

ATPESC Track 4: Scalable Molecular Visualization and Analysis Tools in VMD

John E. Stone

Theoretical and Computational Biophysics Group
Beckman Institute for Advanced Science and Technology
University of Illinois at Urbana-Champaign

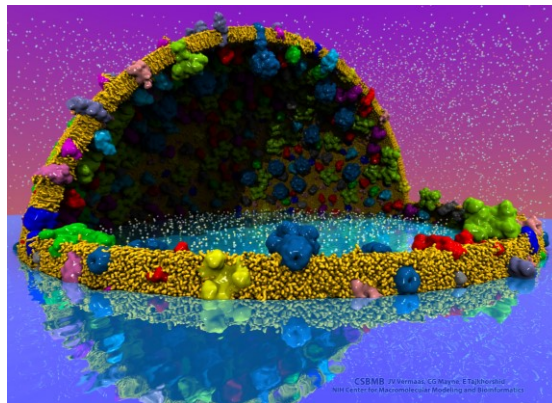
<http://www.ks.uiuc.edu/Research/vmd/>

Argonne Training Program on Exascale Computing (ATPESC)

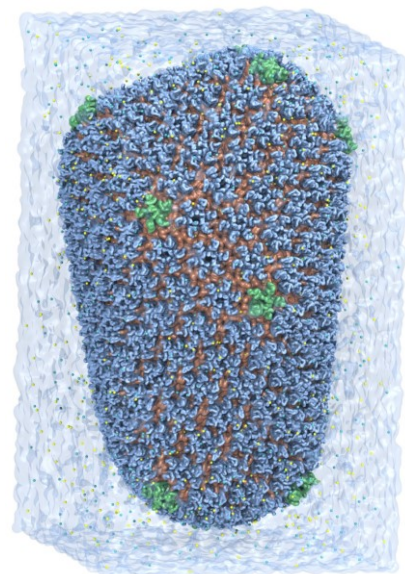
3:45pm-4:30pm, St. Charles Amphitheater, Q Center,
St. Charles, IL, Monday August 5th, 2019

VMD – “Visual Molecular Dynamics”

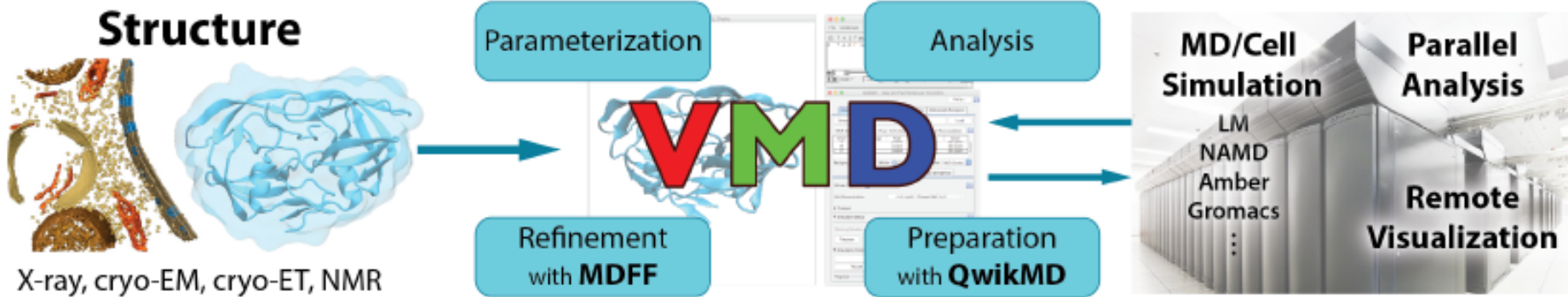
- 100,000 active users worldwide
- Visualization and analysis of:
 - Molecular dynamics simulations
 - Lattice cell simulations
 - Quantum chemistry calculations
 - Cryo-EM densities, volumetric data
- User extensible scripting and plugins
- <http://www.ks.uiuc.edu/Research/vmd/>



Cell-Scale Modeling



MD Simulation



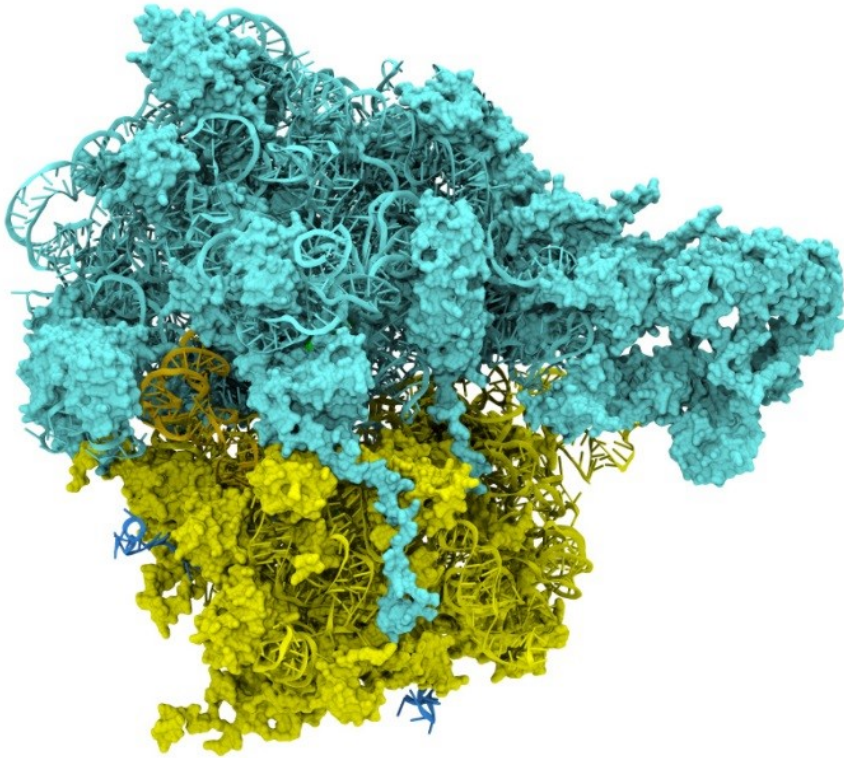
VMD Hands-On Tutorials

- <http://www.ks.uiuc.edu/Training/Tutorials/#vmd>
 - Main VMD tutorial
 - QwikMD simulation preparation and analysis plugin
 - VMD images and movies tutorial
 - Structure check
 - VMD quantum chemistry visualization tutorial
 - Visualization and analysis of CPMD data with VMD
 - Parameterizing small molecules using ffTK

