Using GPUs to Supercharge Visualization and Analysis of Molecular Dynamics Simulations with VMD

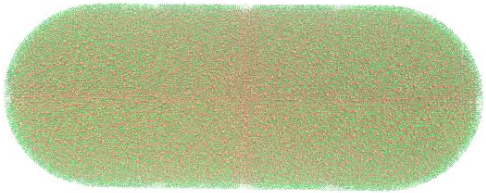
John E. Stone

http://www.ks.uiuc.edu/Research/vmd/

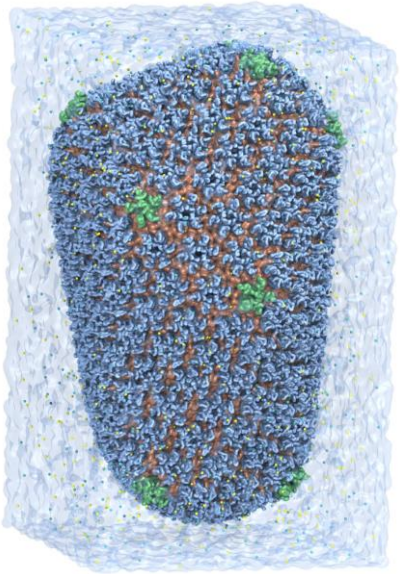
Biomedical Technology Research Center for Macromolecular Modeling and Bioinformatics
Beckman Institute, University of Illinois at Urbana-Champaign - www.ks.uiuc.edu

VMD – “Visual Molecular Dynamics”

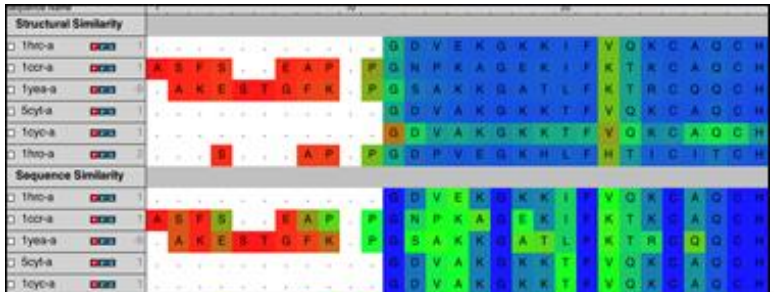
- Visualization and analysis of:
  - molecular dynamics simulations
  - particle systems and whole cells
  - cryoEM densities, volumetric data
  - quantum chemistry calculations
  - sequence information
- User extensible w/ scripting and plugins
- http://www.ks.uiuc.edu/Research/vmd/



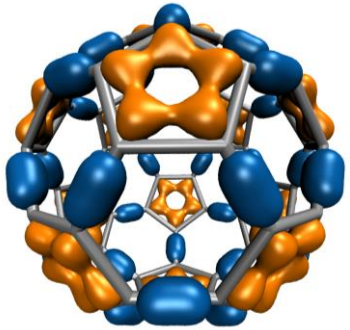
Whole Cell Simulation



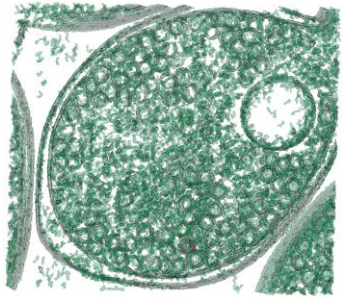
MD Simulations



Sequence Data



Quantum Chemistry

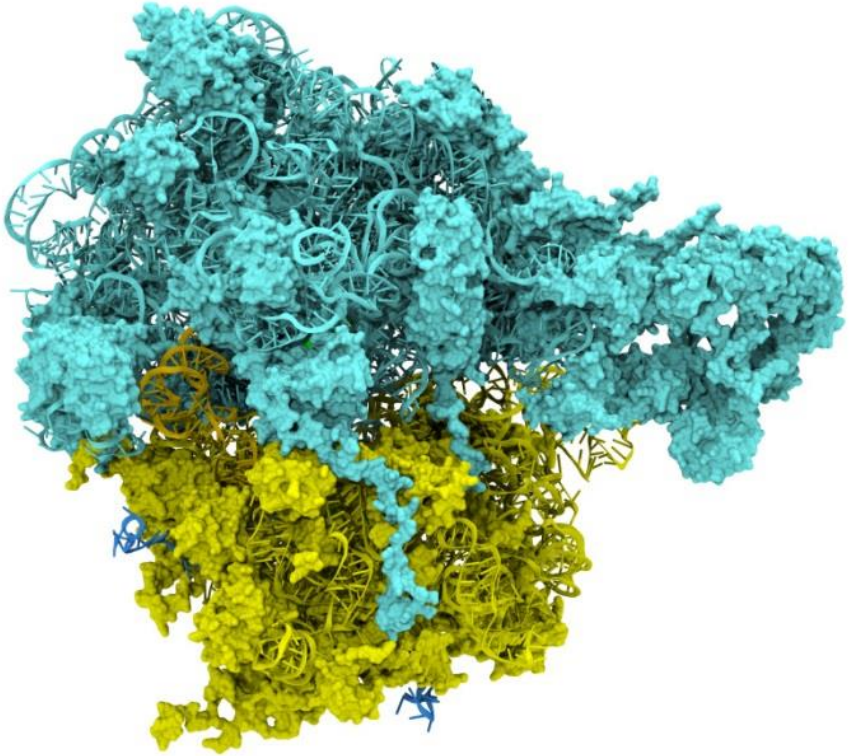


CryoEM, Cellular Tomography

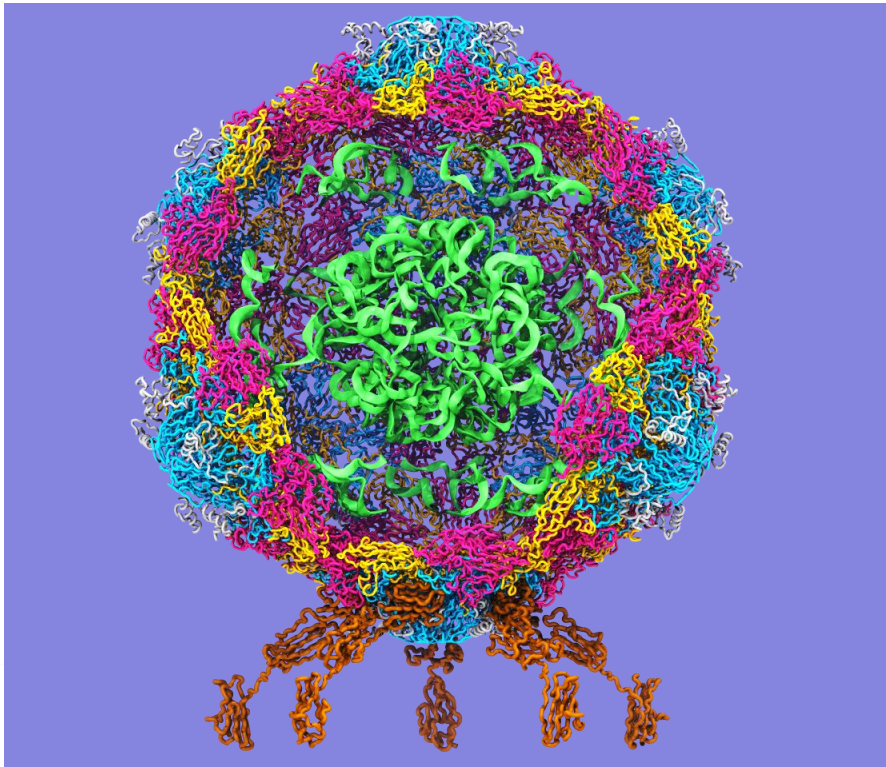
Goal: A Computational Microscope

Study the molecular machines in living cells

Ribosome: target for antibiotics

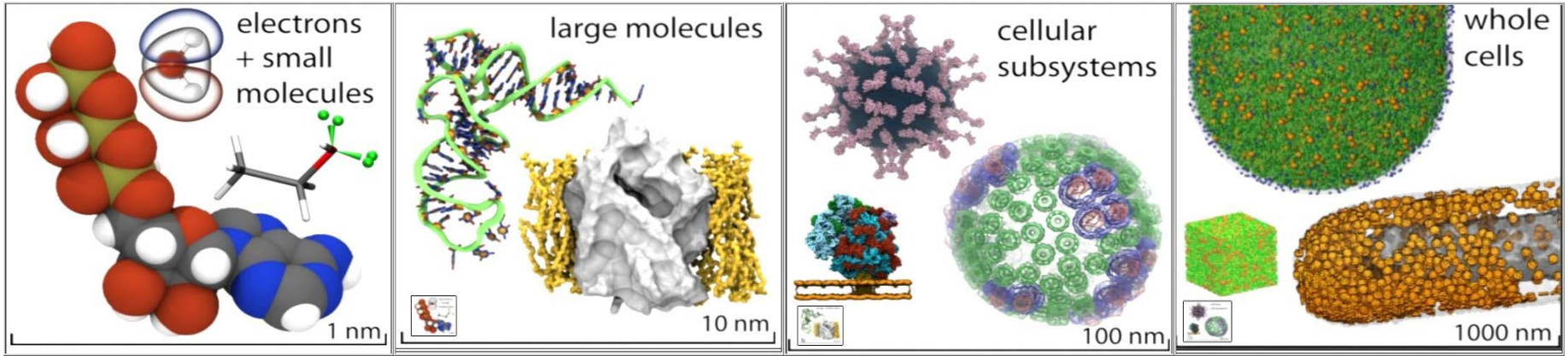


Poliovirus

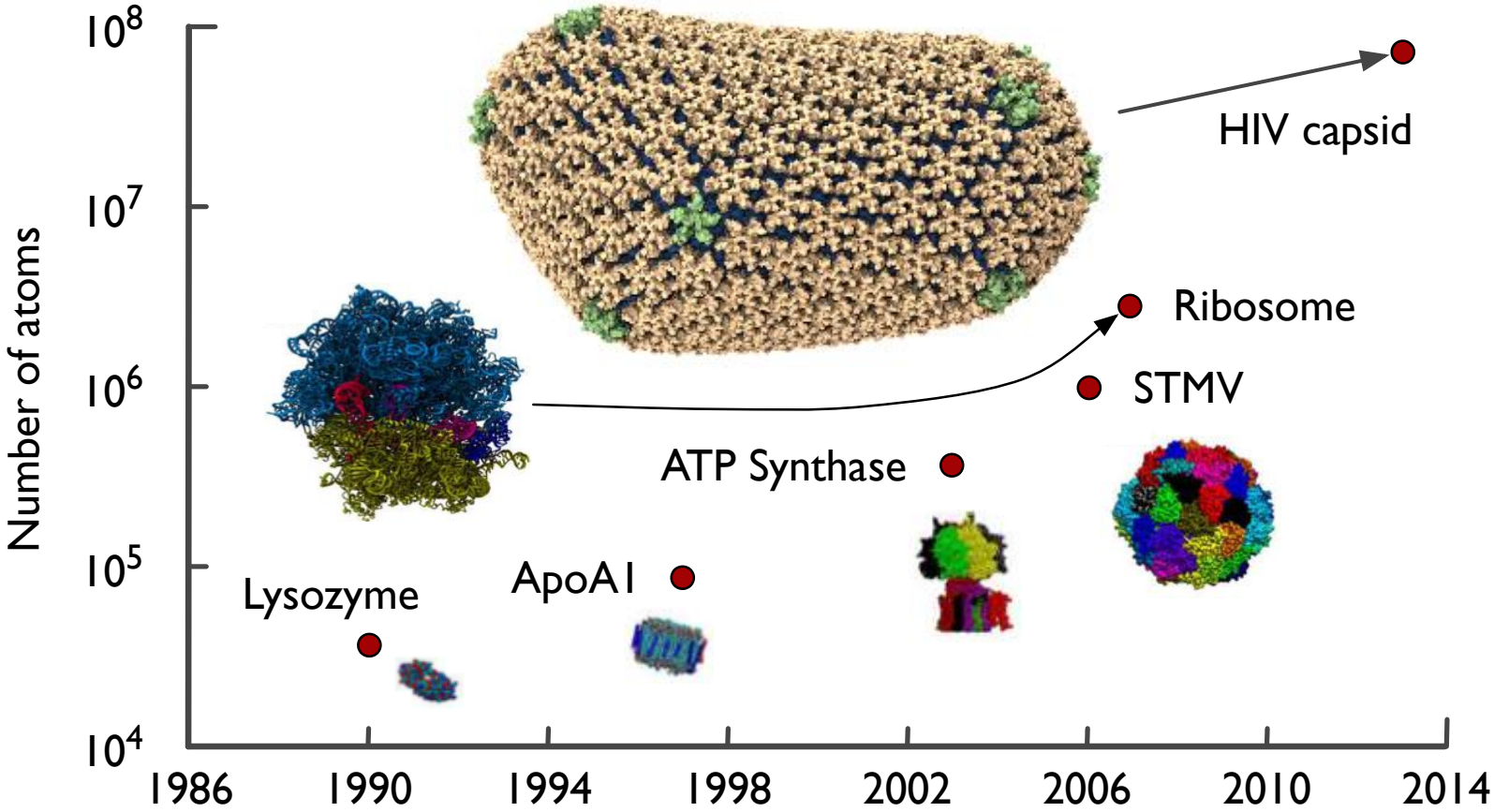


VMD Interoperability Serves Many Communities

- VMD 1.9.1 user statistics:
  - 74,933 unique registered users from all over the world
- Uniquely interoperable with a broad range of tools: AMBER, CHARMM, CPMD, DL_POLY, GAMESS, GROMACS, HOOMD, LAMMPS, NAMD, and many more …
- Supports key data types, file formats, and databases, e.g. electron microscopy, quantum chemistry, MD trajectories, sequence alignments, super resolution light microscopy
- Incorporates tools for simulation preparation, visualization, and analysis



