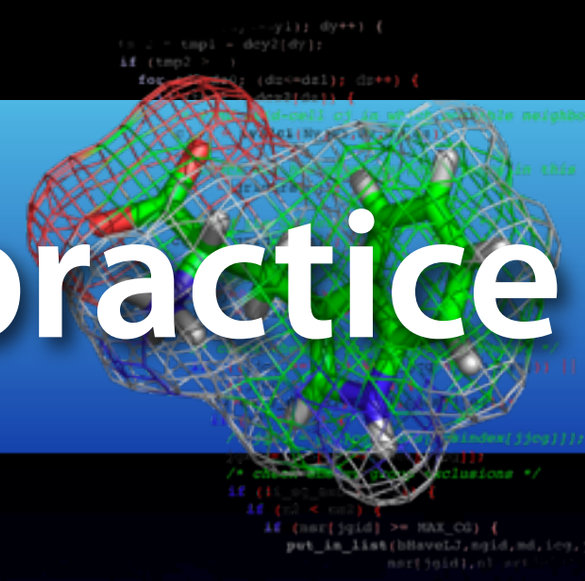
Prefer to do entire simulation on GPU

Don’t rewrite 2M lines-of-code in a separate CUDA-Gromacs...

OpenMM: Core MD functionality in separate library
Stanford, Stockholm, Nvidia & AMD

Fully public API, hardware-agnostic, use anywhere

Gromacs & OpenMM in practice



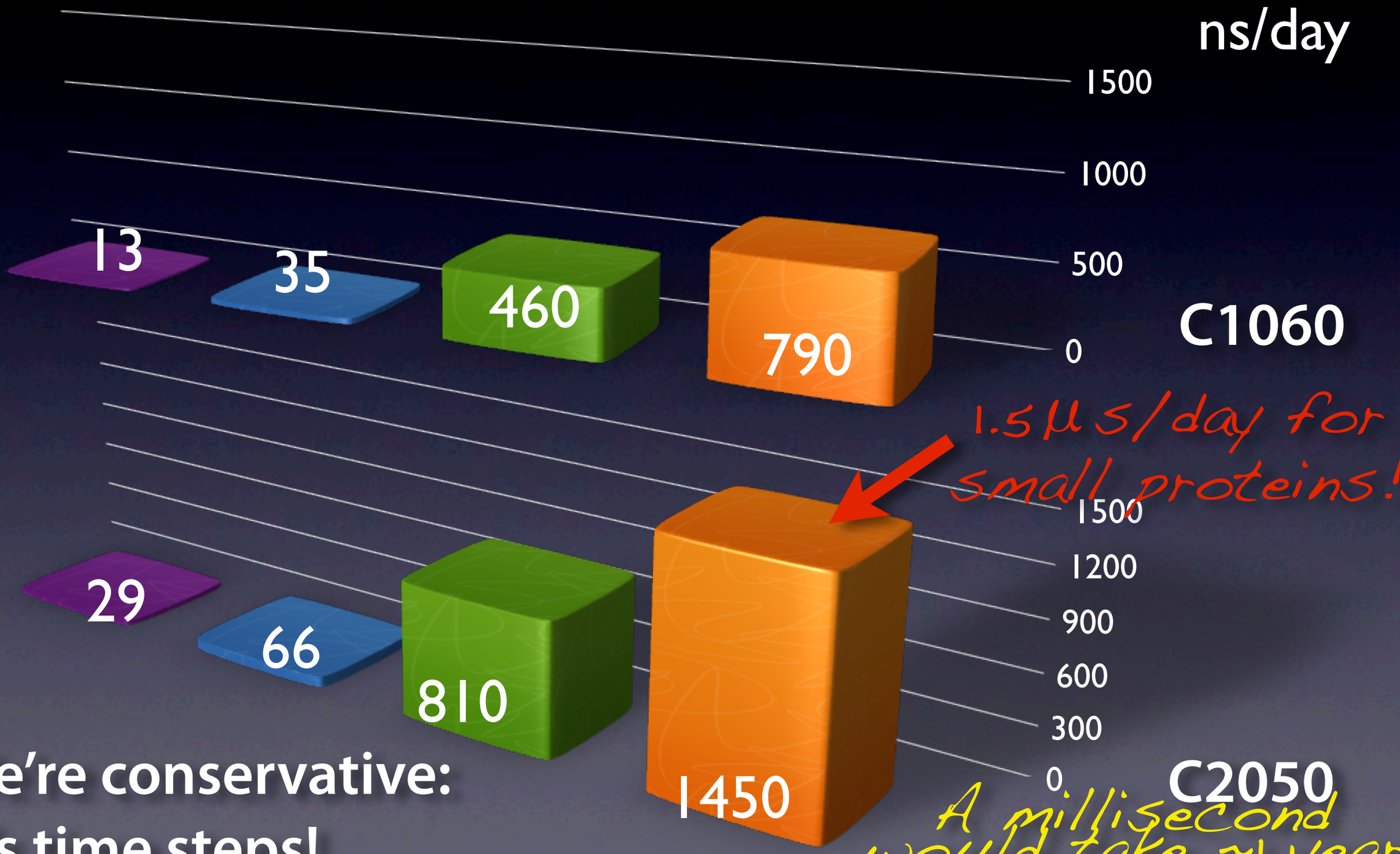
- GPUs supported in Gromacs 4.5
  mdrun ... -device "OpenMM:Cuda"
- Same input files, same output files: "It just works"
- Subset of features work on GPUs for now (checked)
- No shortcuts taken on the GPU:
  - At least same accuracy as on the CPU (<1e-6)
  - Potential energies calculated, free energy works
- Prerelease availability: NOW! www.gromacs.org/gpu