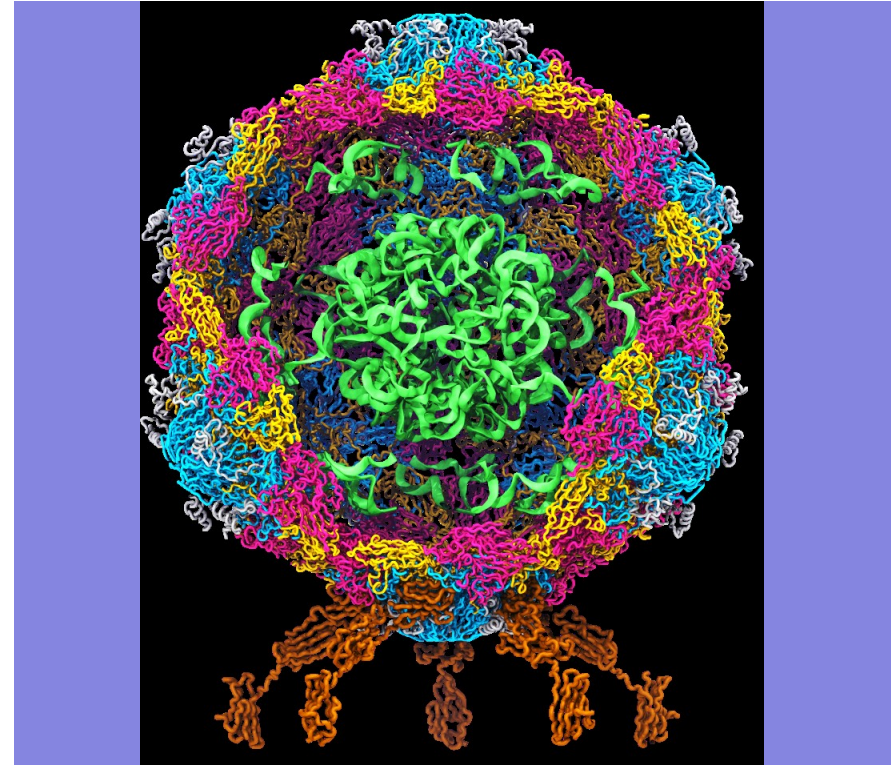
Goal: A Computational Microscope

Study the molecular machines in living cells

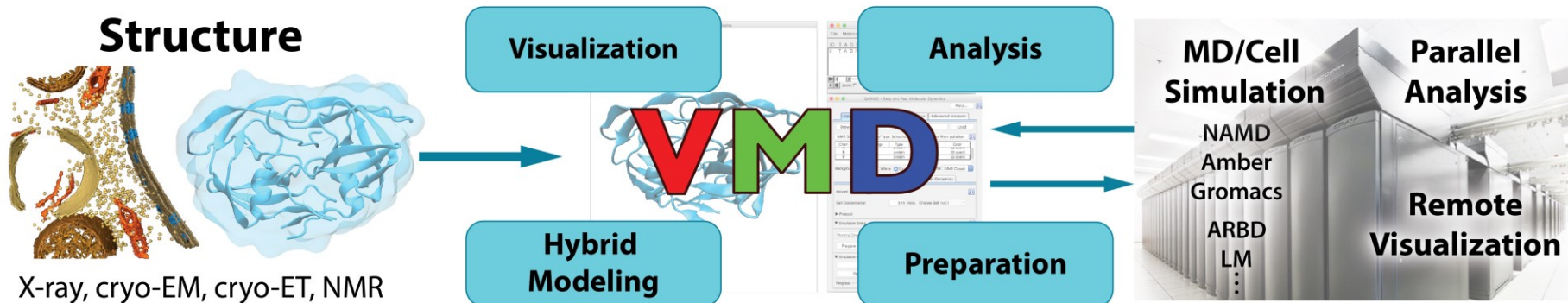
Ribosome: target for antibiotics



Poliovirus



VMD: Building A Next Generation Modeling Platform

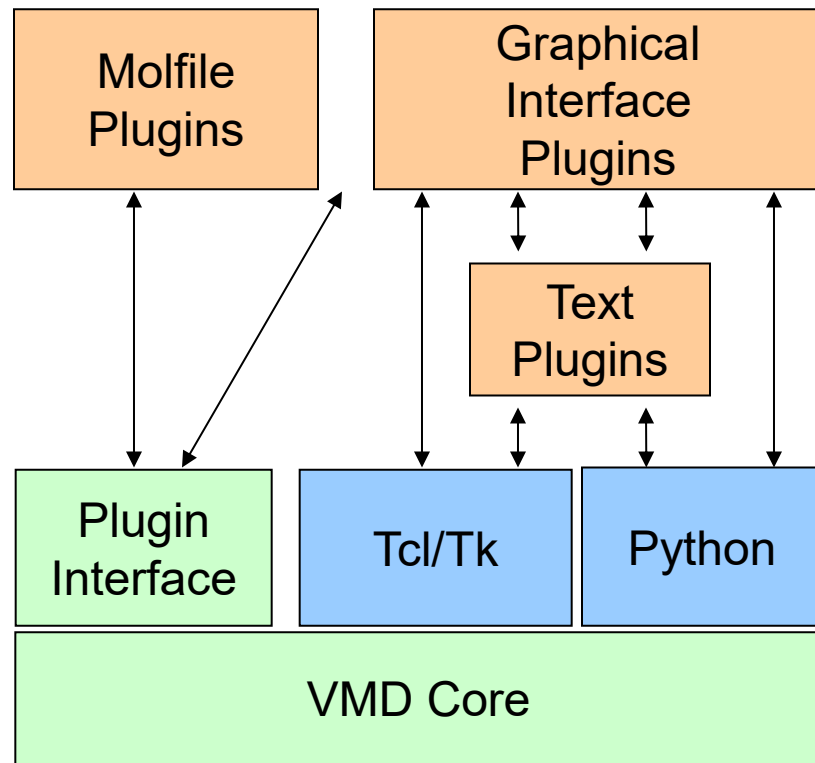


- Provide tools for simulation preparation, visualization, and analysis
 - Reach cell-scale modeling w/ all-atom MD, coarse grained, Lattice Microbes
 - Improved performance, visual fidelity, exploit advanced technologies (GPUs, VR HMDs)
- Enable hybrid modeling and computational electron microscopy
 - Load, filter, process, interpret, visualize multi-modal structural information
- Connect key software tools to enable state-of-the-art simulations
 - Support new data types, file formats, software interfaces
- Openness, extensibility, and interoperability are VMD hallmarks
 - Reusable algorithms made available in NAMD, for other tools

VMD is a Platform for Developing Research Tools

Over 110 VMD Plugins, Half Developed by Users

- VMD user-extensible scripting w/ Tcl/Tk, Python
- User-developed plugins:
 - Alanine Scanning
 - Collective Variable Analyzer
 - Clustering Tool
 - Carbon Nanostructure Builder
 - TorsionPlot
 - RMSD Trajectory Tool
 - Many others...



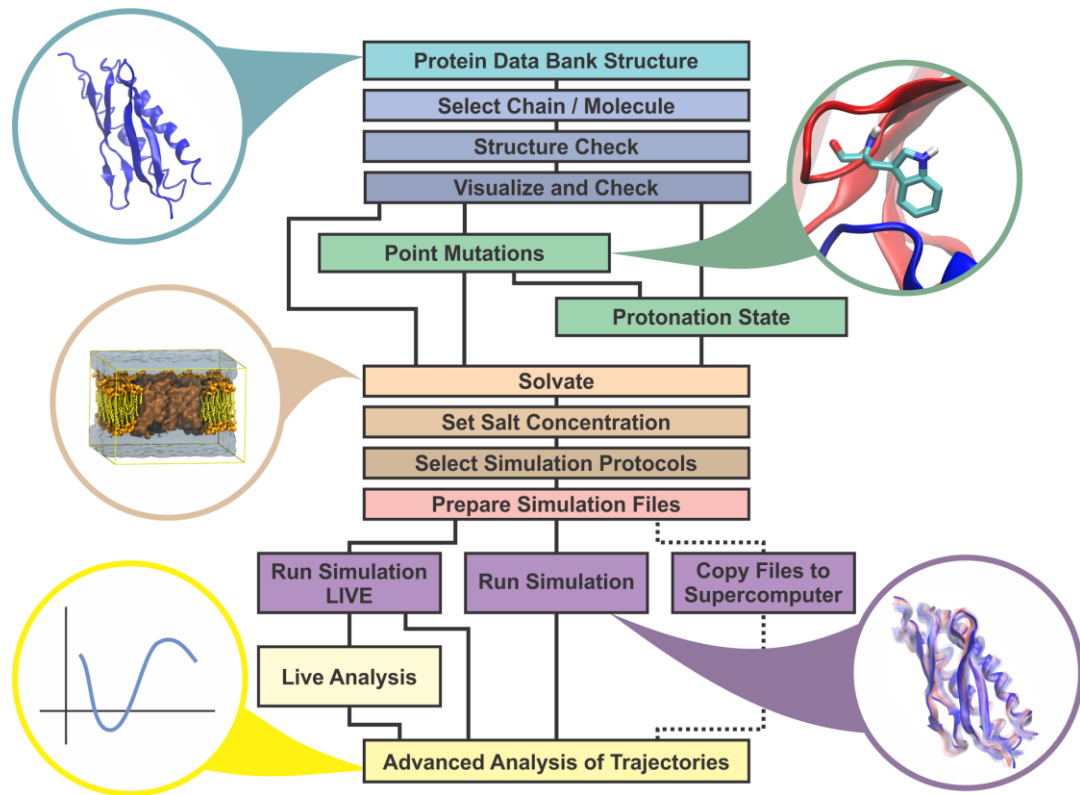
QwikMD: Guided MD Simulation and Training

Smooths initial learning curve (non-expert users)

Speed up tedious simulation preparation tasks (expert users)

Reproducibility:
detailed log of all steps

Interactive preparation, simulation, and analysis

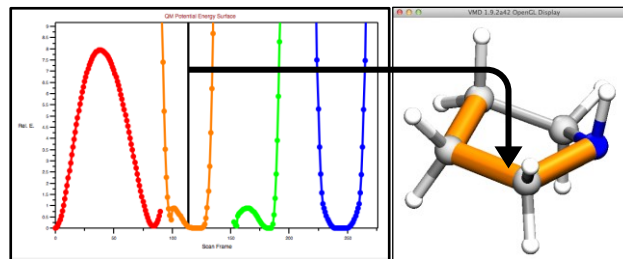
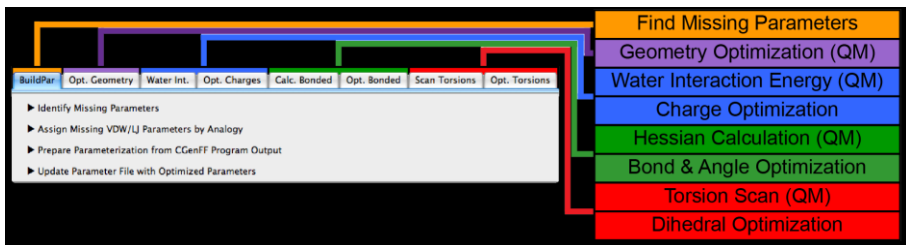


Parametrization with the Force Field Toolkit (FFTK)

A plugin enabling the simulation of drug-like small molecules by lowering force field development barriers

A graphical interface organized to follow the standardized CGenFF workflow

Leverages VMD's powerful graphics capabilities to visualize parameter-related data within the context of molecular structure



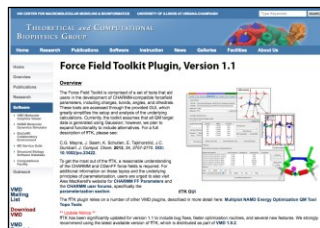
Import data from the CGenFF Program webserver to assign initial parameters by analogy

Interactively explore energy profiles

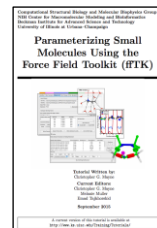
Future Plans:

- Support free-for-academics QM software, **ORCA (Neese, Max Planck)**
- Parametrization tools for additional MM force fields
 - AMBER force field (Gumbart, Georgia Tech.)
 - Drude polarizable force fields (Roux, U. Chicago; MacKerell, U. Maryland)