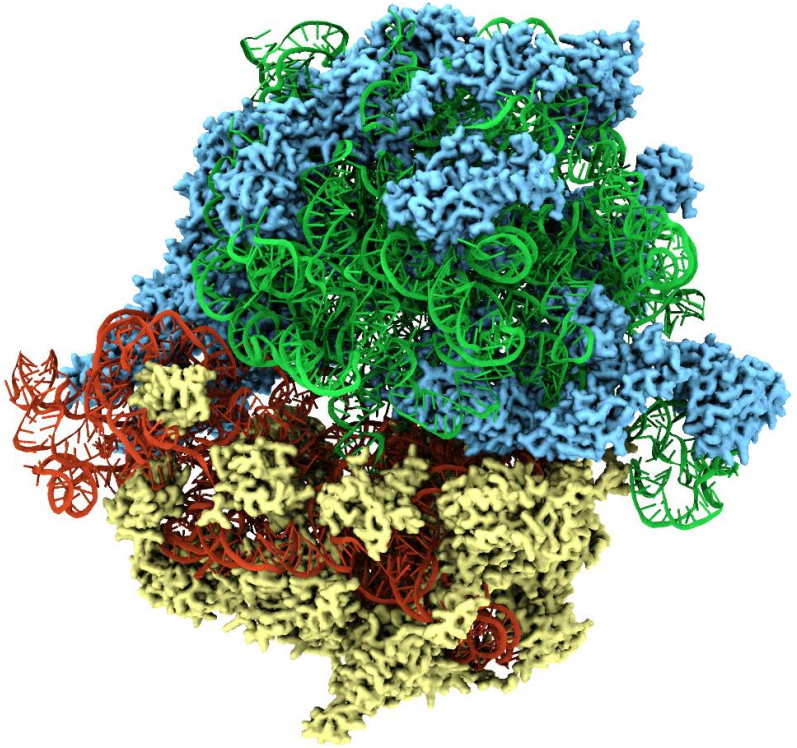
NAMD and VMD Use GPUs and Petascale Computing to Meet Computational Biology's Insatiable Demand for Processing Power



Large-Size and Long-Timescale MD Simulations Drive VMD Development

Extend VMD to enable large state-of-the-art simulations to be performed “routinely”

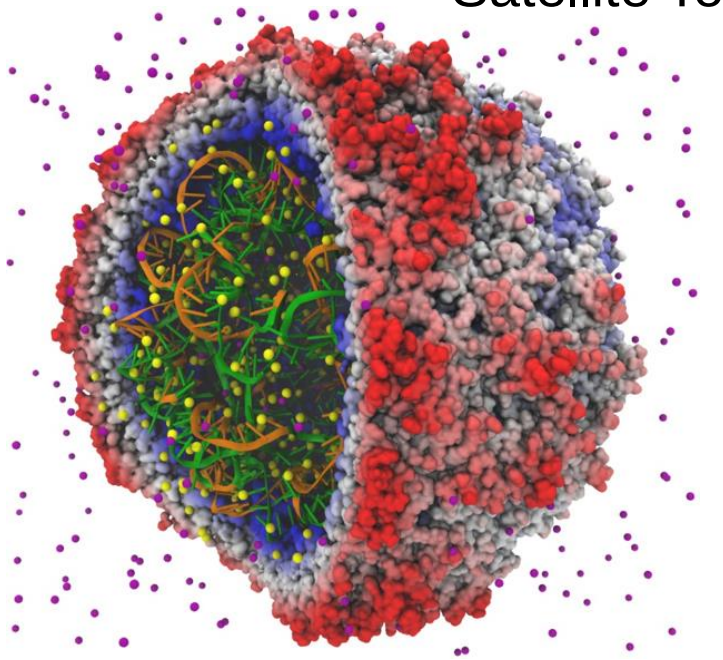
- Improve display fidelity and performance
- Improve model building tools
- Enable flexible and rapid analysis of multi-terabyte simulation trajectories
- Enable development of force field parameters for drug compounds
- Adapt VMD file formats and internal data structures for new simulation types



Ribosome

First Simulation of a Virus Capsid (2006)

Satellite Tobacco Mosaic Virus (STMV)



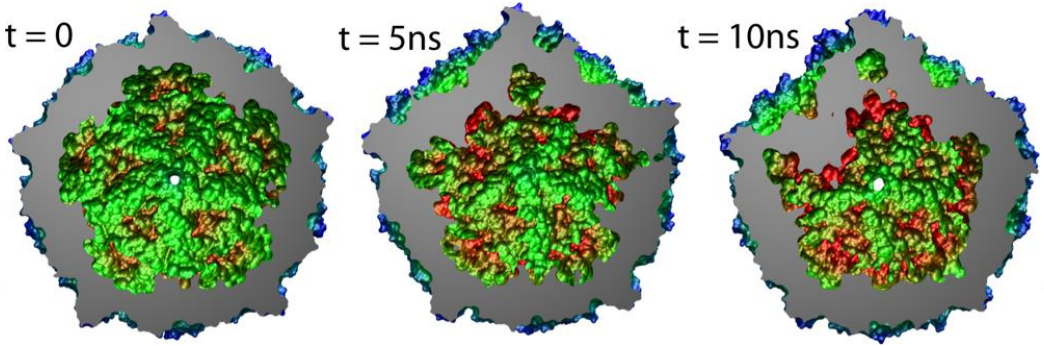
1 million atoms
A huge system for 2006

First MD simulation of a complete virus capsid

STMV smallest available capsid structure

STMV simulation, visualization, and analysis pushed us toward GPU computing!

MD showed that STMV capsid collapses without its RNA core



Freddolino et al., ***Structure***, **14**:437 (2006)

Taking STMV From a “Hero” Simulation to a “Routine” Simulation with GPUs

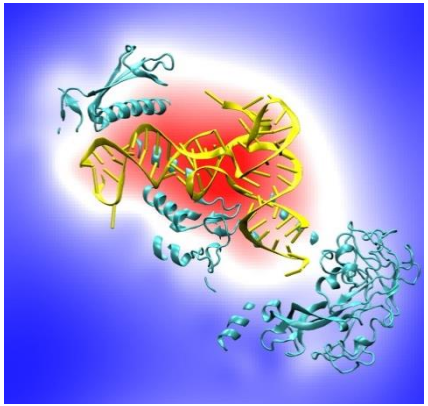
- The STMV project was a turning point
  - Preparing STMV models and placing ions tremendously demanding computational task
  - Existing approaches to visualizing and analyzing the simulation began to break down
- It was already clear in 2006 that the study of viruses relevant to human health would require a long-term investment in **better parallel algorithms and extensive use of acceleration technologies in NAMD and VMD**
- **These difficulties led us to accelerate key modeling tasks with GPUs**

VMD Electrostatics: First Use of CUDA

- **Spring 2007: CUDA v0.7**
- Electrostatic potential maps evaluated on 3-D lattice:

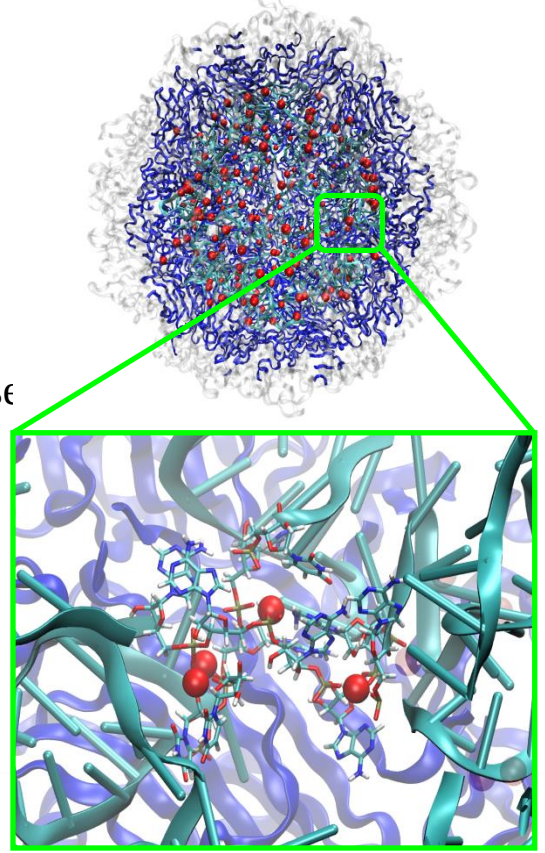
$$V_i = \sum_j \frac{q_j}{4\pi\epsilon_0 |\mathbf{r}_j - \mathbf{r}_i|}$$

- Applications include:
  - Ion placement for structure building
  - Visualization
  - Trajectory analysis
  - **Speedups up to 11x vs. 4-core CPUs**



Isoleucine tRNA synthetase

STMV Ion Placement



Accelerating Molecular Modeling Applications with Graphics Processors. Stone et al., J. Computational Chemistry, 28:2618-2640, 2007.

Multilevel summation of electrostatic potentials using graphics processing units. Hardy, et al. *J. Parallel Computing*, 35:164-177, 2009.

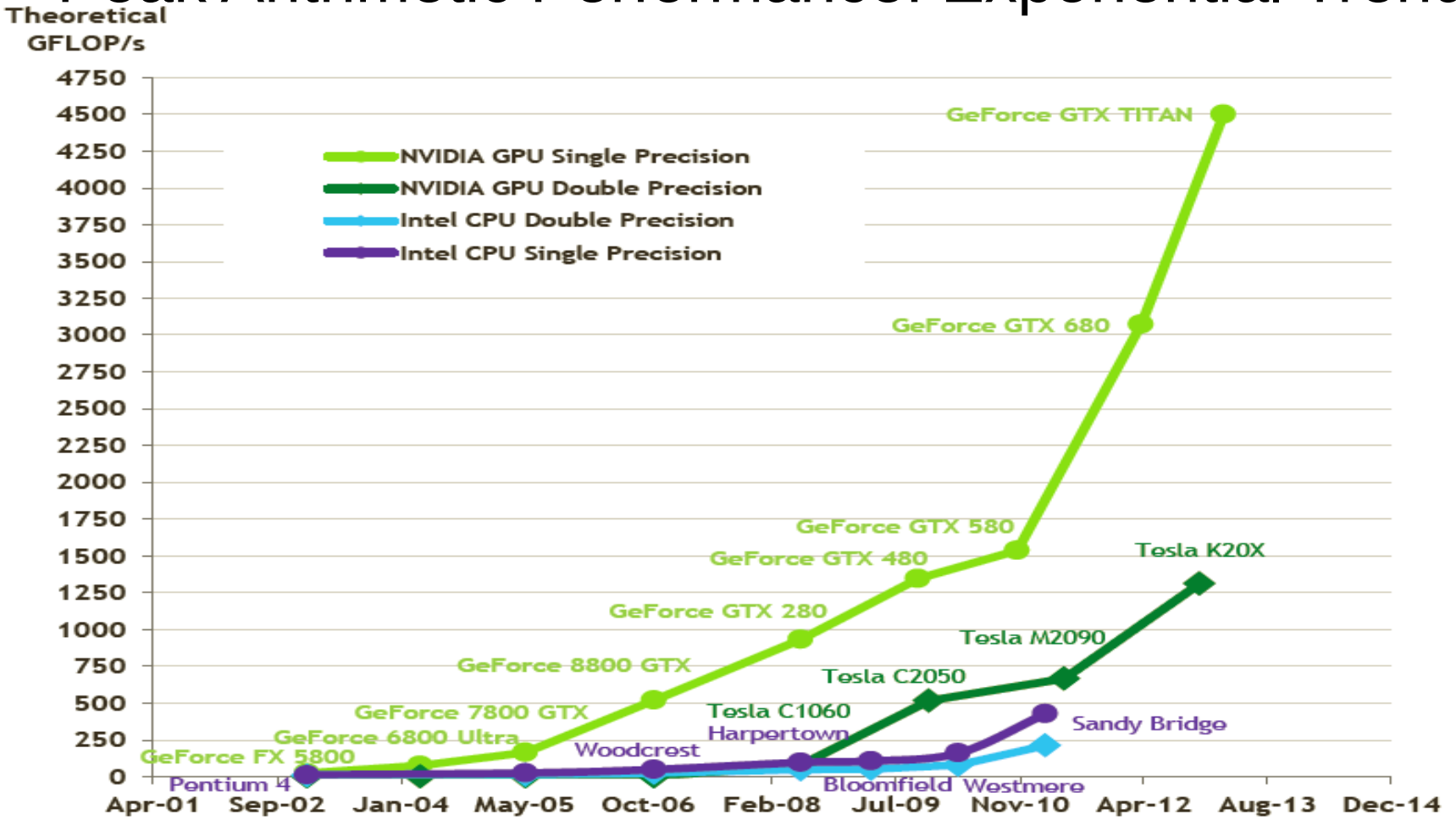
GPU Computing

- Commodity devices, omnipresent in modern computers (over a **million** sold per **week**)
- Massively parallel hardware, hundreds of processing units, **throughput oriented architecture**
- Standard integer and floating point types supported
- Programming tools allow software to be written in dialects of familiar C/C++ and integrated into legacy software
- GPU algorithms are often multicore friendly due to attention paid to **data locality** and **data-parallel** work decomposition