Fermi (C20) performance over C10

BPTI (~21k atoms)

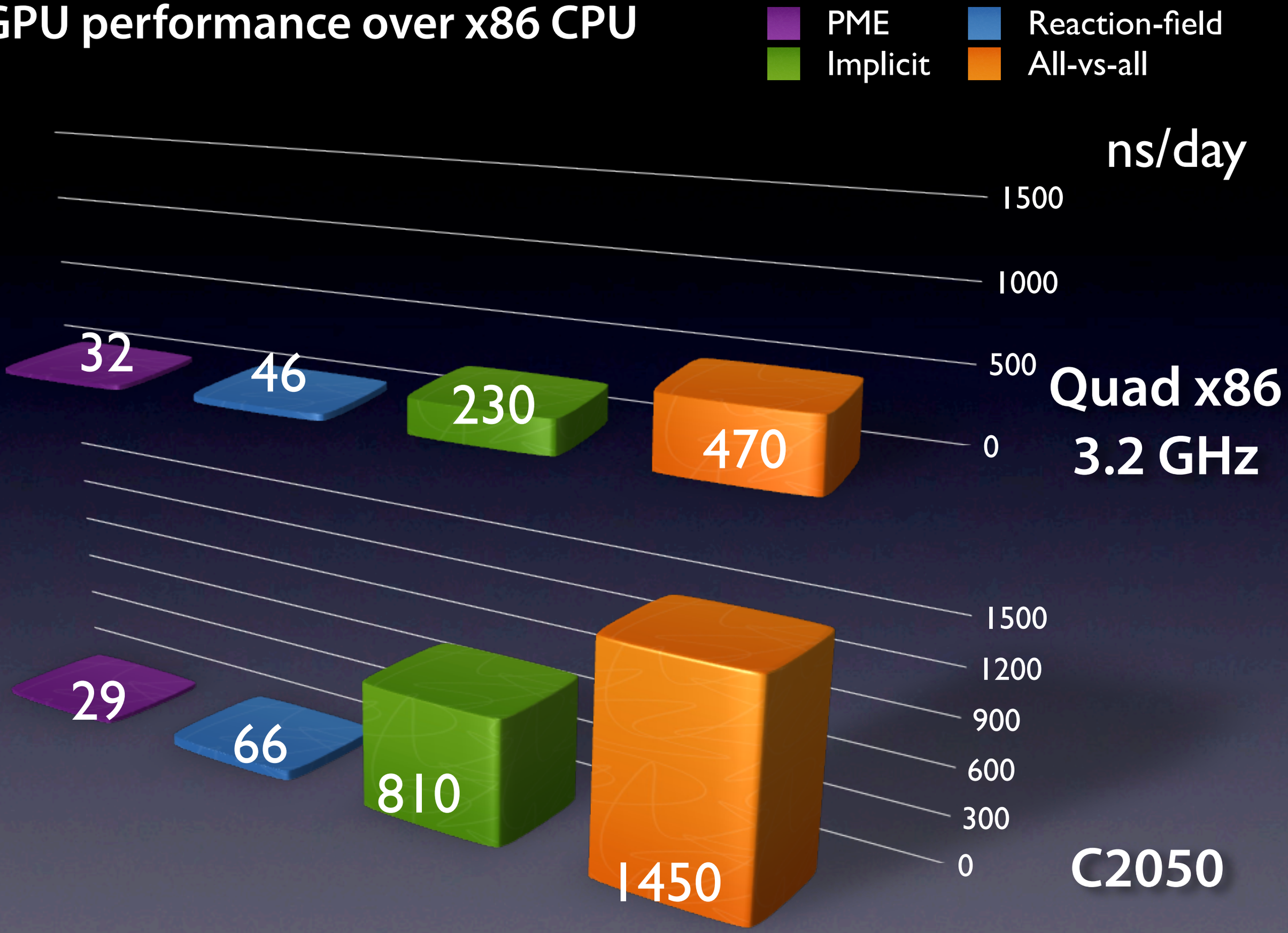
Villin (600 atoms, implicit)