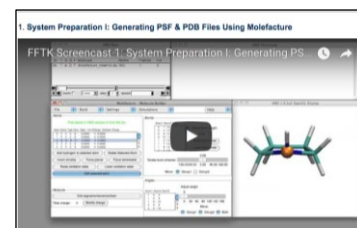
Extensive training material available from the Center's website



Documentation Website



Workshop Tutorial



Screencast Demos

Selected VMD Plugins: Center Developed, and User Developed

Analysis

APBSRun
CatDCD
Contact Map
[GofRGUI](#)
[HeatMapper](#)
ILSTools
[IRSpecGUI](#)
MultiSeq
NAMD Energy
NAMD Plot
NetworkView
[NMWiz](#)
[ParseFEP](#)
PBCTools
PMEpot
[PropKa GUI](#)
RamaPlot
RMSD Tool
[RMSD Trajectory Tool](#)
[RMSD Visualizer Tool](#)
Salt Bridges
Sequence Viewer
Symmetry Tool
Timeline
[TorsionPlot](#)
VolMap

Modeling

AutoIonize
AutoPSF
Chirality
Cionize
Cispeptide
CGTools
Dowser
ffTK
Inorganic Builder
MDFP
Membrane
Merge Structs
Molefacture
Mutator
[Nanotube](#)
Psfgen
[RESPTool](#)
RNAView
Solvate
SSRestrains
Topotools

Visualization

Clipping Plane Tool
[Clone Rep](#)
DemoMaster
[Dipole Watcher](#)
[Intersurf](#)
[Navigate](#)
NavFly
[MultiMolAnim](#)
Color Scale Bar
Remote
Palette Tool
ViewChangeRender
ViewMaster
[Virtual DNA Viewer](#)
VMD Movie Maker

Simulation

AlaScan
AutoIMD
IMDMenu
NAMD GUI
NAMD Server
QMTTool

Collaboration

Remote Control

Data Import and Plotting

Data Import
Multiplot
PDBTool
MultiText

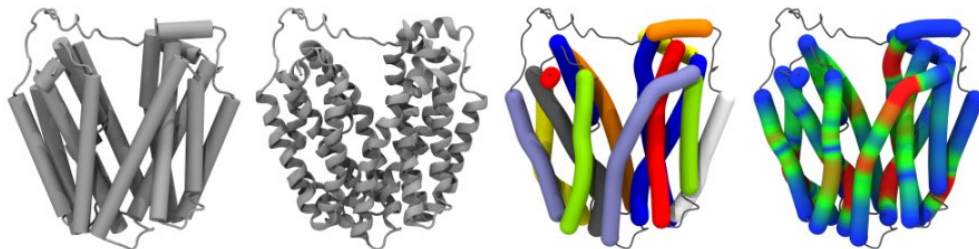
Externally Hosted Plugins and Extensions

[Check sidechains](#)
[MultiMSMS](#)
[Interactive Essential Dynamics](#)
[Mead Ionize](#)
[Clustering Tool](#)
[iTrajComp](#)
[Swap RMSD](#)
[Intervor](#)
[SurfVol](#)
[vmdICE](#)

75 MolFile I/O Plugins:
structure, trajectory, sequence,
and density map

<http://www.ks.uiuc.edu/Research/vmd/plugins/>

Example VMD Visualization and Analysis Plugins



Bendix

Dahl ACE, Chavent M and Sansom MSP Bendix: intuitive helix geometry analysis and abstraction. *Bioinformatics* 2012 28(16): 2193-2194.



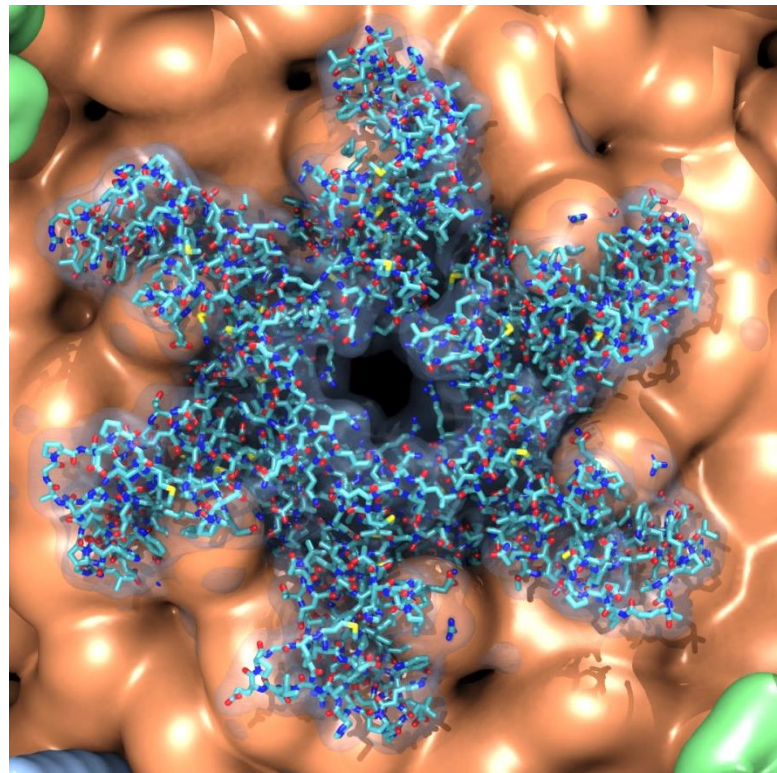
Normal Mode Wizard

Bakan A, Meireles LM, Bahar I ProDy: Protein Dynamics Inferred from Theory and Experiments. *Bioinformatics* 2011 27(11):1575-1577.

VMD “Coming Soon”:
VMD 1.9.4 and VMD-Next

VMD 1.9.4, and VMD-Next

- Python 3.x support
- New “molefactory” structure editor plugin
- Improved structure building and analysis tools
- High performance GPU structure+data clustering
- Density map and volume processing features: high performance GPU image segmentation, density map simulation, masking, visualization
- Many new and updated user-contributed plugins
- Deeper integration of interactive ray tracing
 - Seamless interactive RT in main VMD display window
 - Support trajectory playback in interactive RT
 - Enable multi-node interactive RT on HPC systems
- Built-in (basic) interactive remote visualization on HPC clusters and supercomputers



GPU Ray Tracing of
HIV-1 Capsid Detail

VMD Data Model

VMD “Molecule”

- Collection of self-consistent molecular information
 - Atomic structure, time-varying atomic coordinates, velocities, ...
 - Volumetric data
 - Graphics objects
- Key missing information required for visualization is auto-generated heuristically – “guessed”, if not provided in input files
 - Bond topology
- Limitations:
 - Fixed atom count per-timestep, requires use of “dummy atoms” for so-called “open system” simulations
 - VMD doesn’t store or use MD force field parameters, various other data that are typically specific to particular simulation tools, force fields, etc.

Atomic Structure

- Per-Atom attributes:
 - Name
 - Type
 - Residue ID, Unique residue ID, Residue Name, Residue Type
 - Bond list
 - Atomic number
 - Alternate location identifier
 - Insertion code
 - Chain
 - Segment name
 - Atom/Residue classification (protein, nucleic, hydrogen, water, “other”)

Other, Optional Per-Atom Fields

- Optional per-atom attributes:
 - Mass
 - Charge
 - Radius
 - Occupancy
 - B-factor
 - Fields primarily used by simulation preparation tools:
 - Bond Orders
 - Angles, Dihedrals, Improvers, Cross-Term Maps

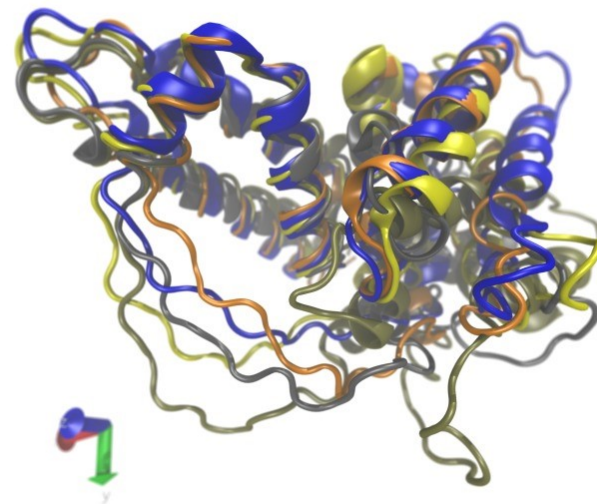
Time-Varying Trajectory Data

- Atomic coordinates
- Atomic velocities
- PBC unit cell
- QM, QM/MM molecular orbital data
- User-defined time-varying scalar quantities:
 - user, user2, user3, user4