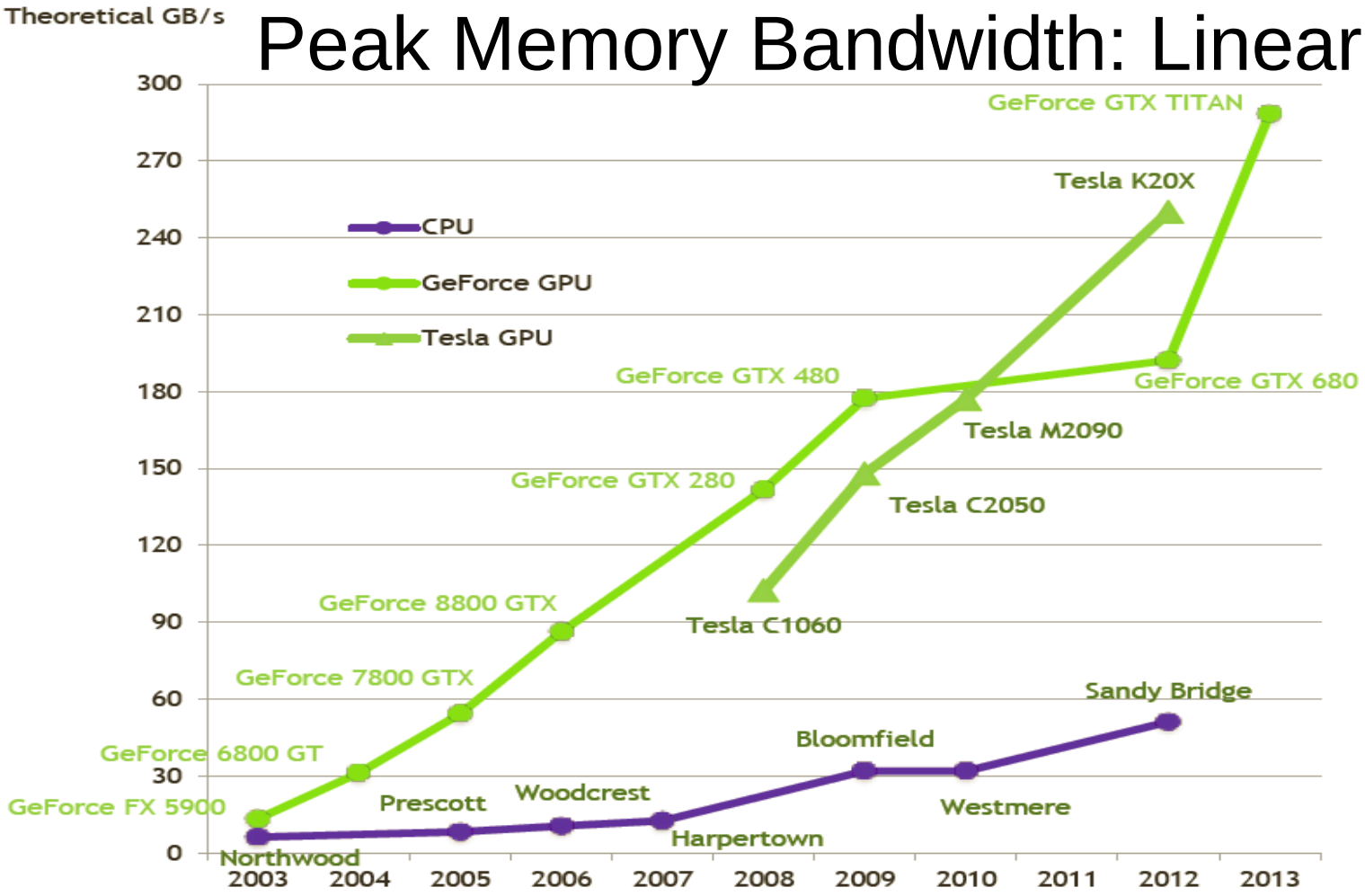
What Speedups Can GPUs Achieve?

- Single-GPU speedups of **2.5x** to **7x** vs. **4-core CPU** core are common
- Best speedups can reach **25x** or more, attained on codes dominated by floating point arithmetic, especially native GPU machine instructions, e.g. expf(), rsqrtf(), …
- **Amdahl's Law** can prevent legacy codes from achieving peak speedups with shallow GPU acceleration efforts

Peak Arithmetic Performance: Exponential Trend

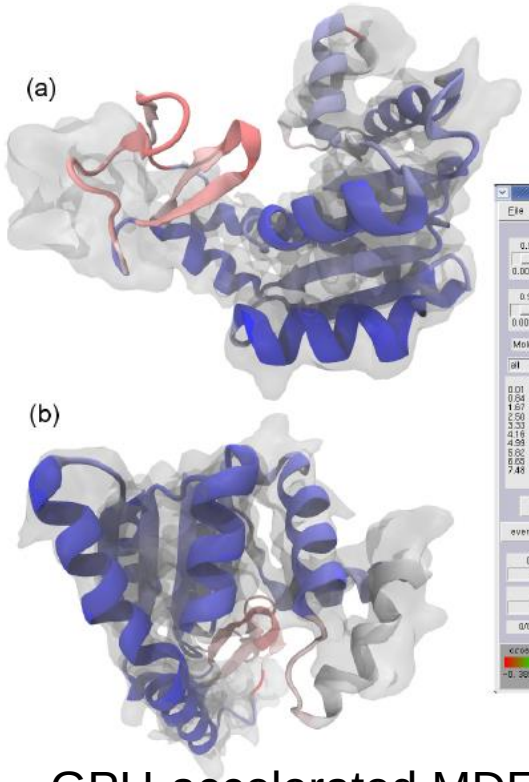


Peak Memory Bandwidth: Linear Trend

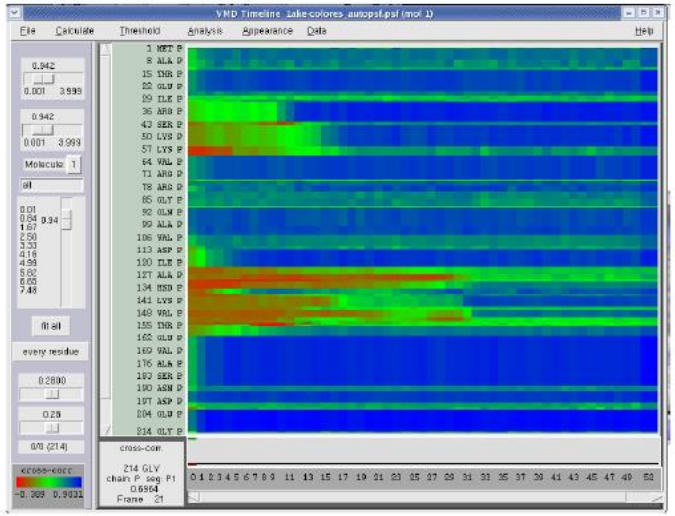


GPUs Can Reduce Trajectory Analysis Runtimes from Hours to Minutes

Enable laptops and desktop workstations to handle tasks that would have previously required a cluster, or a very long wait…



(c)



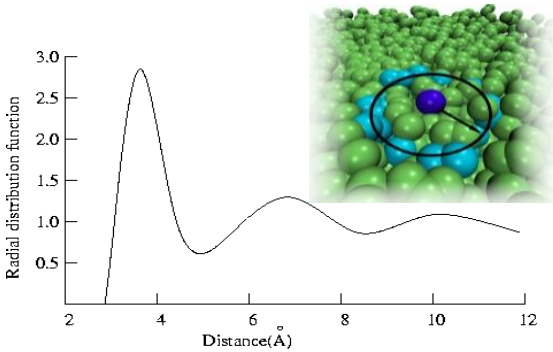
GPU-accelerated MDFF Cross Correlation Timeline

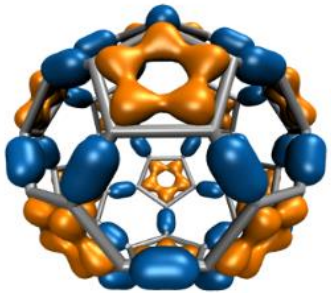
Example VMD Startup Messages with GPU Accelerators:

```
Info) VMD for LINUXAMD64, version 1.9.2a36 (December 12, 2013)
Info) http://www.ks.uiuc.edu/Research/vmd/
Info) Email questions and bug reports to vmd@ks.uiuc.edu
Info) Please include this reference in published work using VMD:
Info)    Humphrey, W., Dalke, A. and Schulten, K., `VMD - Visual
Info)    Molecular Dynamics', J. Molec. Graphics 1996, 14.1, 33-38.
Info) ------------------------------------------------------------
Info) Multithreading available, 16 CPUs detected.
Info) Free system memory: 58392MB (96%)
Info) Creating CUDA device pool and initializing hardware...
Info) Detected 4 available CUDA accelerators:
Info) [0] Tesla K20c         13 SM_3.5 @ 0.71 GHz, 5.0GB RAM, OIO, ZCP
Info) [1] Tesla K20c         13 SM_3.5 @ 0.71 GHz, 5.0GB RAM, OIO, ZCP
Info) [2] Quadro 7000          16 SM_2.0 @ 1.30 GHz, 6.0GB RAM, KTO, OIO, ZCP
Info) [3] Quadro 7000          16 SM_2.0 @ 1.30 GHz, 6.0GB RAM, KTO, OIO, ZCP
Info) Dynamically loaded 2 plugins in directory:
Info) /Projects/vmd/pub/linux64/lib/vmdtest/plugins/LINUXAMD64/molfile
```

CUDA GPU-Accelerated Trajectory Analysis and Visualization in VMD

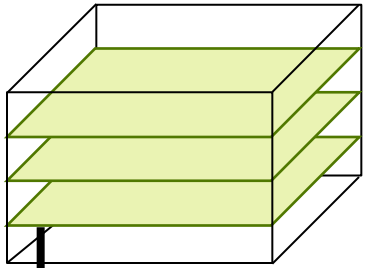
VMD GPU-Accelerated Feature or GPU Kernel	Exemplary speedup vs. contemporary 4-core CPU
Molecular orbital display	**30x**
Radial distribution function	**23x**
Molecular surface display	**15x**
Electrostatic field calculation	**11x**
Ray tracing w/ shadows, AO lighting	**7x**
cryoEM cross correlation quality-of-fit	**7x**
Ion placement	6x
MDFF density map synthesis	6x
Implicit ligand sampling	6x
Root mean squared fluctuation	6x
Radius of gyration	5x
Close contact determination	5x
Dipole moment calculation	4x





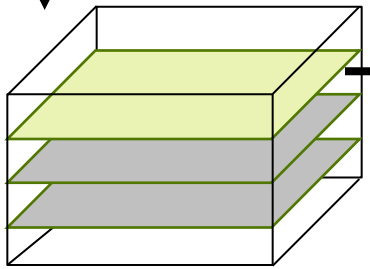
GPU-Accelerated C_{60} Molecular Orbitals

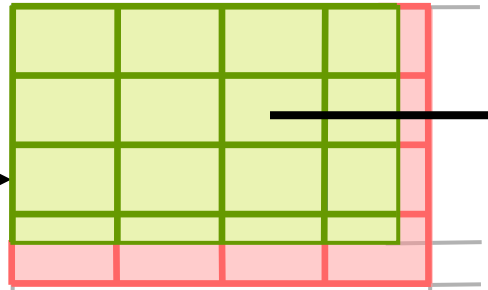
3-D orbital lattice: millions of points



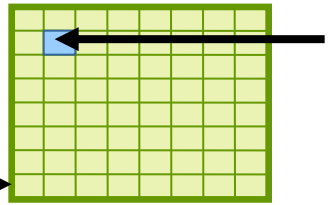
Device	CPUs, GPUs	Runtime (s)	Speedup
2x Intel X5550-SSE	8	4.13	1
GeForce GTX 480	1	0.255	16
GeForce GTX 480	4	0.081	51

Lattice slices computed on multiple GPUs



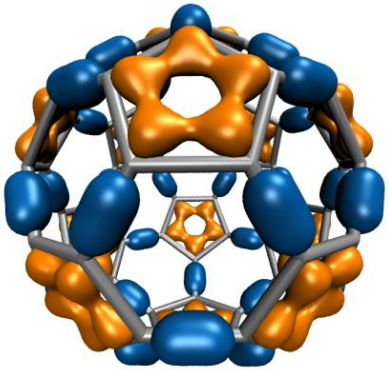


2-D CUDA grid on each GPU



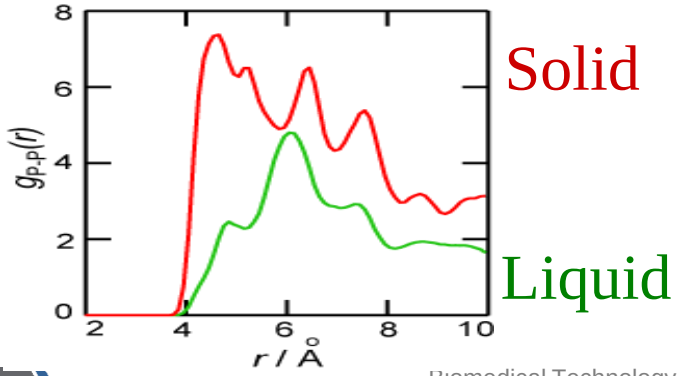
CUDA thread blocks

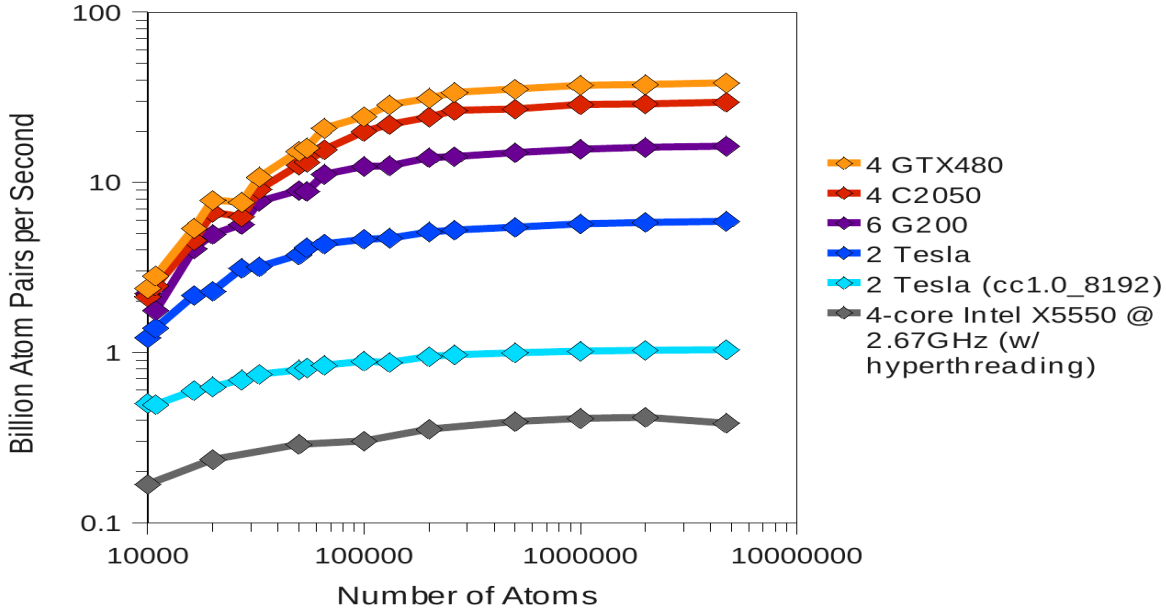
GPU threads compute one point



Multi-GPU RDF Performance

- 4 NVIDIA GTX480 GPUs 30 to 92x faster than 4-core Intel X5550 CPU
- Fermi GPUs ~3x faster than GT200 GPUs: larger on-chip shared memory