PME
Implicit
Reaction-field
All-vs-all

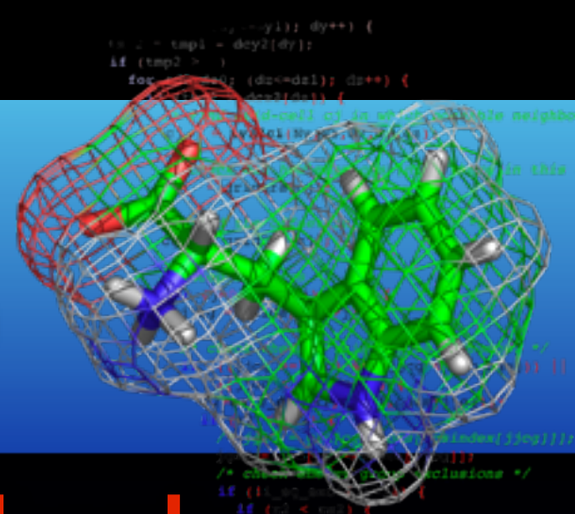


We're conservative: 2fs time steps!

GPU performance over x86 CPU

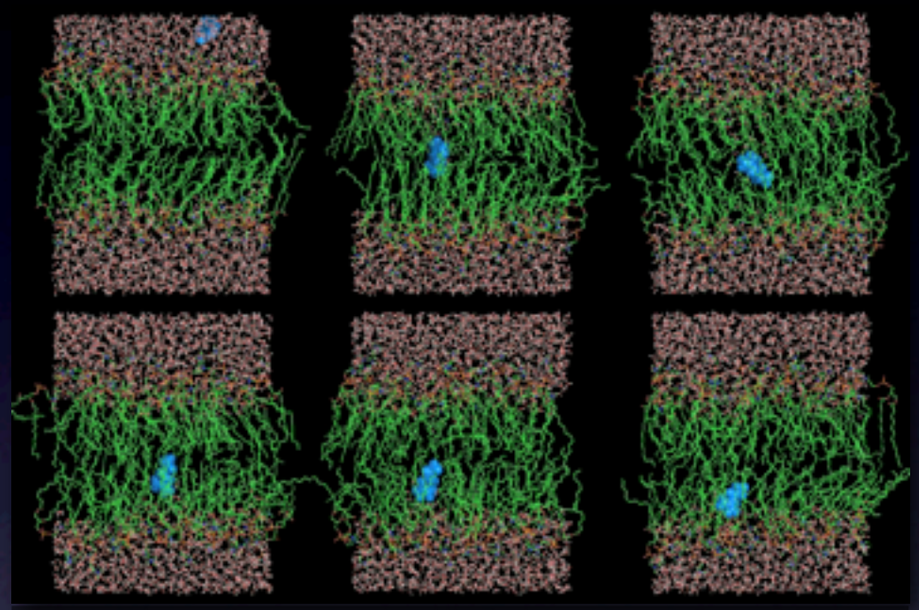


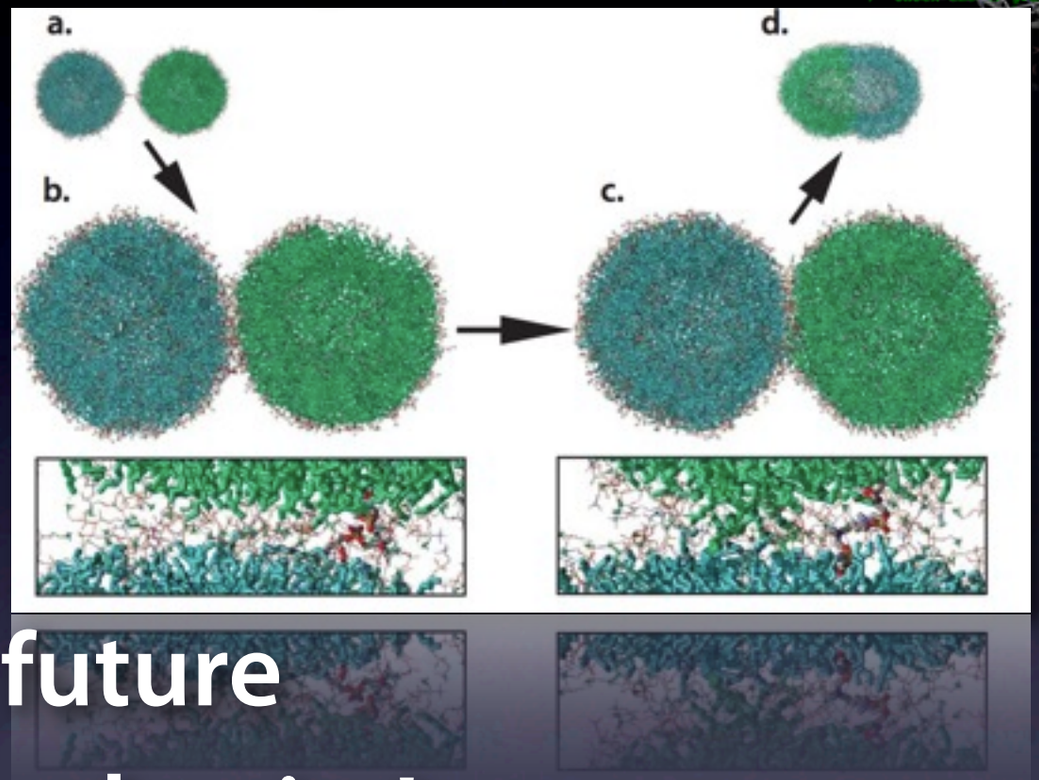
Caveats/Limitations



- Beware of Memory Errors: happens on all hardware
- Gromacs runs tests to check for GPU memory errors
  - Low-end consumer cards can sometimes be bad
- Even fine cards can exhibit random errors
- For production scientific work we strongly recommend Fermi Tesla C20 cards
  - Why? ECC memory! (C2050/C2070)

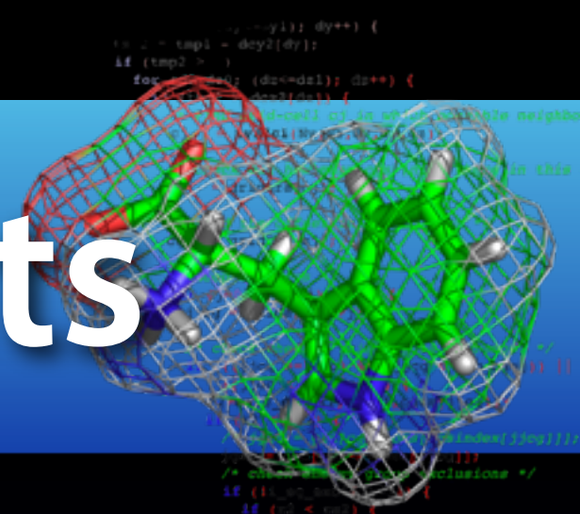
Think Massively Parallel - Scale the Problem, not Runs





- Stream Computing is the future
- Everything is *statistical* mechanics!
- No algorithm will parallelize 5000 degrees of freedom over 1 billion processors
- Parallelize in the problem domain instead
- CUDA provides amazing performance on each node

Acknowledgments



- **GROMACS:** Berk Hess, David van der Spoel, Per Larsson
- **OpenMM:** Rossen Apostolov, Szilard Pall, Peter Eastman, Vijay Pande
- **Nvidia:** Scott LeGrand, Duncan Poole, Andrew Walsh, Chris Butler
- **Ensemble Simulations:** Peter Kasson

erc European Research Council

SJUNDE RAMPROGRAMMET

Vetenskapsrådet

Stiftelsen för Strategisk Forskning

HFSP

NVIDIA

AMD