VMD Development Efforts

Supporting Integrative Hybrid Modeling

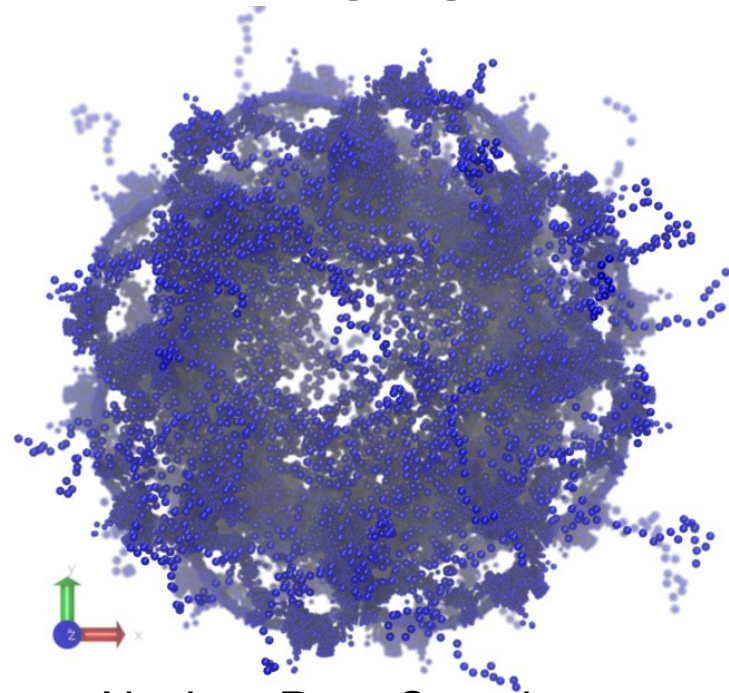
- Extending mmCIF PDBx parser to encompass new IHM-specific records, data types
- Revising VMD “molfile plugin” APIs to communicate IHM data to VMD and represent it natively
- New atom selection keywords that encompass IHM structure data
- New graphical interfaces to query and interact with IHM data both quantitatively and visually



Serum Albumin Domains,
PDB-DEV IHM #5

Coarse-Grained IHM Data

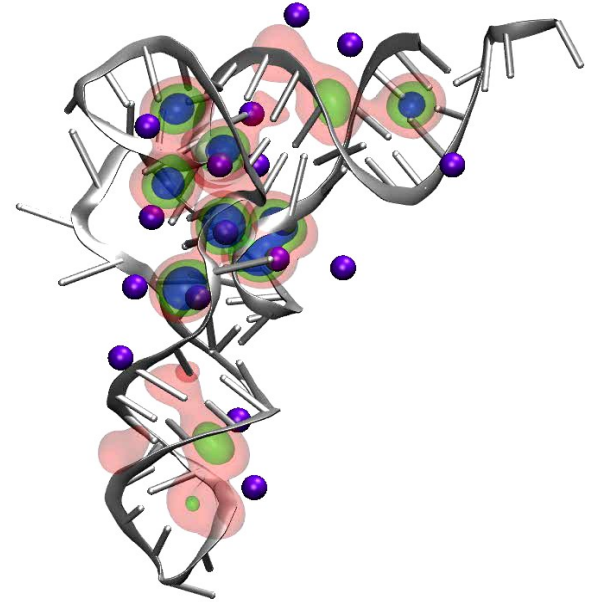
- Coarse grained sphere/bead models
- Restraint information from experiments
- Multi-modal structure alignments, comparisons
- Linkage to underlying experimental images, statistics, etc.



Nuclear Pore Complex,
PDB-DEV IHM #12

Display of Uncertainty, Error in IHM Models

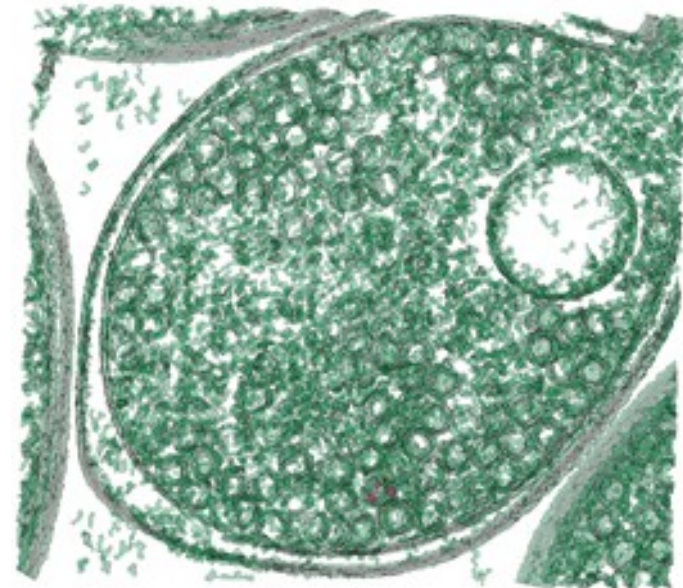
- Query, visualize uncertainty, error, variance, in EM density maps, tomograms, atomic structure
- Requires IHM models to specify these statistics in the files
- Modeling tools, graphical interfaces can use this to guide user modeling tasks, analyses



**tRNA magnesium ion occupancy probability
density surfaces, VMD volmap plugin**

Computing Challenges Posed by Large Hybrid Models

- Techniques like coarse graining allow modeling to reach the cell scale, but data sizes and interactivity remain a tremendous challenge
- Next-generation parallel- and GPU-accelerated computing techniques can make powerful analytical and visualization tools **interactive** for the first time:
 - Clustering analyses (structure RMSD, quality-of-fit, docking scores, etc)
 - Image segmentation, docking, alignment, fitting, coarse-graining...



VMD supports analysis and visualization of multi-gigavoxel EM tomograms, density maps

Interoperability Challenges Posed by IHM Data

- Reliable interoperability between modeling and simulation tools hinges upon deep understanding of IHM data semantics
- When creating multi-modal hybrid models, quantitative and visual representation of “uncertainty” and “error” becomes important, so that the best information modalities are used to make analyses, decisions, and interpretations
- What are the physical properties assigned to coarse-grained “beads”, Gaussians?
- Can those properties be understood among many different modeling tools?

Overcoming the Biomolecular Orientation of VMD for Materials Science Work

- VMD molecular data structures are optimized for biomolecules
 - Expects atom/residue names that (roughly) follow RCSB Protein Databank conventions
 - Significant up-front structure analysis enables increased interactivity later
 - Cost of this is decreased generality in various cases:
 - Fixed atom count per timestep
 - Unrecognized atom naming conventions can inhibit use of some of the high-level atom selections
- Nanodevice and materials science data can be displayed in VMD
 - Assign per-atom name/type fields after loading
 - Enable VMD to better recognize structural elements
 - Example: xyz or LAMMPS dump files lack some expected fields, so it helps to use special features of LAMMPS plugin to assign them

VMD LAMMPS Molfile Plugin

<https://sites.google.com/site/akohlmey/software/lammps-plugin>

- Reads LAMMPS text mode trajectory “dump” files
- Map LAMMPS per-atom fields to VMD molfile API:
 - LAMMPSREMAPFIELDS environment variable allows user to control mapping of per-atom fields to VMD molfile API
 - List of mappings of LAMMPS fields into VMD fields, e.g. “VMD=LAMMPS”
 - Example to map LAMMPS forces into VMD velocity fields:
set env(LAMMPSREMAPFIELDS) “vx=fx,vy=fy,vz=fz”
- Adapt LAMMPS trajectories with varying atom counts to VMD
 - LAMMPSMAXATOMS environment variable sets peak atom count for VMD, adding “dummy” atoms as placeholders:
Dummy atoms initialized with properties: name = @, type = X, rename = @@@, segid = @@@, chain = @, mass =0, charge=0, radius=0, element=X.
 - LAMMPSDUMMYPOS sets default atomic coords for dummy atoms that don’t exist for a given frame:
set env(LAMMPSDUMMYPOS) {0.0 0.0 -10.0}

Exporting Molecular Data

- VMD can be used to prepare MD simulations in conjunction with plugins and user scripting
- Need to watch out for assumptions made by heuristics, structure building tools
- Example: bond topology may need to be modified or deleted for some force fields
- Use the most appropriate structure manipulation tools for the MD package in use, e.g.:
 - NAMD: psfgen plugin
 - LAMMPS: topotools plugin

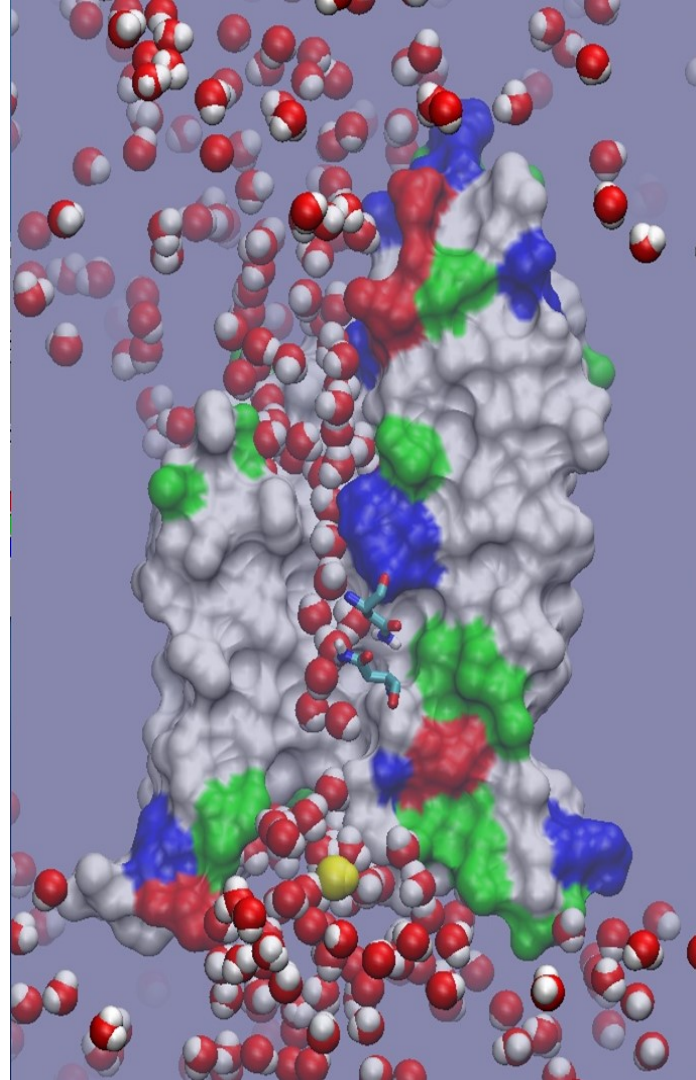
VMD Visualization Concepts

VMD Approach to Visualization

- Molecular scene is composed of “graphical representations”
- Each representation encapsulates a group of selected atoms, a drawing style, coloring style, and other parameters
- Representations are independent of each other, can be toggled on/off easily, allowing molecular scenes to be built-up incrementally
- VMD atom selection language is shared with its analytical and scripting interfaces