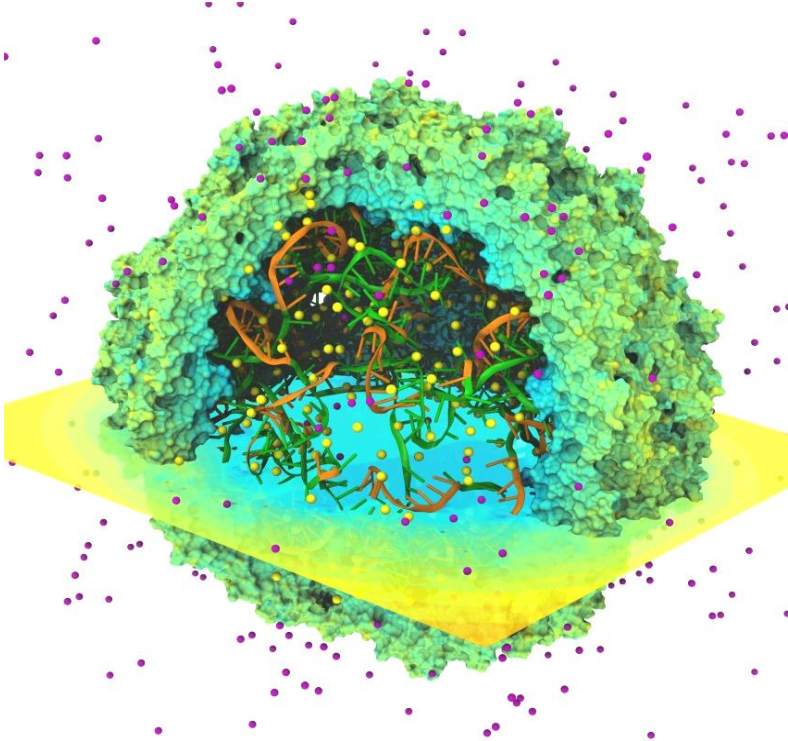




Fast Analysis of Molecular Dynamics Trajectories with Graphics Processing Units – Radial Distribution Functions.
Levine, et al., *J. Comp. Physics*, 230(9):3556-3569, 2011.

Time-Averaged Electrostatics Analysis on Energy-Efficient GPU Cluster

- **1.5 hour** job (CPUs) reduced to **3 min** (CPUs+GPU)
- Electrostatics of thousands of trajectory frames averaged
- Per-node power consumption on NCSA “AC” GPU cluster:
  - CPUs-only: 448 Watt-hours
  - CPUs+GPUs: 43 Watt-hours
- GPU Speedup: **25.5x**
- Power efficiency gain: **10.5x**



Quantifying the Impact of GPUs on Performance and Energy Efficiency in HPC Clusters.
Enos, et al. *The Work in Progress in Green Computing,* pp. 317-324, 2010.

Time-Averaged Electrostatics Analysis on Blue Waters Cray XK6/XK7

NCSA Blue Waters Node Type	**Seconds per trajectory frame for one compute node**
Cray XE6 Compute Node: 32 CPU cores (2xAMD 6200 CPUs)	9.33
Cray XK6 GPU-accelerated Compute Node: 16 CPU cores + **NVIDIA X2090 (Fermi) GPU**	2.25
Speedup for GPU XK6 nodes vs. CPU XE6 nodes	**XK6 nodes are 4.15x faster overall**
XK7 tests indicate MSM is CPU-bound with the Kepler K20X GPU**.** **Performance is not much faster (yet) than Fermi X2090:** **Move spatial hashing, prolongation, interpolation onto the GPU…**	**XK7 nodes 4.3x faster overall** **In progress….**

Performance for VMD time-averaged electrostatics w/ Multilevel Summation Method on the NCSA Blue Waters Early Science System

Getting Past the “Chicken and the Egg”

- GPU clusters still rare circa 2009-2011, most were not quite big enough to be used for large scale production science yet … but the potential was definitely there
- **Performance and power efficiency** benefits were seen for VMD and NAMD, on ever larger node counts
- Larger GPU accelerated systems were on the horizon

GPU Clusters for High Performance Computing. Kindratenko et al., IEEE Cluster’09, pp. 1-8, 2009.

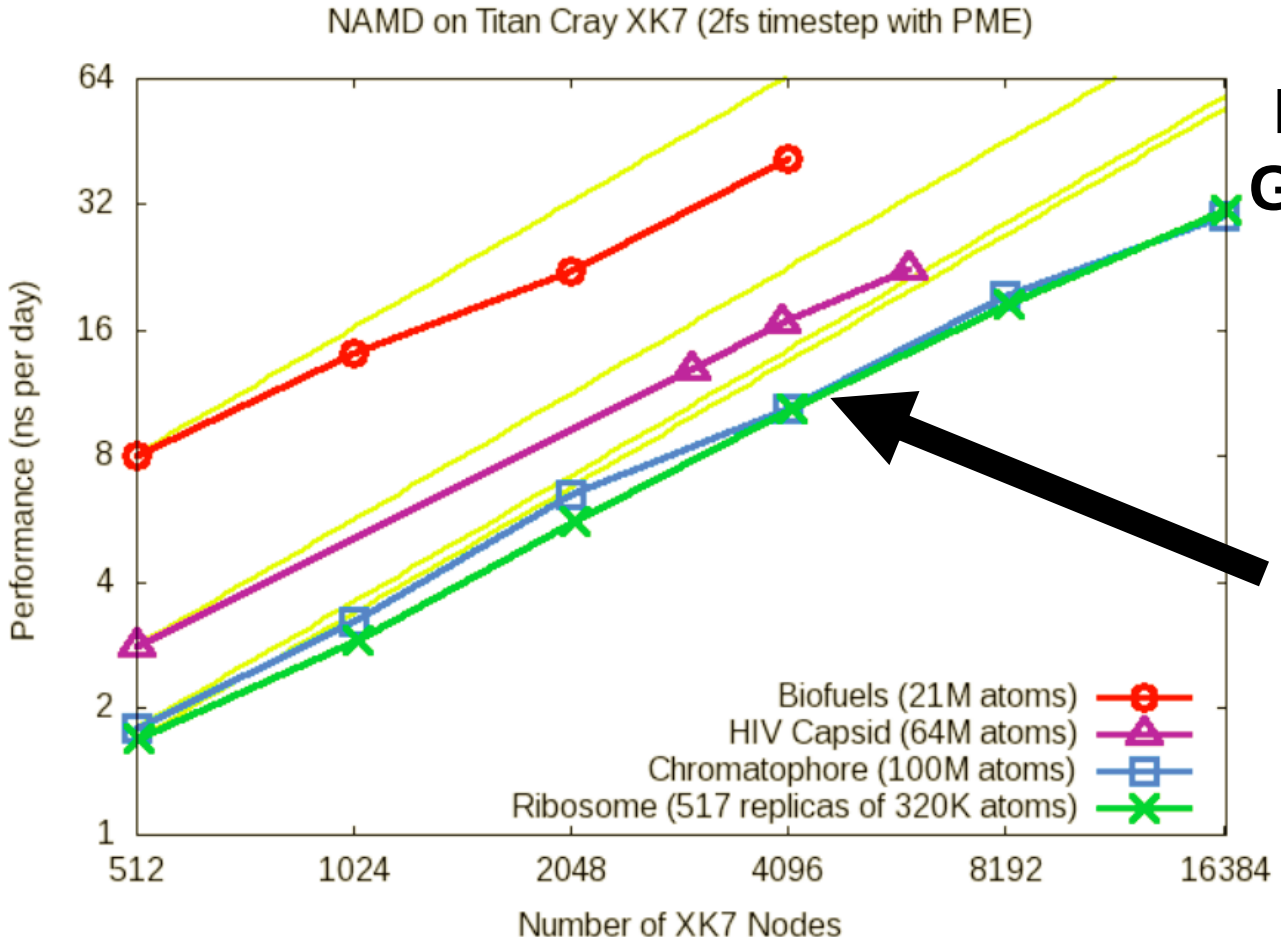
Probing biomolecular machines with graphics processors. Phillips et al. CACM, 52:34-41, 2009.

GPU-accelerated molecular modeling coming of age. Stone et al., J. Mol. Graphics and Modelling, 29:116-125, 2010.

Quantifying the impact of GPUs on performance and energy efficiency in HPC clusters. Enos et al., International Conference on Green Computing, pp. 317-324, 2010.

Fast analysis of molecular dynamics trajectories with graphics processing units-radial distribution function histogramming. Levine et al., J. Computational Physics, 230:3556-3569, 2011.

NAMD Titan XK7 Performance August 2013



NAMD XK7 vs. XE6
GPU Speedup: 3x-4x

HIV-1 Trajectory:
~1.2 TB/day
@ 4096 XK7
nodes

VMD Petascale Visualization and Analysis

- Analyze/visualize large trajectories too large to transfer off-site:
  - User-defined parallel analysis operations, data types
  - Parallel rendering, movie making
- Supports GPU-accelerated Cray XK7 nodes for both visualization and analysis:
  - **GPU accelerated trajectory analysis w/ CUDA**
  - **OpenGL and GPU ray tracing for visualization and movie rendering**
- Parallel I/O rates up to **275 GB/sec** on 8192 Cray XE6 nodes – can read in **231 TB in 15 minutes!**

Parallel VMD currently available on:

ORNL Titan, NCSA Blue Waters, Indiana Big Red II

NCSA Blue Waters Hybrid Cray XE6 / XK7
22,640 XE6 dual-Opteron CPU nodes
4,224 XK7 nodes w/ Telsa K20X GPUs

Example of parallel VMD startup on IU Big Red II:

johstone@login1:~> qsub -I -V -l nodes=4:ppn=16 -q gpu
qsub: waiting for job 338070 to start
qsub: job 338070 ready

johstone@login1:~> module add vmd
Visual Molecular Dynamics version 1.9.2a36 loaded.

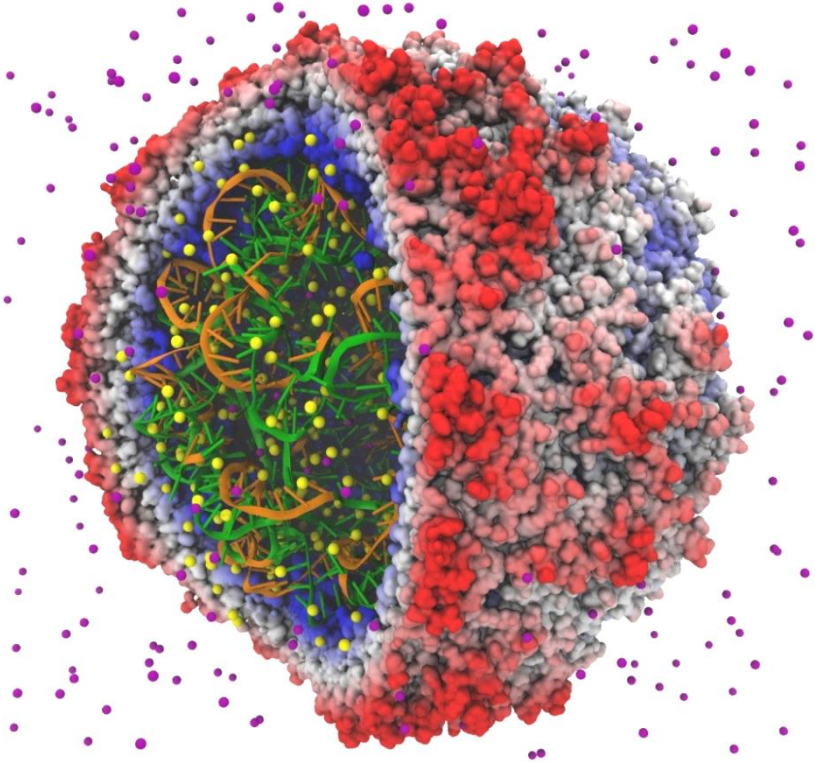
johstone@aprun1:~> vmd -e testmpi.vmd
Launching VMD w/ mppwidth=4, mppdepth=16, mppnppn=1
Info) VMD for BLUEWATERS, version 1.9.2a36 (January 24, 2014)
Info) http://www.ks.uiuc.edu/Research/vmd/
Info) Email questions and bug reports to vmd@ks.uiuc.edu
Info) Please include this reference in published work using VMD:
Info) Humphrey, W., Dalke, A. and Schulten, K., `VMD - Visual
Info) Molecular Dynamics', J. Molec. Graphics 1996, 14.1, 33-38.
Info) --
Info) Creating CUDA device pool and initializing hardware...
Info) Initializing parallel VMD instances via MPI...
Info) Found 4 VMD MPI nodes containing a total of 64 CPUs and 4 GPUs:
Info) 0: 16 CPUs, 30.78GB (97%) free mem, 1 GPUs, Name: nid00838
Info) 1: 16 CPUs, 30.78GB (97%) free mem, 1 GPUs, Name: nid00600
Info) 2: 16 CPUs, 30.79GB (97%) free mem, 1 GPUs, Name: nid00743
Info) 3: 16 CPUs, 30.79GB (97%) free mem, 1 GPUs, Name: nid00749

Visualization Goals, Challenges

- Increased GPU acceleration for visualization of **petascale molecular dynamics trajectories**
- **Overcome GPU memory capacity limits**, enable high quality visualization of >100M atom systems
- Use GPU to accelerate not only interactive-rate visualizations, but also photorealistic ray tracing with **artifact-free ambient occlusion lighting**, etc.
- Maintain **ease-of-use**, intimate link to VMD analytical features, atom selection language, etc.

VMD “QuickSurf” Representation

- Displays continuum of structural detail:
  - All-atom, coarse-grained, cellular models
  - Smoothly variable detail controls
- Linear-time algorithm, scales to millions of particles, as **limited by memory capacity**
- Uses multi-core CPUs and GPU acceleration to enable **smooth interactive animation** of molecular dynamics trajectories w/ up to ~1-2 million atoms
- **GPU acceleration yields 10x-15x speedup vs. multi-core CPUs**



Satellite Tobacco Mosaic Virus

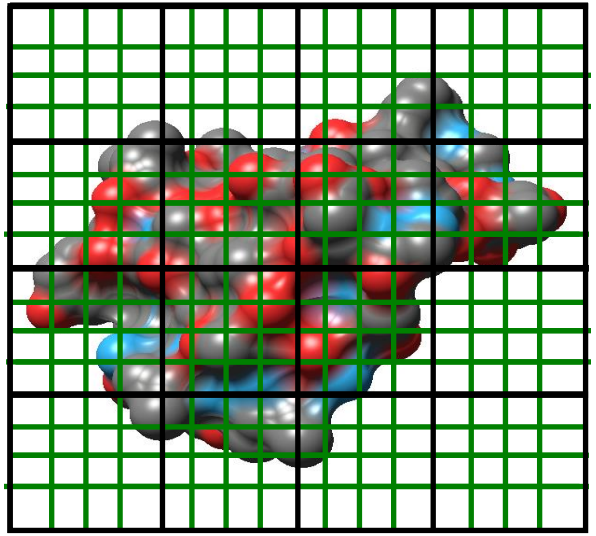
Fast Visualization of Gaussian Density Surfaces for Molecular Dynamics and Particle System Trajectories.
M. Krone, J. E. Stone, T. Ertl, K. Schulten. *EuroVis Short Papers*, pp. 67-71, 2012

VMD 1.9.2 QuickSurf Algorithm Improvements

- **50%-66% memory use, 1.5x-2x speedup**
- Build spatial acceleration data structures, optimize data for GPU
- Compute 3-D density map, 3-D color texture map with **data-parallel *"gather"* algorithm**:

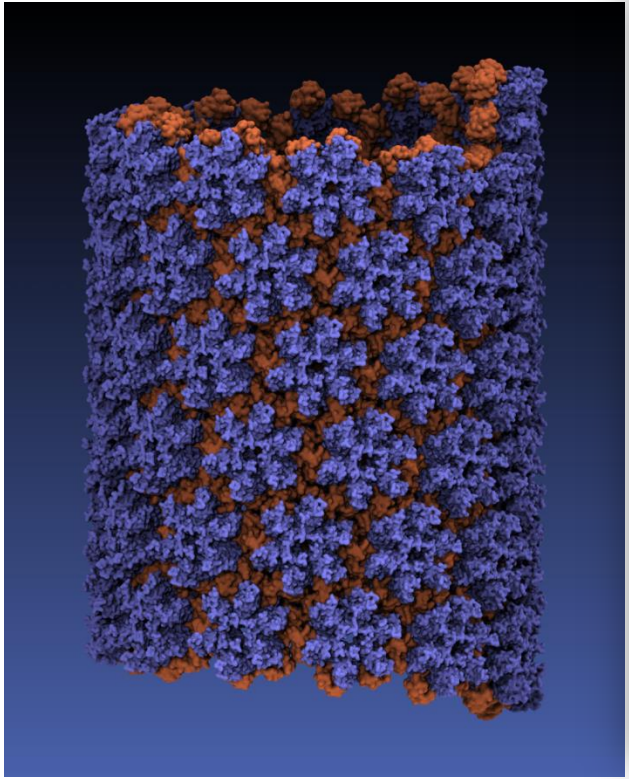
$$\rho(\vec{r}; \vec{r}_1, \vec{r}_2, \ldots, \vec{r}_N) = \sum_{i=1}^{N} e^{\frac{-|\vec{r}-\vec{r}_i|^2}{2\alpha^2}}$$

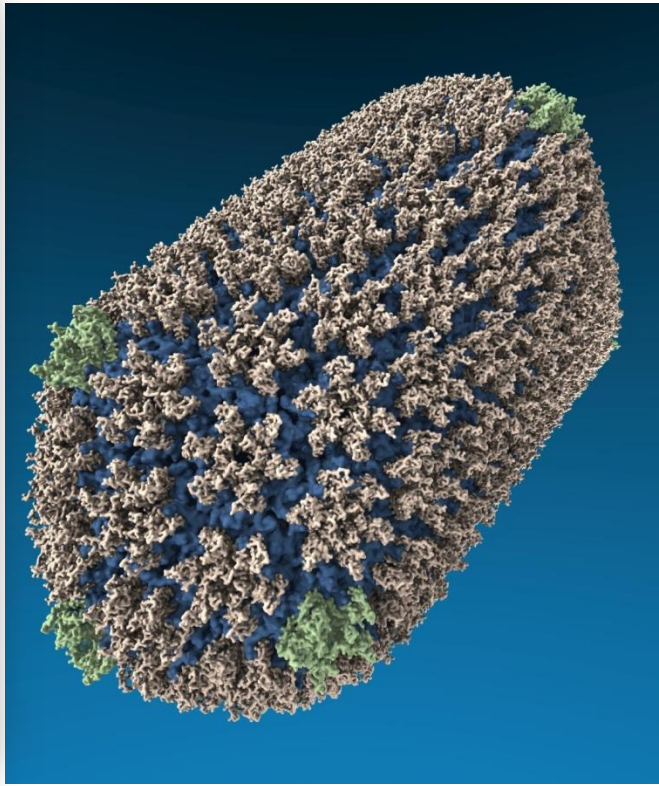
- Normalize, quantize, and compress density, color, surface normal data **while in registers**, before writing out to GPU global memory
- Extract isosurface, maintaining quantized/compressed data representation



3-D density map lattice, spatial acceleration grid, and extracted surface

VMD “QuickSurf” Representation, Ray Tracing





All-atom HIV capsid simulations w/ up to 64M atoms on Blue Waters

VMD w/ OpenGL GLSL vs. GPU Ray Tracing

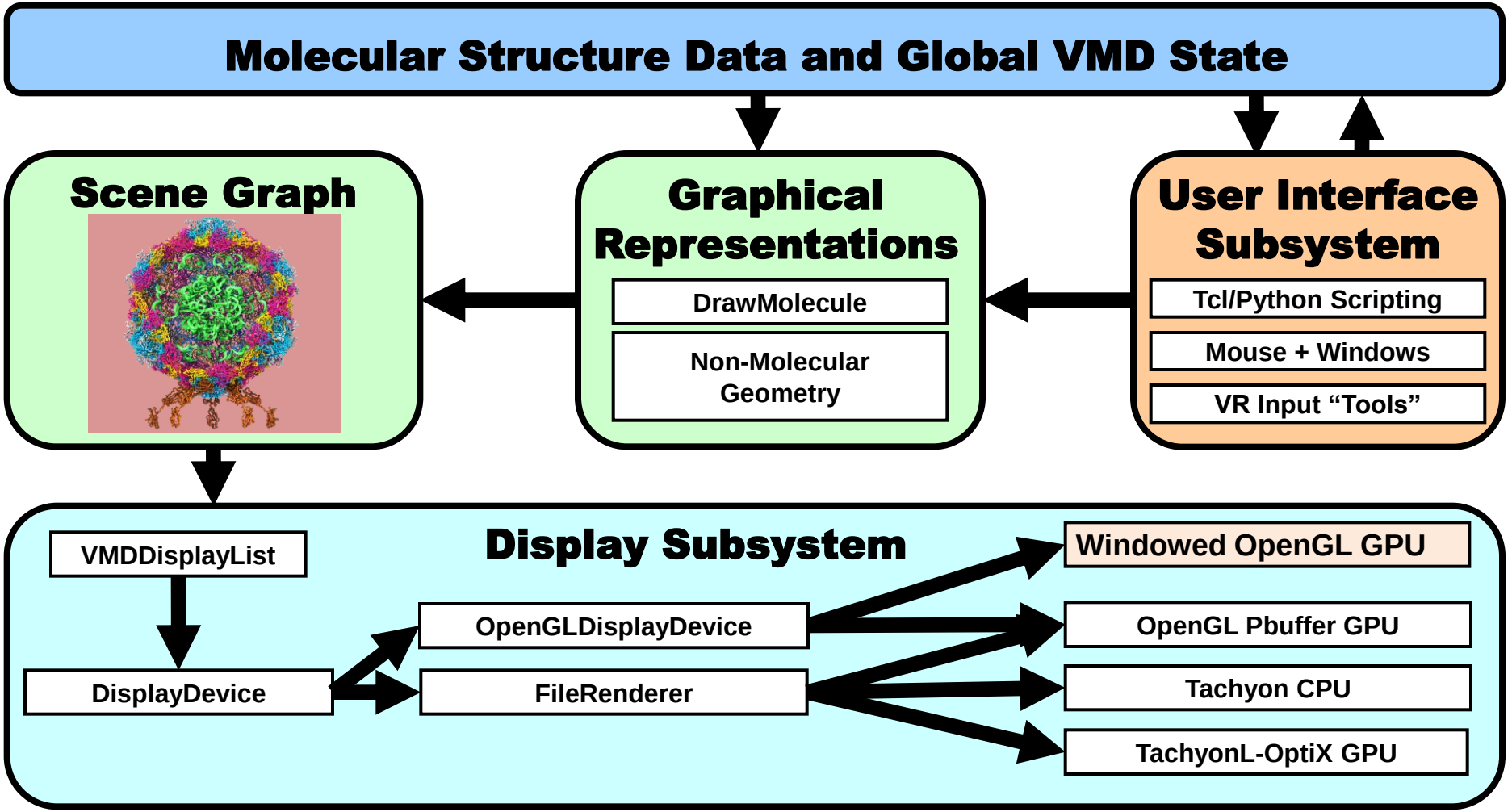
- OpenGL GLSL:
  - No significant per-frame preprocessing required
  - Minimal persistent GPU memory footprint
  - Implements point sprites, ray cast spheres, pixel-rate lighting, …
- GPU Ray Tracing:
  - **Entire scene resident in GPU on-board memory for speed**
  - RT performance is **heavily dependent on BVH** acceleration, particularly for scenes with large secondary ray workloads – shadow rays, ambient occlusion shadow feelers, transmission rays

VMD GPU-Accelerated Ray Tracing Engine

- Complementary to VMD OpenGL GLSL renderer that uses fast, interactivity-oriented rendering techniques
- **Key ray tracing benefits: ambient occlusion lighting, shadows, high quality transparent surfaces, …**
  - Subset of Tachyon parallel ray tracing engine in VMD
  - GPU acceleration w/ CUDA+OptiX ameliorates long rendering times associated with advanced lighting and shading algorithms
    - **Ambient occlusion generates large secondary ray workload**
    - **Transparent surfaces and transmission rays can increase secondary ray counts by another order of magnitude**
  - Adaptation of Tachyon to the GPU required careful avoidance of GPU branch divergence, use of GPU memory layouts, etc.

Why Built-In VMD Ray Tracing Engines?

- **No disk I/O** or communication to outboard renderers
- **Eliminate unnecessary data replication and host-GPU memory transfers**
- Directly operate on VMD internal molecular scene, **quantized/compressed data formats**
- Implement all **curved surface primitives**, volume rendering, texturing, shading features required by VMD
- **Same scripting, analysis, atom selection**, and rendering features are available on all platforms, **graceful CPU fallback**

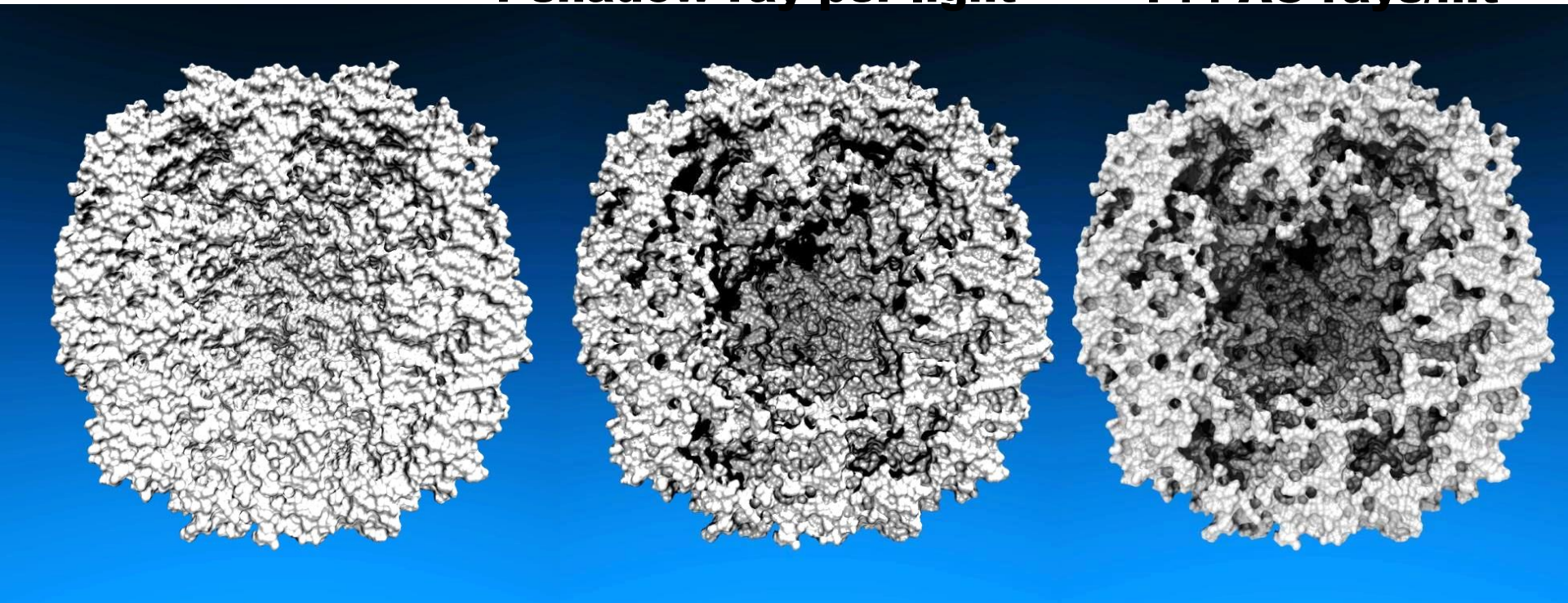

Molecular Structure Data and Global VMD State
Scene Graph
Graphical Representations
DrawMolecule
Non-Molecular Geometry
User Interface Subsystem
Tcl/Python Scripting
Mouse + Windows
VR Input “Tools”
Display Subsystem
VMDDisplayList
DisplayDevice
OpenGLDisplayDevice
FileRenderer
Windowed OpenGL GPU
OpenGL Pbuffer GPU
Tachyon CPU
TachyonL-OptiX GPU