Selection, Filtering

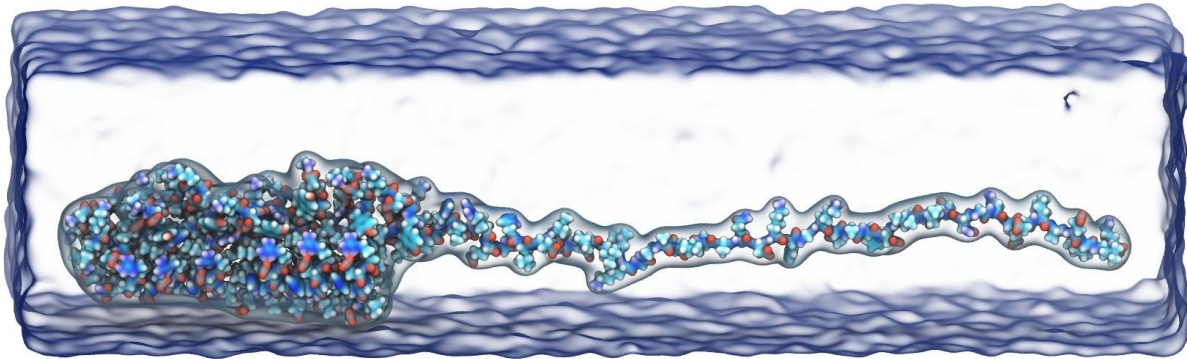
- Most viz tools allow interactive visual picking, menu-driven selections of structure components to display or operate on
- VMD also extensively uses a text-based selection language (think google):
 - water within 10 of protein and $z > 0$
 - nucleic or protein or ions
 - segname BR
 - name "C.*"
- Allows selection on user-defined data fields
- ***Promotes synergy between interactive and scripting interfaces, visualization and quantitative analysis tasks***
- Works well with huge time-varying structures



Structure Visualization

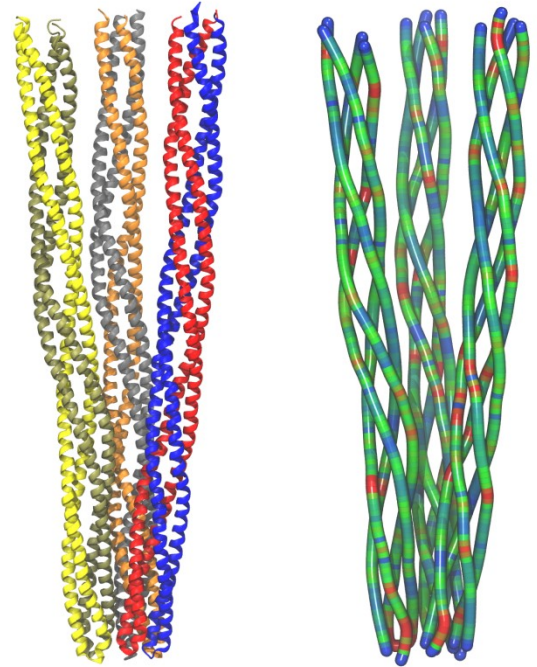
Molecular representations provide different levels of abstraction, atomic detail vs. higher level organizational information

- Atoms, VdW spheres, bonds, ball-stick, ...
- Coarse-grained “beads”
- Ribbons, secondary structure, “cartoon” reps, RNA/DNA
- Molecular surfaces
- Molecular orbitals (quantum chemistry)



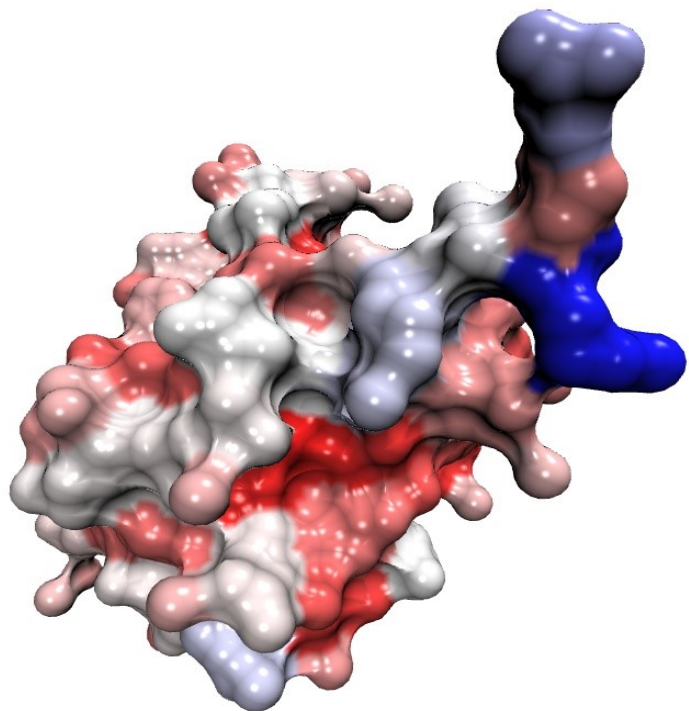
Computed Properties

- Smoothing of thermal noise
- Secondary structure
- Hydrogen bonds, salt bridges
- Forces, energies, stress, strain
- Time averaging of electrostatic fields, occupancy maps
- Quality-of-fit cross correlation with cryo-EM density maps
- Normal modes, principal component analysis, essential dynamics
- Cluster simulation trajectory timesteps by structural similarity