Lighting Comparison, STMV Capsid

Two lights, no shadows

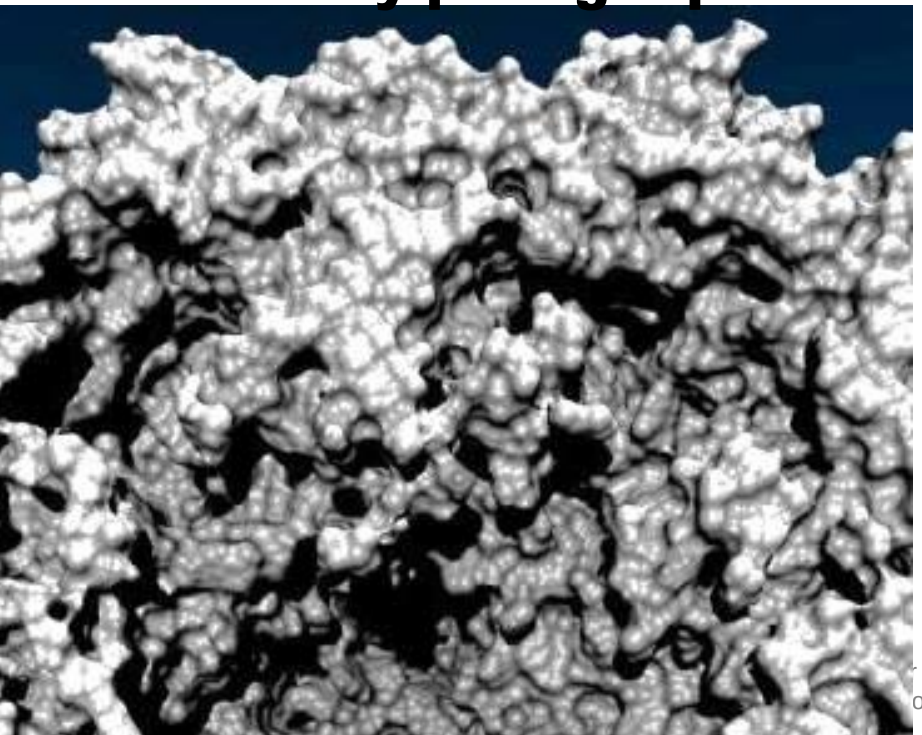
Two lights, hard shadows, 1 shadow ray per light

Ambient occlusion + two lights, 144 AO rays/hit

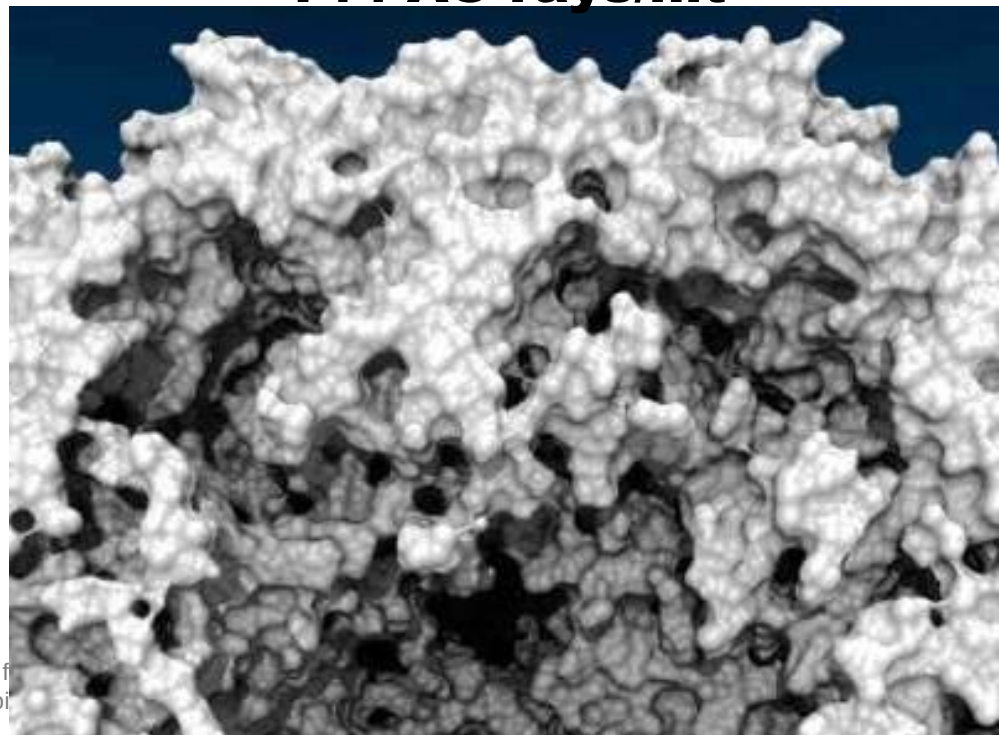


“My Lights are Always in the Wrong Place…”

**Two lights,
harsh shadows,
1 shadow ray per light per hit**

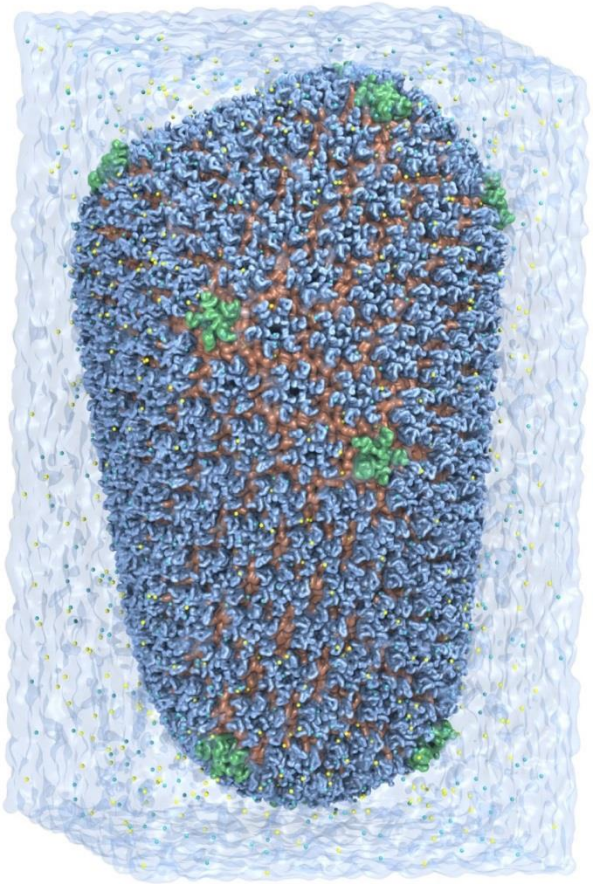


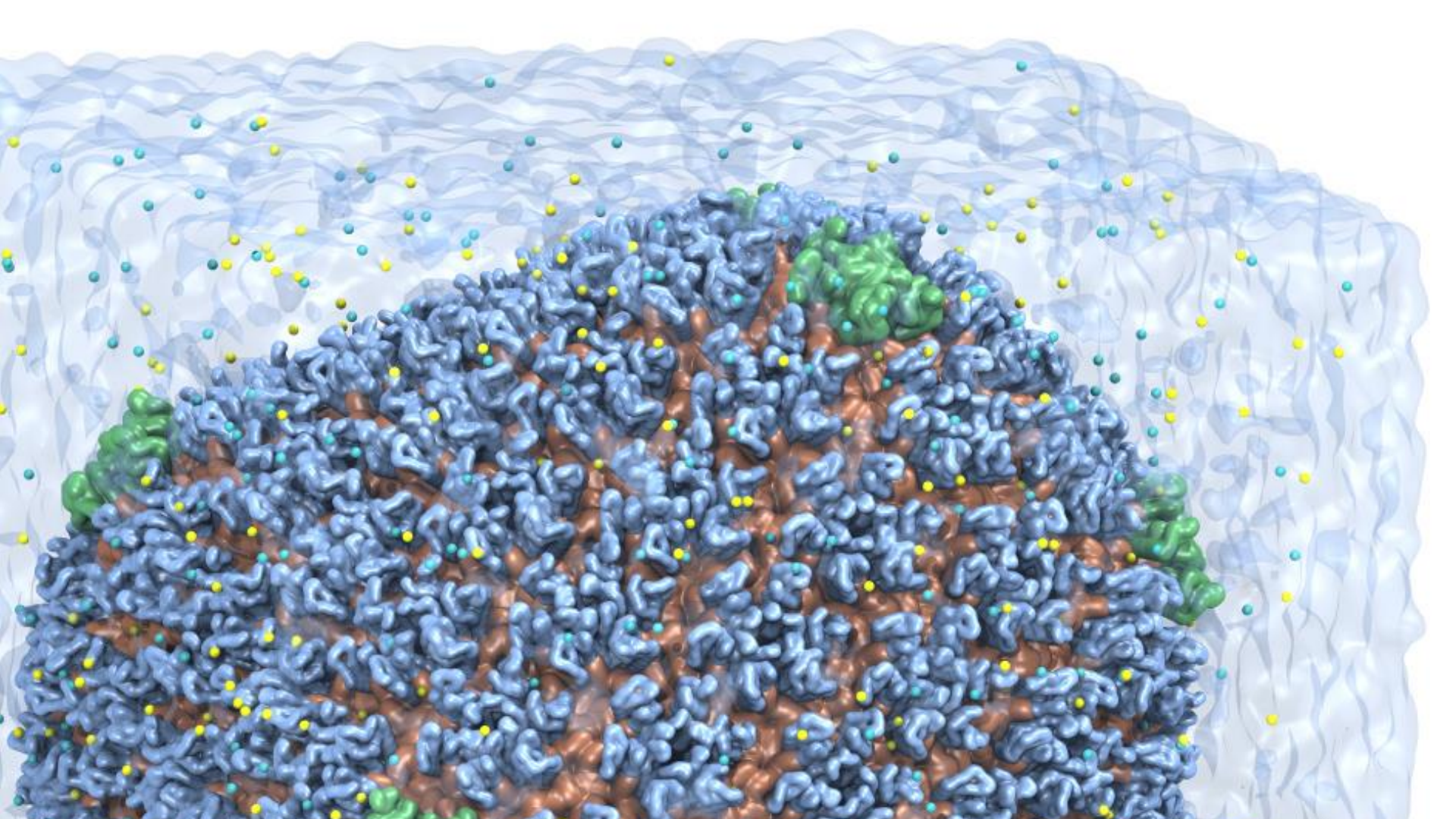
**Ambient occlusion (~80%) +
two directional lights (~20%),
144 AO rays/hit**



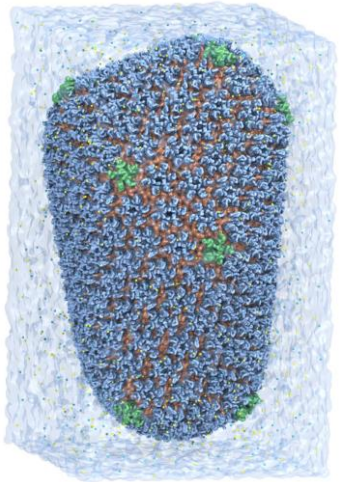
GPU Ray Tracing of HIV-1 on Blue Waters

- 64M atom simulation, 1079 movie frames
- **Ambient occlusion lightin**g, shadows, transparency, antialiasing, depth cueing, **144 rays/pixel minimum**
- GPU memory capacity hurdles:
  - Surface calc. and ray tracing each use **over 75% of K20X 6GB on-board GPU memory** even with quantized/compressed colors, surface normals, …
  - Evict non-RT GPU data to host prior to ray tracing
  - Eviction was **still required** on a test machine with a **12GB Quadro K6000** GPU – the multi-pass "QuickSurf" surface algorithm grows the per-pass chunk size to reduce the number of passes





VMD 1.9.2 HIV-1 Parallel Movie Rendering on Blue Waters Cray XE6/XK7



HIV-1 “HD” 1920x1080 movie rendering:
GPUs speed up geom+ray tracing by **up to eight times**

Node Type and Count	Script Load Time	State Load Time	Geometry + Ray Tracing	Total Time
256 XE6 CPUs	7 s	160 s	**1,374 s**	**1,541 s**
512 XE6 CPUs	13 s	211 s	808 s	1,032 s
64 XK7 Tesla K20X GPUs	2 s	38 s	655 s	695 s
128 XK7 Tesla K20X GPUs	4 s	74 s	331 s	410 s
256 XK7 Tesla K20X GPUs	7 s	110 s	**171 s**	**288 s**

GPU-Accelerated Molecular Visualization on Petascale Supercomputing Platforms,
Stone et al. UltraVis'13: Eighth Workshop on Ultrascale Visualization Proceedings, 2013.

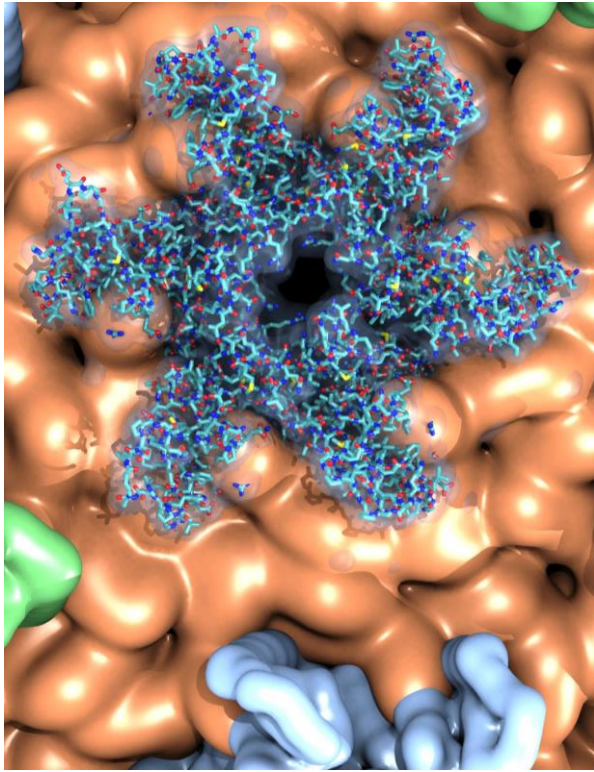
VMD 1.9.2 Coming Soon

- Handle very large structures:
  - Tested with up to 240M atoms/particles
  - Atom selections: 1.5x to 4x faster
  - QuickSurf surface display: 1.5x to 2x faster
  - GPU ray tracing: 4x-8x faster than CPU

- Improved movie making tools, off-screen OpenGL movie rendering, parallel movie rendering

- Improved structure building tools

- Many new and updated user-contributed plugins:
  - Bendix – intuitive helix visualization and analysis
  - NMWiz – visual analysis of normal modes
  - Topotools – structure preparation, e.g. for LAMMPS



GPU Ray Tracing of HIV-1 Capsid Detail

VMD 1.9.2

Force Field Toolkit (*ff*TK)

rapid parameterization of small molecules

Current Features

Optimize charges, bonds, angles, dihedrals

GUI with a defined modular workflow

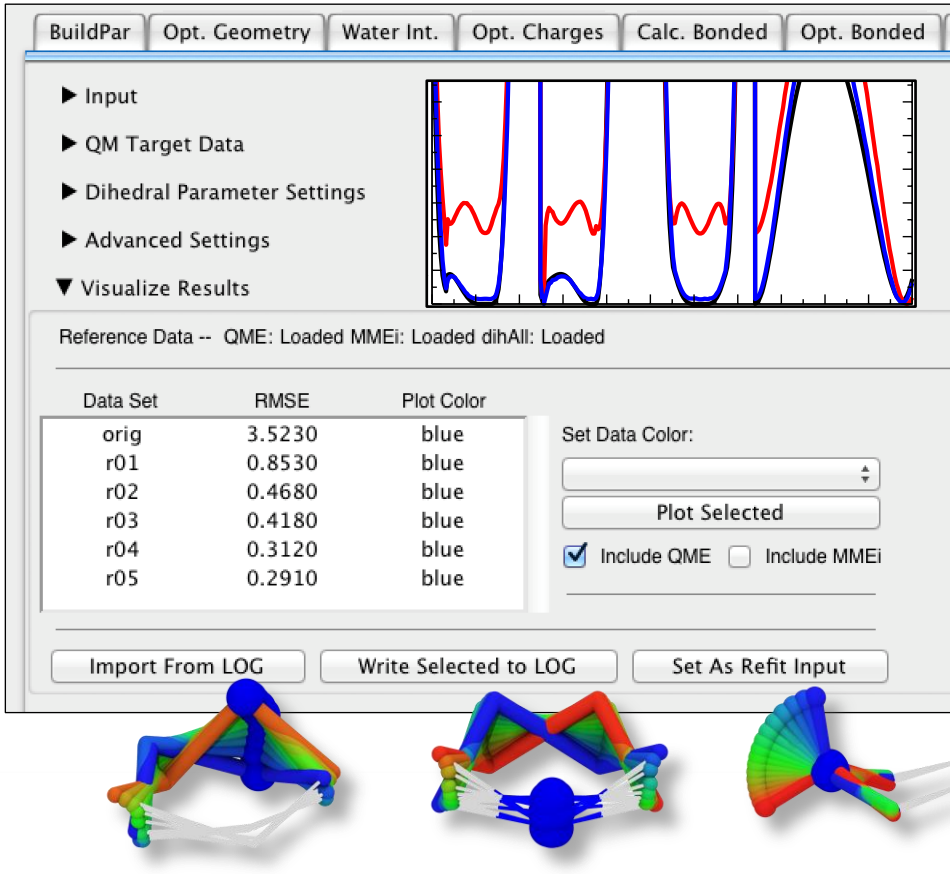
Automation of tedious tasks

Tools to assess parameter performance

Planned Features

Support multiple QM software packages

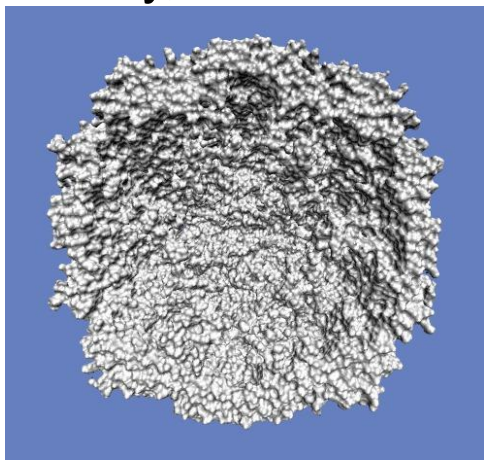
Optimize AMBER parameters



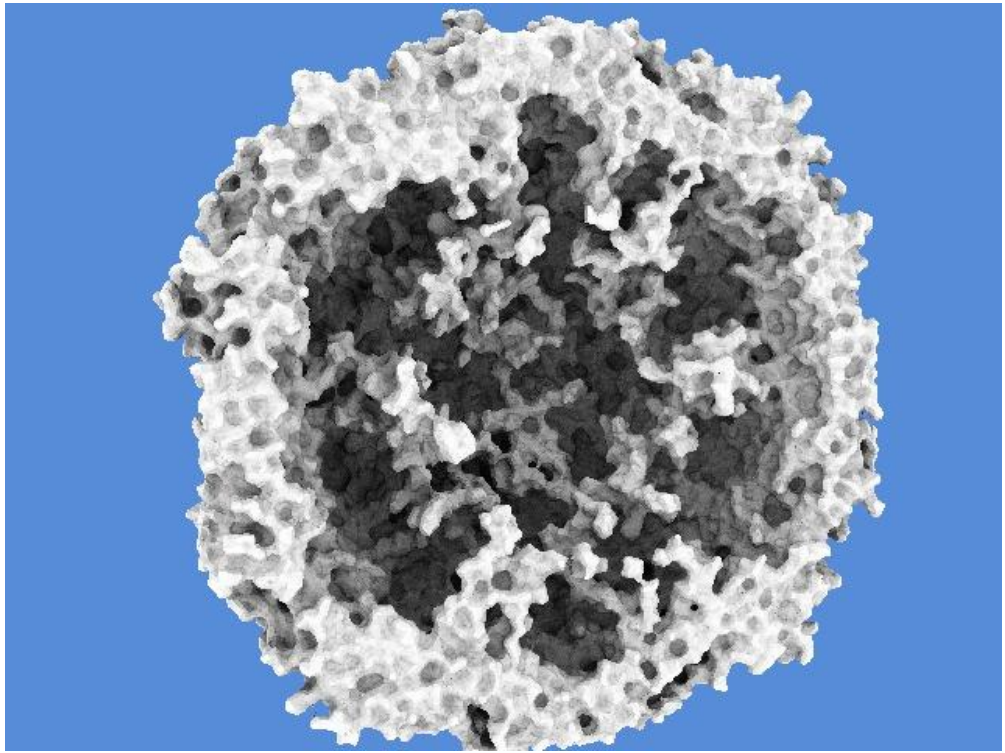
J. Comp. Chem, 34:2757-2770, 2013

Plans: **Interactive** Ray Tracing of Molecular Graphics

- STMV virus capsid on a **laptop** GeForce GTX 560M
- **Ambient occlusion lighting**, shadows, reflections, transparency, and much more…



Standard OpenGL rasterization

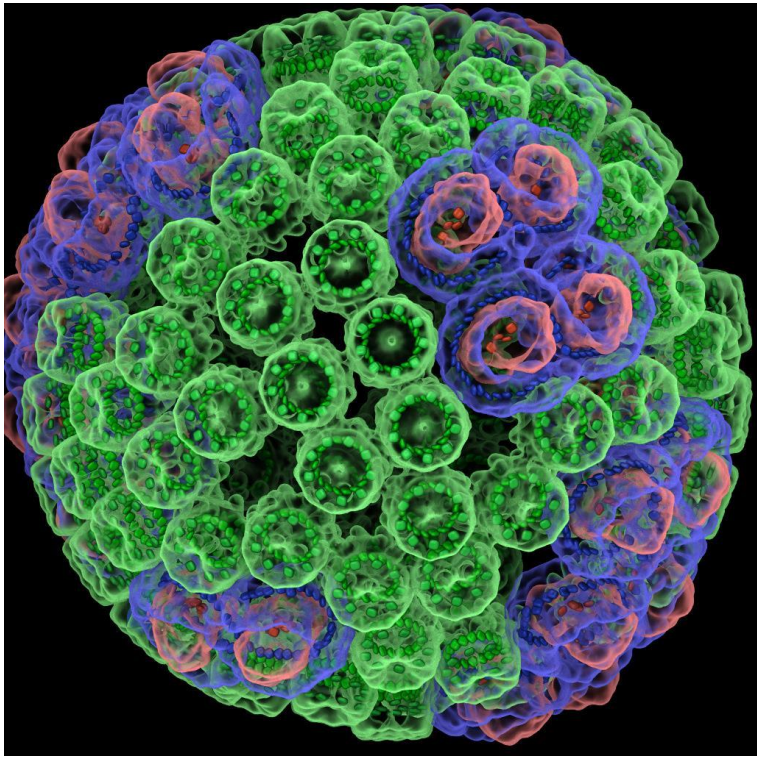


VMD w/ new GPU ray tracing engine based on CUDA + OptiX: 5-10 FPS

Plans: Extend/Improve Structure Building Features of VMD

Exemplary features:

- New tools for **solvation**, **ion placement**, other common structure preparation tasks
- Improve structure building tools for very large biomolecular complexes, e.g. HIV-1 capsid
  - Increase performance
  - Improve user-customizability
  - Support for more structure file formats
- Add infrastructure for development of new cell packing tools for whole cell simulations



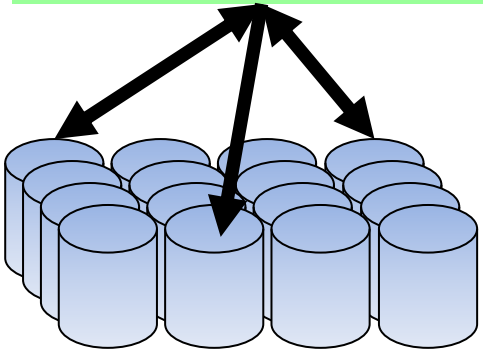
10M atom chromatophore

Plans: Extend Analysis Features of VMD

Exemplary features:

- New secondary structure determination algorithm
  - Support large biomolecular complexes
  - Compute and display time-varying secondary structure interactively
- Simplify analysis of multi-terabyte MD trajectories
  - Circumvent storing large trajectories in memory
  - **Out-of-core SSD trajectory access: 7.5 GB/sec**
- Automate parallelization of user-defined analysis calculations, interfaced to Timeline plugin

Parallel VMD Analysis

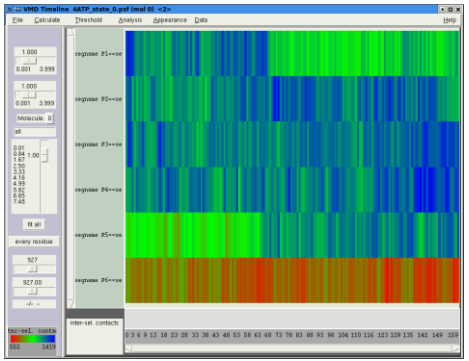


Tens to hundreds of multi-TB trajectories

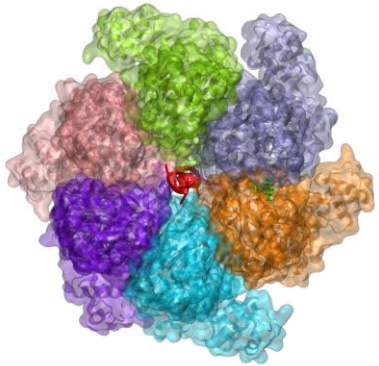
Immersive Out-of-Core Visualization of Large-Size and Long-Timescale Molecular Dynamics Trajectories. J. Stone, K. L. Vandivort, and K. Schulten. G. Bebis et al. (Eds.): *7th International Symposium on Visual Computing (ISVC 2011)*, LNCS 6939, pp. 1-12, 2011.

Early Experiences Scaling VMD Molecular Visualization and Analysis Jobs on Blue Waters, Proceedings of the Extreme Scaling Workshop, Stone, et al., XSEDE Extreme Scaling Workshop, 2013.