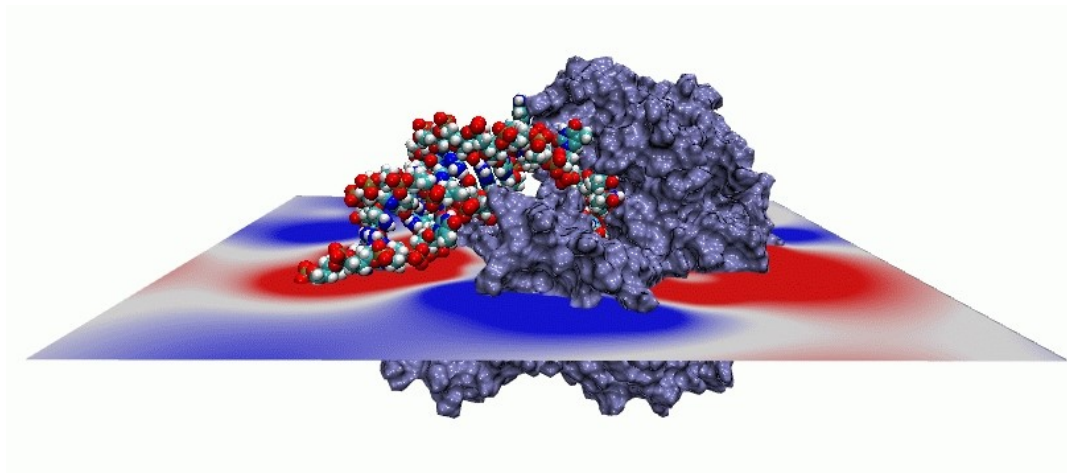


Chemoreceptor trimer-of-dimers analysis with Bendix plugin in VMD

Display of Computed Properties on Structures

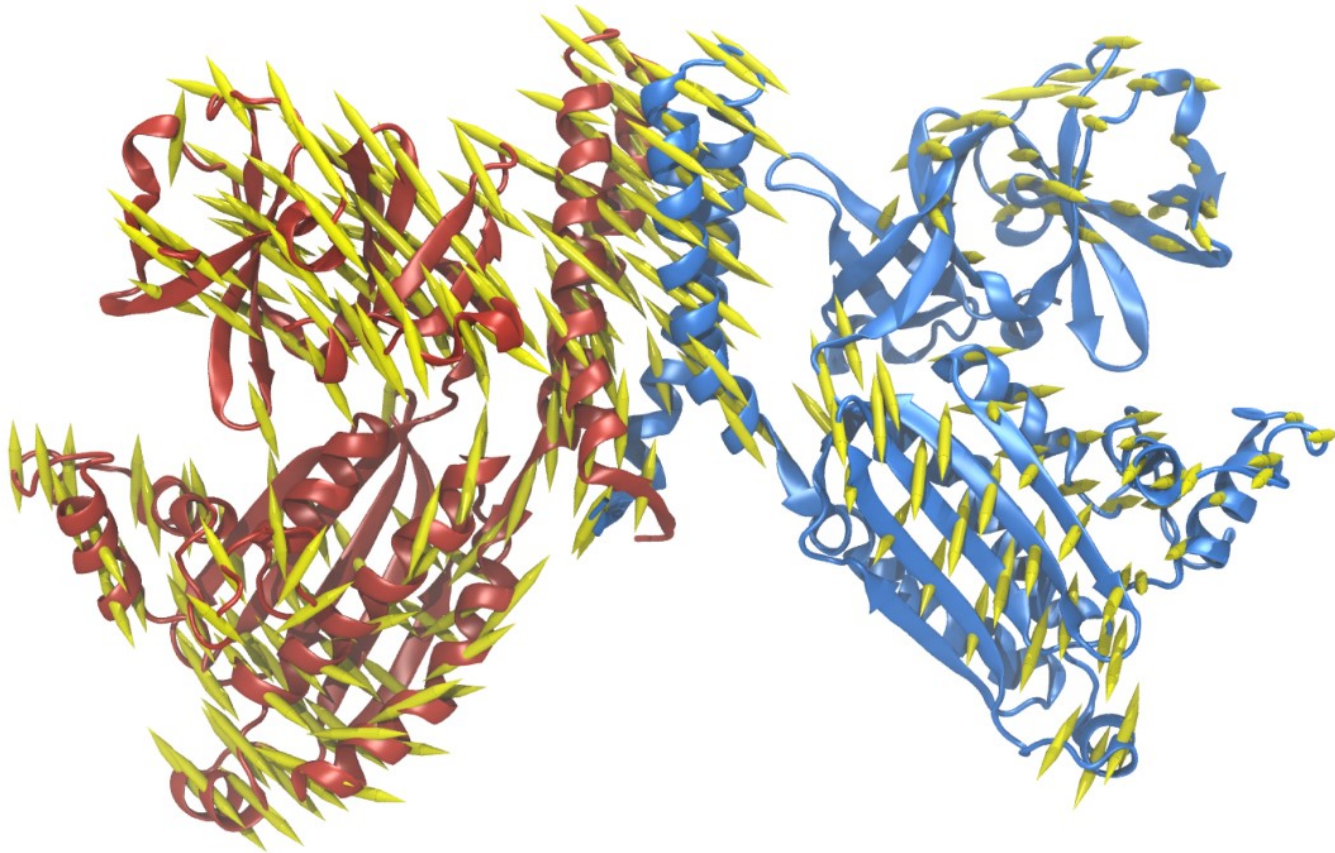


Per-residue solvent-accessible surface area of Ubiquitin



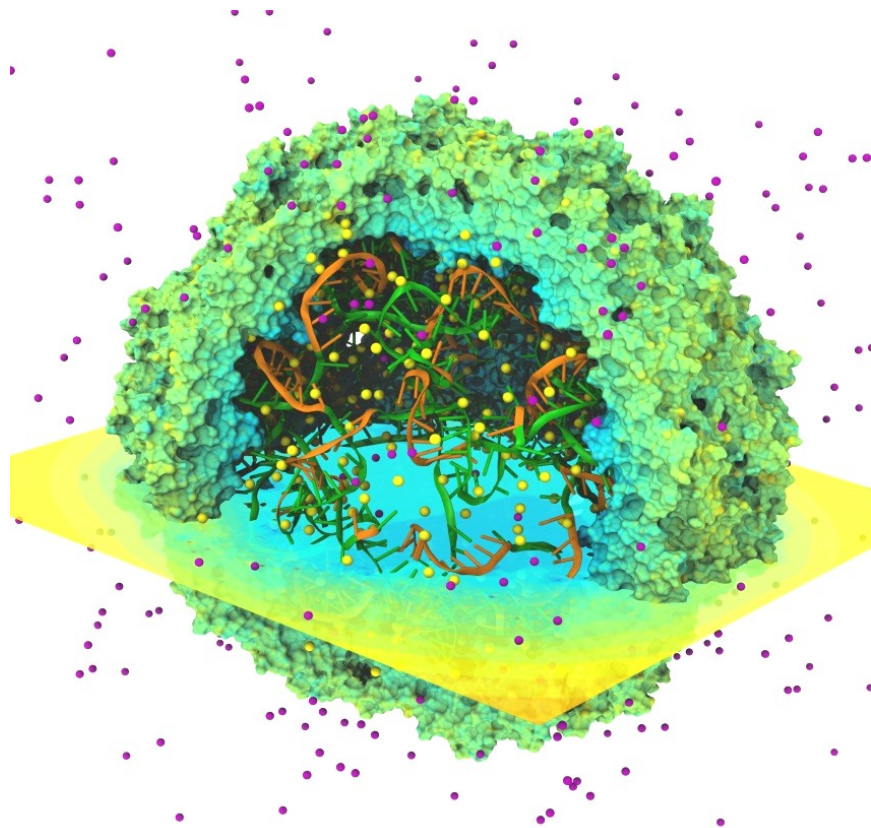
PME electrostatic potential contour for a helicase on a volumetric slice plane

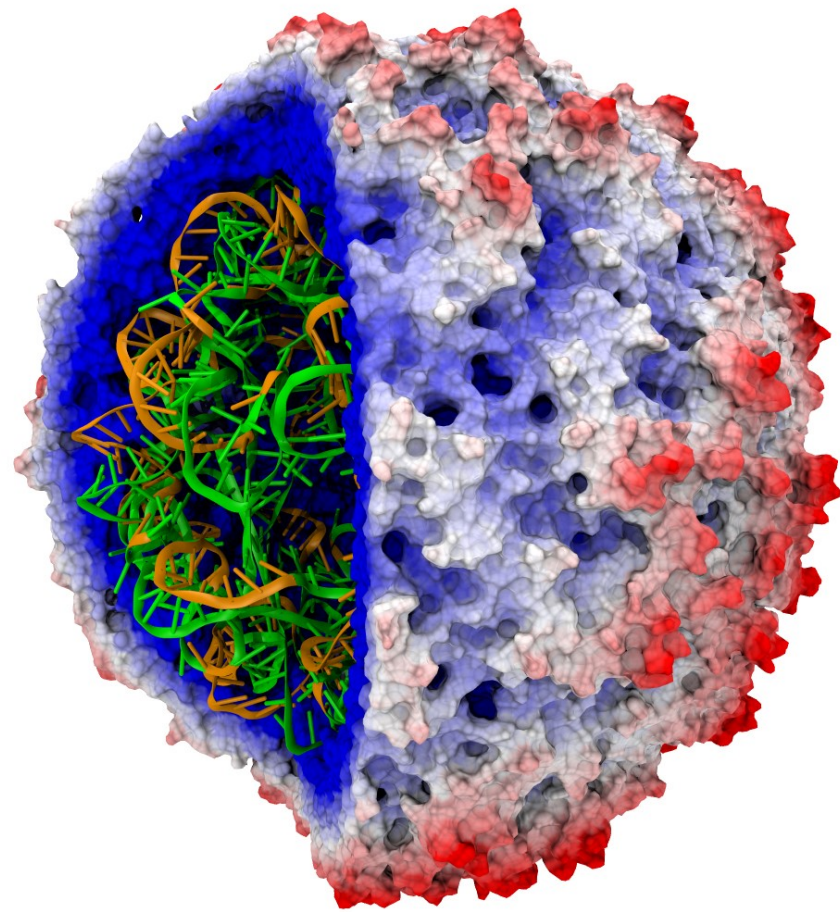
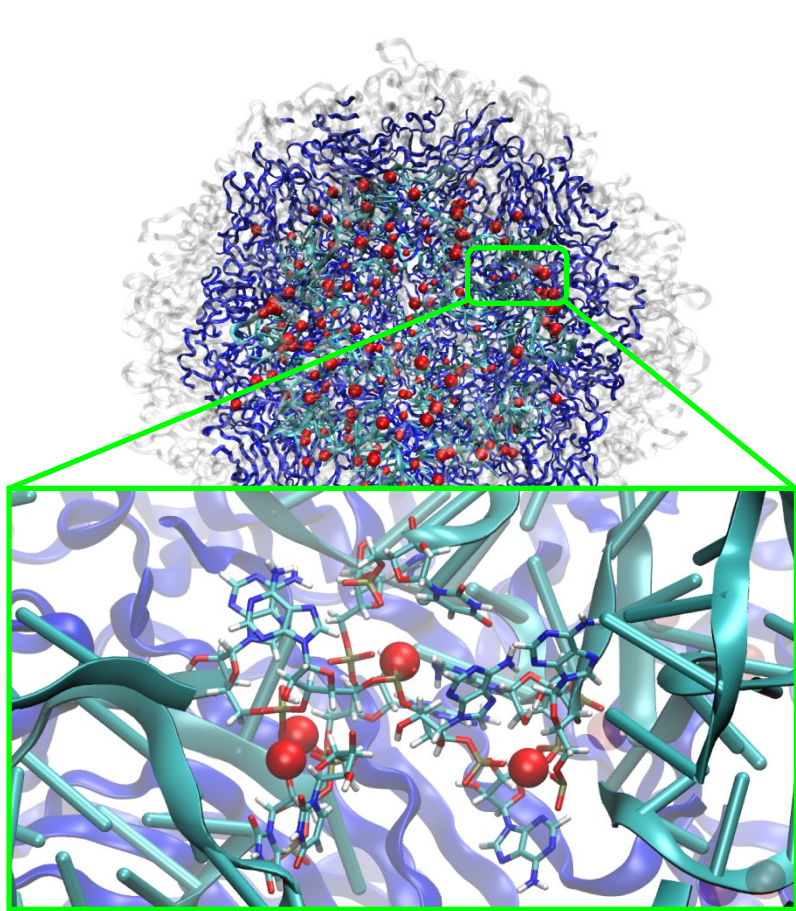
CheA kinase PCA: first principal component porcupine plot