Plans: Analyze Long Simulations with Timeline



TimeLine 2D plot



Rho hexameric helicase 3D structure

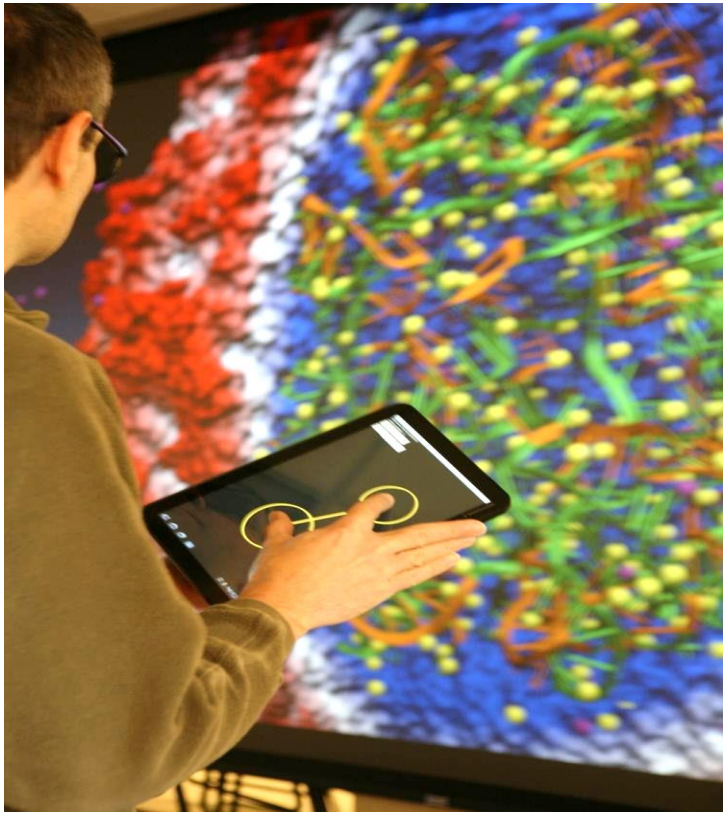
Timeline:

- graphing and analysis tool to **identify events** in an MD trajectory
- live 2D whole-trajectory plot linked to 3D structure
- user-extendable

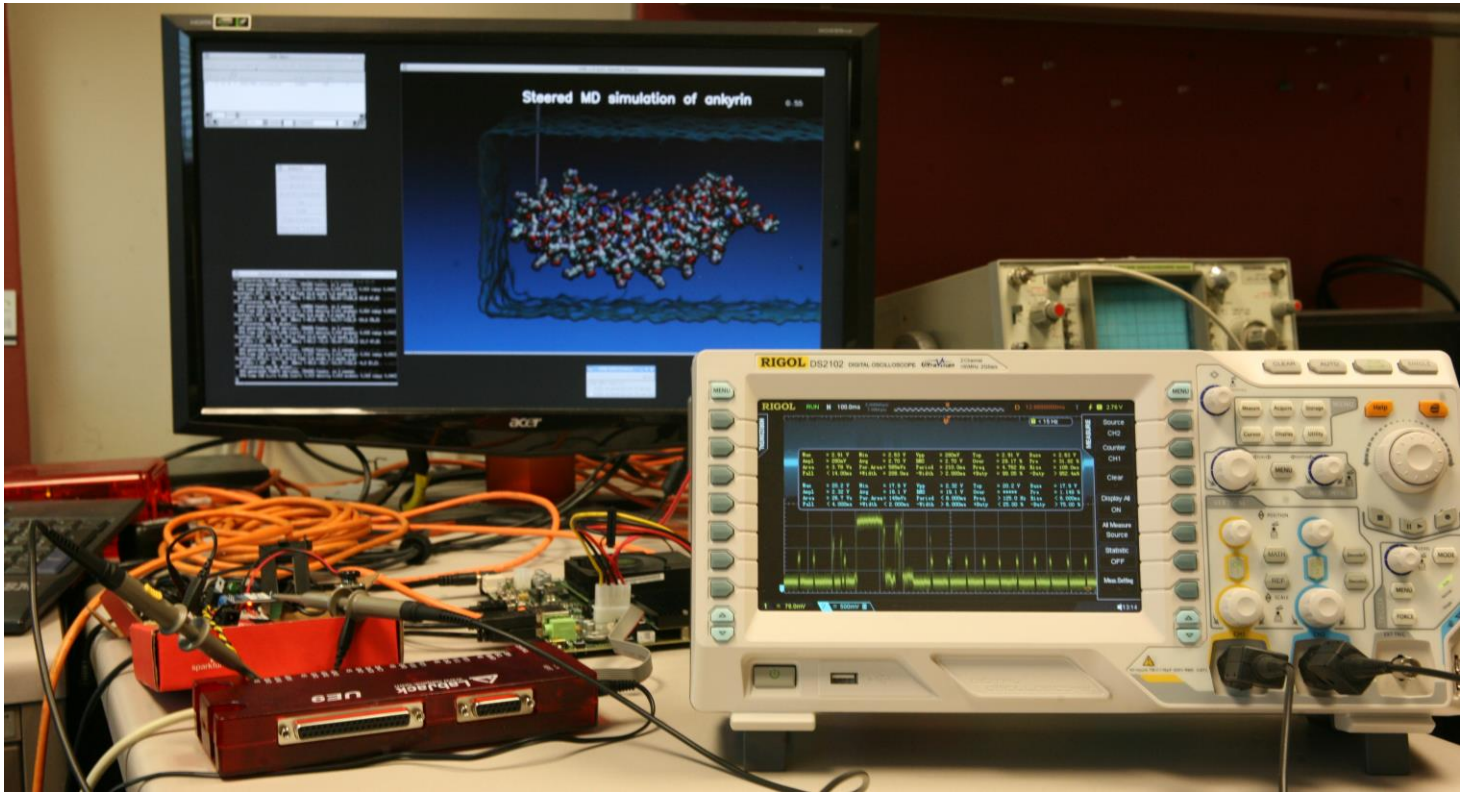
- Perform analysis faster
  - High-performance parallel trajectory analysis on supercomputers and clusters
  - Prototypes show **3500x speedup** on Blue Waters
- Analysis types: filtering, time series analysis, sorting (e.g. bond energies)
- Remote interactive analysis: data at supercomputer center; view in office

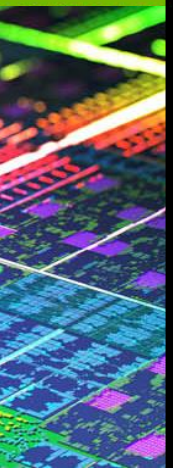
Plans: Tablet VMD, Improved Touch Interfaces

- Developed first multi-touch VMD interface
  - Early technology development in advance of devices
  - Collaborative multi-user wireless control of VMD session
- Features in development:
  - tablet display of trajectory timelines, sequence data, plots, and tabular information
- **Goal: full tablet-native VMD**



Optimizing VMD for Power Consumption





GPU TECHNOLOGY CONFERENCE

Upcoming GTC Express Webinars

February 27: Massive Acceleration of Installed Antenna Performance Simulations Using NVIDIA GPUs

April 3: Getting the Most Out of Citrix XenServer with NVIDIA® vGPU™ Technology

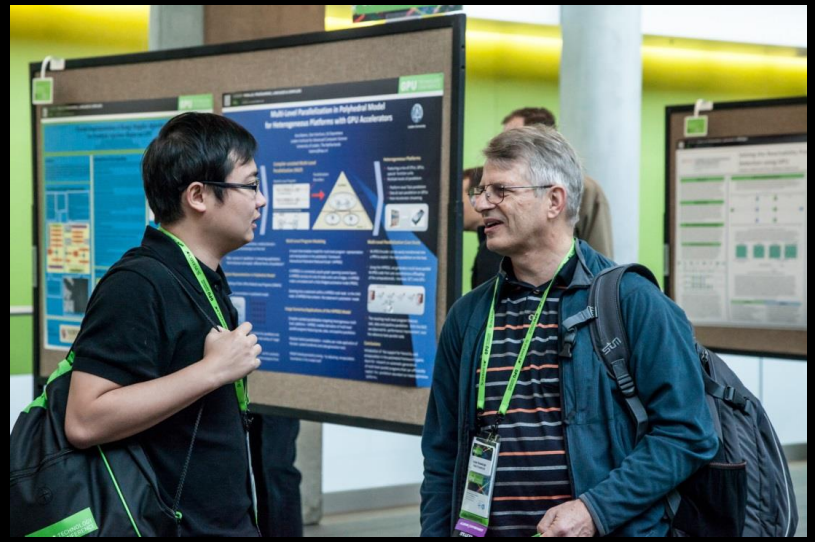
May 27: The Next Steps for Folding@home

Register at www.gputechconf.com/gtcexpress

GTC 2014 Registration is Open

Hundreds of sessions including:

- Molecular Dynamics
- Bioinformatics
- Quantum Chemistry
- Scientific Visualization
- Computational Physics
- Performance Optimization



Register with GM20EXP for 20% discount

www.gputechconf.com

Test Drive the World's Fastest GPU

Accelerate Your Science on the New Tesla K40 GPU

Accelerate applications like
AMBER, NAMD, GROMACS, LAMMPS

OR

Your own Molecular Dynamics code

on latest K40 GPUs today

Sign up for FREE GPU Test Drive
on remotely hosted clusters

www.nvidia.com/GPUTestDrive

Acknowledgements

- Theoretical and Computational Biophysics Group, University of Illinois at Urbana-Champaign
- NVIDIA CUDA Center of Excellence, University of Illinois at Urbana-Champaign
- NVIDIA CUDA team
- NVIDIA OptiX team
- NCSA Blue Waters Team
- Funding:
  - DOE INCITE, ORNL Titan: DE-AC05-00OR22725
  - NSF Blue Waters:
    NSF OCI 07-25070, PRAC “The Computational Microscope”
  - NIH support: 9P41GM104601, 5R01GM098243-02


NIH BTRC for Macromolecular Modeling and Bioinformatics
1990-2017
Beckman Institute
University of Illinois at
Urbana-Champaign

GPU Computing Publications

http://www.ks.uiuc.edu/Research/gpu/

- **GPU-Accelerated Molecular Visualization on Petascale Supercomputing Platforms.** J. Stone, K. L. Vandivort, and K. Schulten. UltraVis'13: Proceedings of the 8th International Workshop on Ultrascale Visualization, pp. 6:1-6:8, 2013.
- **Early Experiences Scaling VMD Molecular Visualization and Analysis Jobs on Blue Waters.** J. Stone, B. Isralewitz, and K. Schulten. In proceedings, Extreme Scaling Workshop, 2013.
- **Lattice Microbes: High-performance stochastic simulation method for the reaction-diffusion master equation.** E. Roberts, J. Stone, and Z. Luthey-Schulten. J. Computational Chemistry 34 (3), 245-255, 2013**.**
- **Fast Visualization of Gaussian Density Surfaces for Molecular Dynamics and Particle System Trajectories.** M. Krone, J. Stone, T. Ertl, and K. Schulten. *EuroVis Short Papers,* pp. 67-71, 2012.
- **Immersive Out-of-Core Visualization of Large-Size and Long-Timescale Molecular Dynamics Trajectories.** J. Stone, K. L. Vandivort, and K. Schulten. G. Bebis et al. (Eds.): *7th International Symposium on Visual Computing (ISVC 2011)*, LNCS 6939, pp. 1-12, 2011.
- **Fast Analysis of Molecular Dynamics Trajectories with Graphics Processing Units – Radial Distribution Functions.** B. Levine, J. Stone, and A. Kohlmeyer. *J. Comp. Physics*, 230(9):3556-3569, 2011.

GPU Computing Publications

http://www.ks.uiuc.edu/Research/gpu/

- **Quantifying the Impact of GPUs on Performance and Energy Efficiency in HPC Clusters.** J. Enos, C. Steffen, J. Fullop, M. Showerman, G. Shi, K. Esler, V. Kindratenko, J. Stone, J Phillips. *International Conference on Green Computing,* pp. 317-324, 2010.
- **GPU-accelerated molecular modeling coming of age.** J. Stone, D. Hardy, I. Ufimtsev, K. Schulten. *J. Molecular Graphics and Modeling,* 29:116-125, 2010.
- **OpenCL: A Parallel Programming Standard for Heterogeneous Computing.** J. Stone, D. Gohara, G. Shi. *Computing in Science and Engineering,* 12(3):66-73, 2010.
- **An Asymmetric Distributed Shared Memory Model for Heterogeneous Computing Systems**. I. Gelado, J. Stone, J. Cabezas, S. Patel, N. Navarro, W. Hwu. *ASPLOS '10: Proceedings of the 15th International Conference on Architectural Support for Programming Languages and Operating Systems,* pp. 347-358, 2010.

GPU Computing Publications

http://www.ks.uiuc.edu/Research/gpu/

- **GPU Clusters for High Performance Computing**. V. Kindratenko, J. Enos, G. Shi, M. Showerman, G. Arnold, J. Stone, J. Phillips, W. Hwu. *Workshop on Parallel Programming on Accelerator Clusters (PPAC),* In Proceedings IEEE Cluster 2009, pp. 1-8, Aug. 2009.
- **Long time-scale simulations of in vivo diffusion using GPU hardware**. E. Roberts, J. Stone, L. Sepulveda, W. Hwu, Z. Luthey-Schulten. In *IPDPS'09: Proceedings of the 2009 IEEE International Symposium on Parallel & Distributed Computing*, pp. 1-8, 2009.
- **High Performance Computation and Interactive Display of Molecular Orbitals on GPUs and Multi-core CPUs**. J. Stone, J. Saam, D. Hardy, K. Vandivort, W. Hwu, K. Schulten, *2nd Workshop on General-Purpose Computation on Graphics Pricessing Units (GPGPU-2), ACM International Conference Proceeding Series*, volume 383, pp. 9-18, 2009.
- **Probing Biomolecular Machines with Graphics Processors**. J. Phillips, J. Stone. *Communications of the ACM,* 52(10):34-41, 2009.
- **Multilevel summation of electrostatic potentials using graphics processing units**. D. Hardy, J. Stone, K. Schulten. *J. Parallel Computing*, 35:164-177, 2009.

GPU Computing Publications

http://www.ks.uiuc.edu/Research/gpu/

- **Adapting a message-driven parallel application to GPU-accelerated clusters**. J. Phillips, J. Stone, K. Schulten. *Proceedings of the 2008 ACM/IEEE Conference on Supercomputing*, IEEE Press, 2008.
- **GPU acceleration of cutoff pair potentials for molecular modeling applications**. C. Rodrigues, D. Hardy, J. Stone, K. Schulten, and W. Hwu. *Proceedings of the 2008 Conference On Computing Frontiers*, pp. 273-282, 2008.
- **GPU computing**. J. Owens, M. Houston, D. Luebke, S. Green, J. Stone, J. Phillips. *Proceedings of the IEEE*, 96:879-899, 2008.
- **Accelerating molecular modeling applications with graphics processors**. J. Stone, J. Phillips, P. Freddolino, D. Hardy, L. Trabuco, K. Schulten. *J. Comp. Chem.*, 28:2618-2640, 2007.
- **Continuous fluorescence microphotolysis and correlation spectroscopy**. A. Arkhipov, J. Hüve, M. Kahms, R. Peters, K. Schulten. *Biophysical Journal*, 93:4006-4017, 2007.