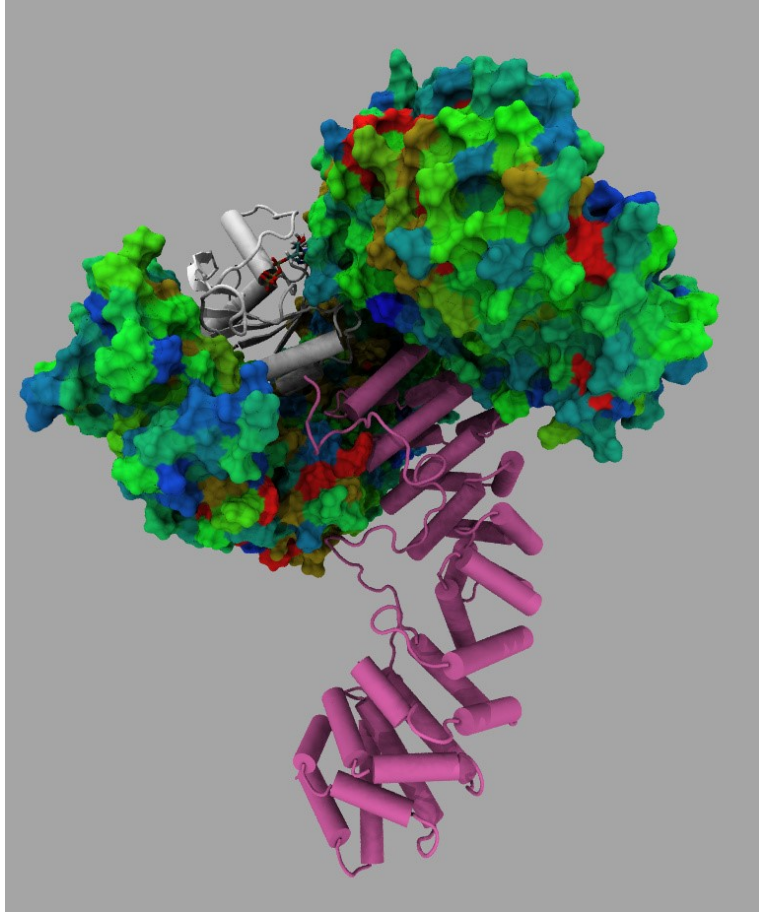
Molecular Visualization Challenges

- Geometrically complex scenes
- Often show a mix of structural and spatial properties
- Time varying!
- Spatial relationships important to see clearly: depth cueing, shadows, AO helpful

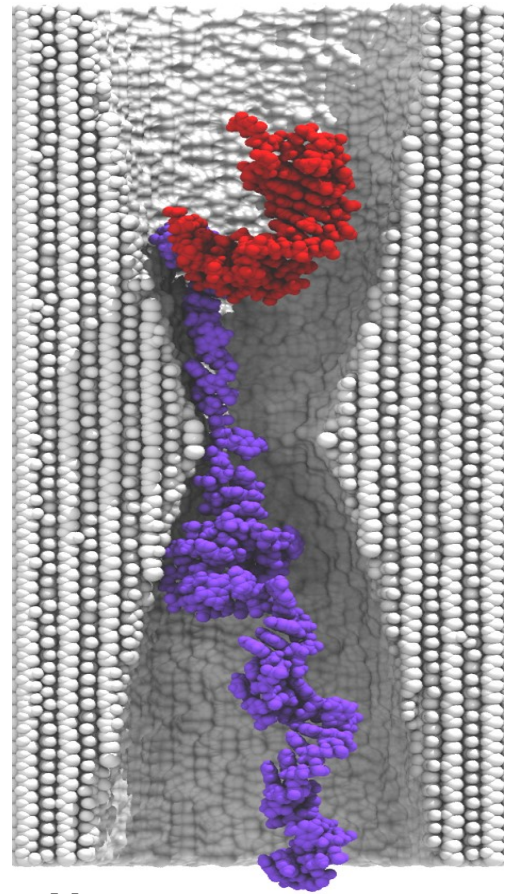




Satellite Tobacco Mosaic Virus



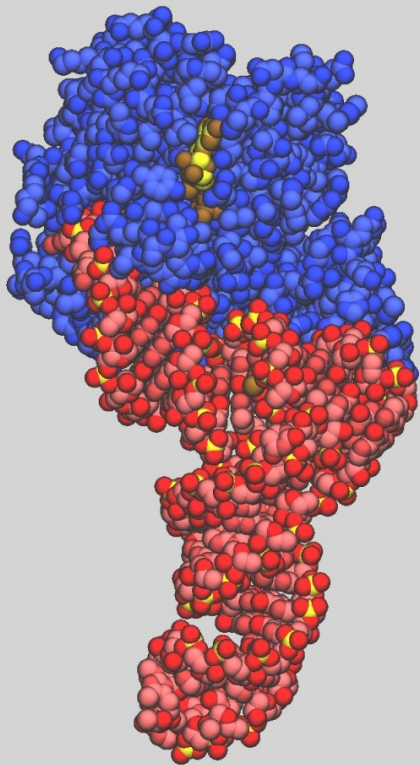
Exportin Cse1p



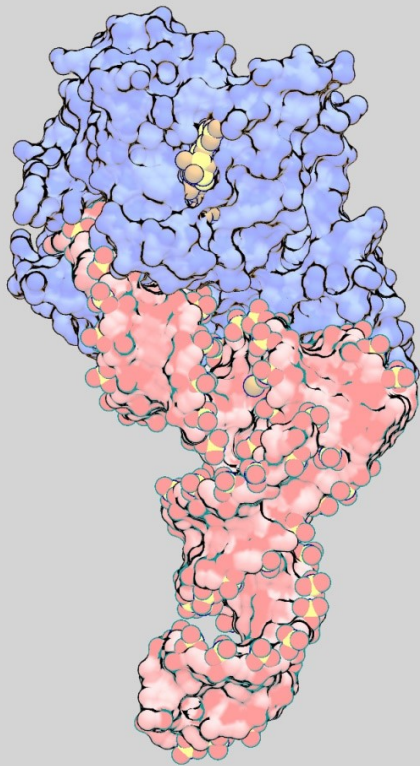
Nanopore

VMD Shading Comparison: EF-Tu

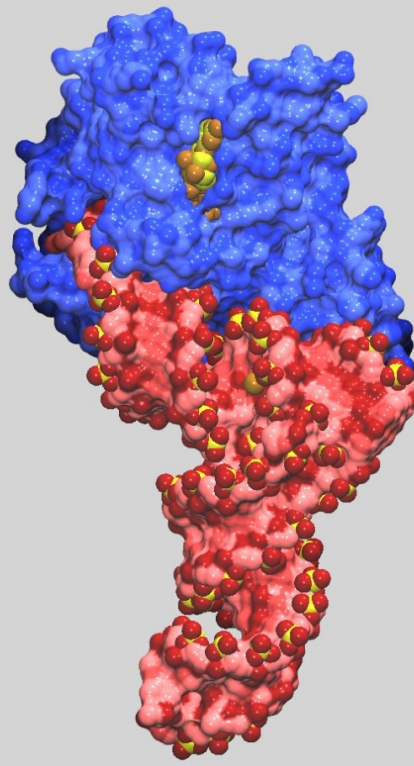
Outline
Shader



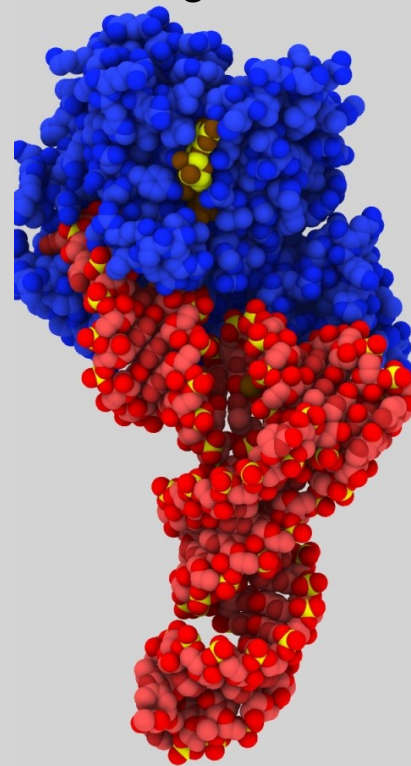
“Goodsell”
Shader



Glossy
Shader



Ray Tracing:
Ambient Occlusion,
Shadowing



VMD Scripting

VMD Scripting

- Built-in Tcl interpreter is standard in all builds
- Custom builds can contain Python interpreter too
- Collection of simple VMD scripts:
http://www.ks.uiuc.edu/Research/vmd/script_library/

Atom Selections

```
set ionssel [atomselect top "ions"]
```

```
set watsel [atomselect top "water"]
```

```
set solute [atomselect top "not water and not ions"]
```

```
set numions [$ionssel num]
```

```
set coords [$solute get {x y z}]
```

Built-in “Measure” Analysis Routines

- Provide high-performance routines for common analysis functions
- Operate on atom selections, sometimes over a frame range

```
set com [measure center $sel weight mass]
```

```
set avpos [measure avpos $sel first 0 last 999 step 1]
```

```
set c [measure contacts 1.0 $sel1 $sel2]
```

```
set txmat [measure fit $sel1 $sel2]
```

```
$sel2 move $txmat
```

```
# Print RMSD between two selections between the first timestep and each
# later timestep for the given molecule id (default: top)
proc print_rmsd_trajectory {{mol top}} {
    set reference [atomselect $mol "protein" frame 0]           # use frame 0 for the reference
    set compare   [atomselect $mol "protein"]                 # the frame being compared
    set numsteps  [molinfo $mol get numframes]                # query number of trajectory frames
    for {set frame 0} {$frame < $numsteps} {incr frame} {
        $compare frame $frame                                # set the correct comparison frame
        set transmat [measure fit $compare $reference]        # compute the transformation
        $compare move $transmat                               # align comparison to reference
        set rmsd [measure rmsd $compare $reference]          # compute the RMSD
        puts "RMSD of $frame is $rmsd"                       # print the RMSD
    }
}
```

Trajectory Analysis and Visualization

Molecular Dynamics Trajectory Analysis

- MD simulations sample femtosecond timescales
- Millions of timesteps stored per trajectory
- Dynamics of biomolecular complexes are main interest, but **solvent often accounts for half or more of the simulation content**

Skip I/O for regions of bulk solvent where possible [1]

- Modern MD tools, e.g., VMD, NAMD, LAMMPS, HOOMD, employ extensive **embedded scripting (Python, Tcl, etc)** to permit simulation preparation, custom simulation protocols, **analysis, and visualization**
- Unified collective variables module allows identical analytical computations to be performed within LAMMPS, NAMD, and VMD, during **pre-simulation modeling, in-situ, and post-hoc [2]**

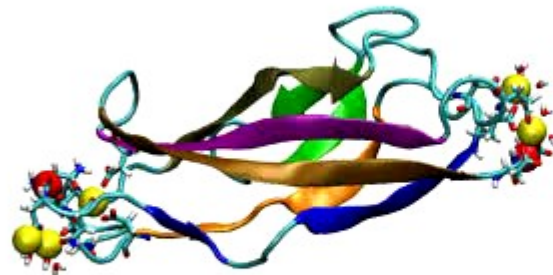
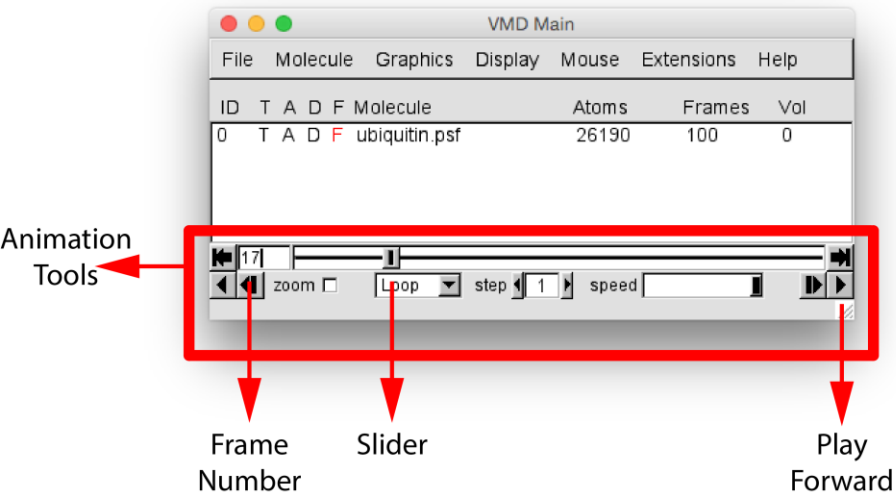
[1] Immersive Out-of-Core Visualization of Large-Size and Long-Timescale Molecular Dynamics

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.

[2] Using collective variables to drive molecular dynamics simulations. G. Fiorin, M. L. Klein, and J. Hénin. *Molecular Physics*, 111:22-23, 3345-3362, 2013.

Visualization of MD Trajectories

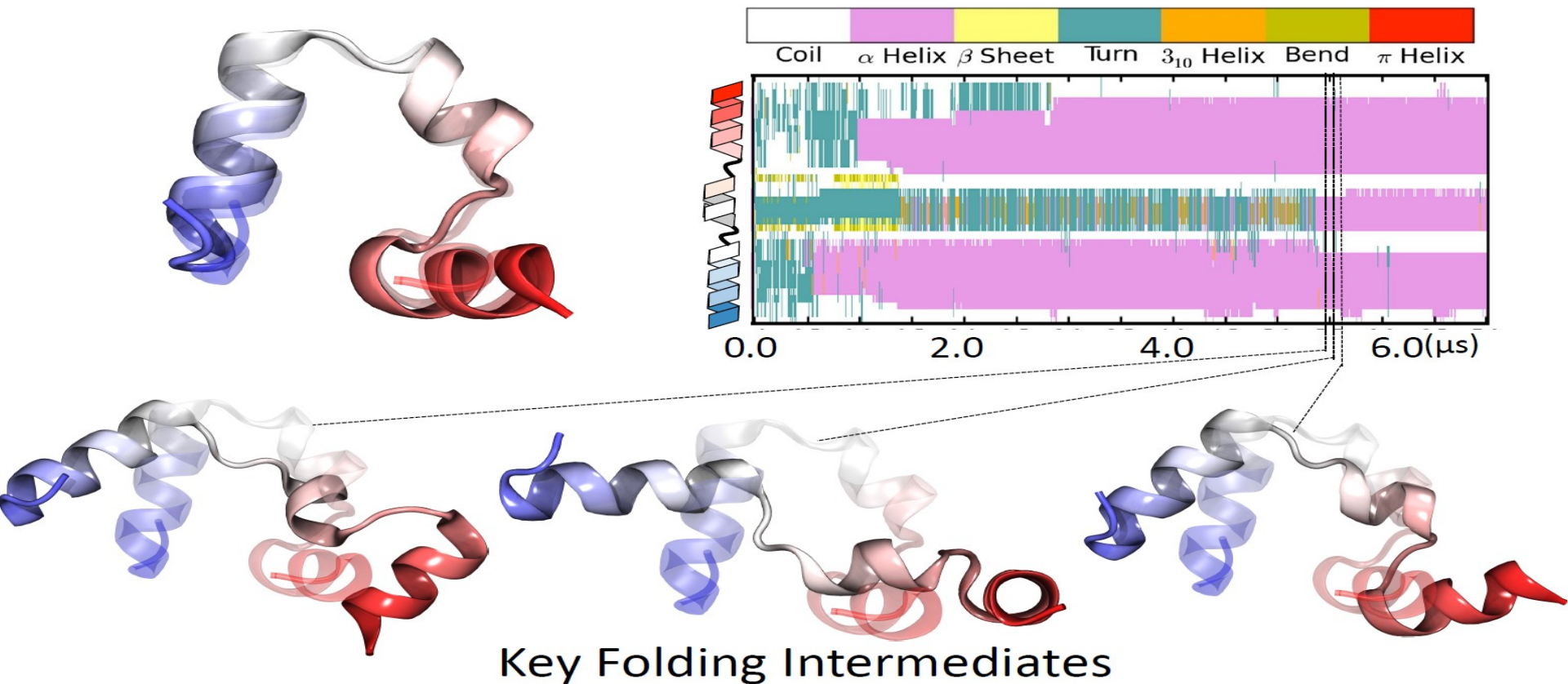
- Allow researchers study trajectories by analyzing force profiles, energies, structural changes, etc.
- **Visualization selections, graphics, structure properties, colors can all be recomputed for each trajectory timestep!**



Folding Dynamics of Villin Headpiece Unveiled

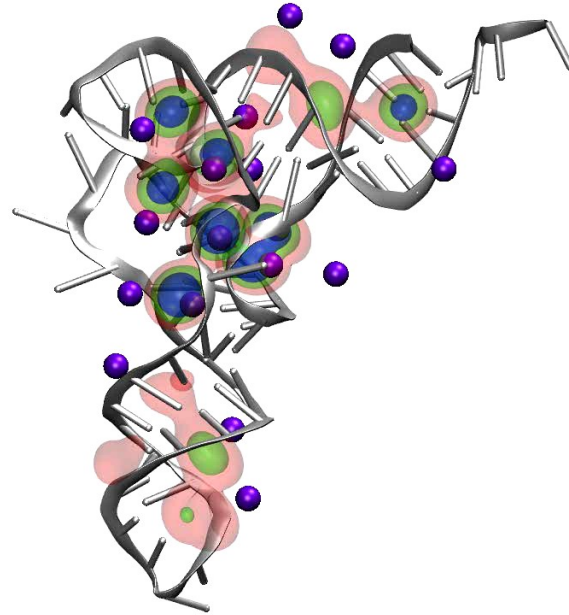
6.9 μ s folding simulation of 30K atoms: 380GB trajectory

Schulten et al. *Biophys J* 94:L75, 2008, 97: 2009



Time-Averaged Volumetric Properties

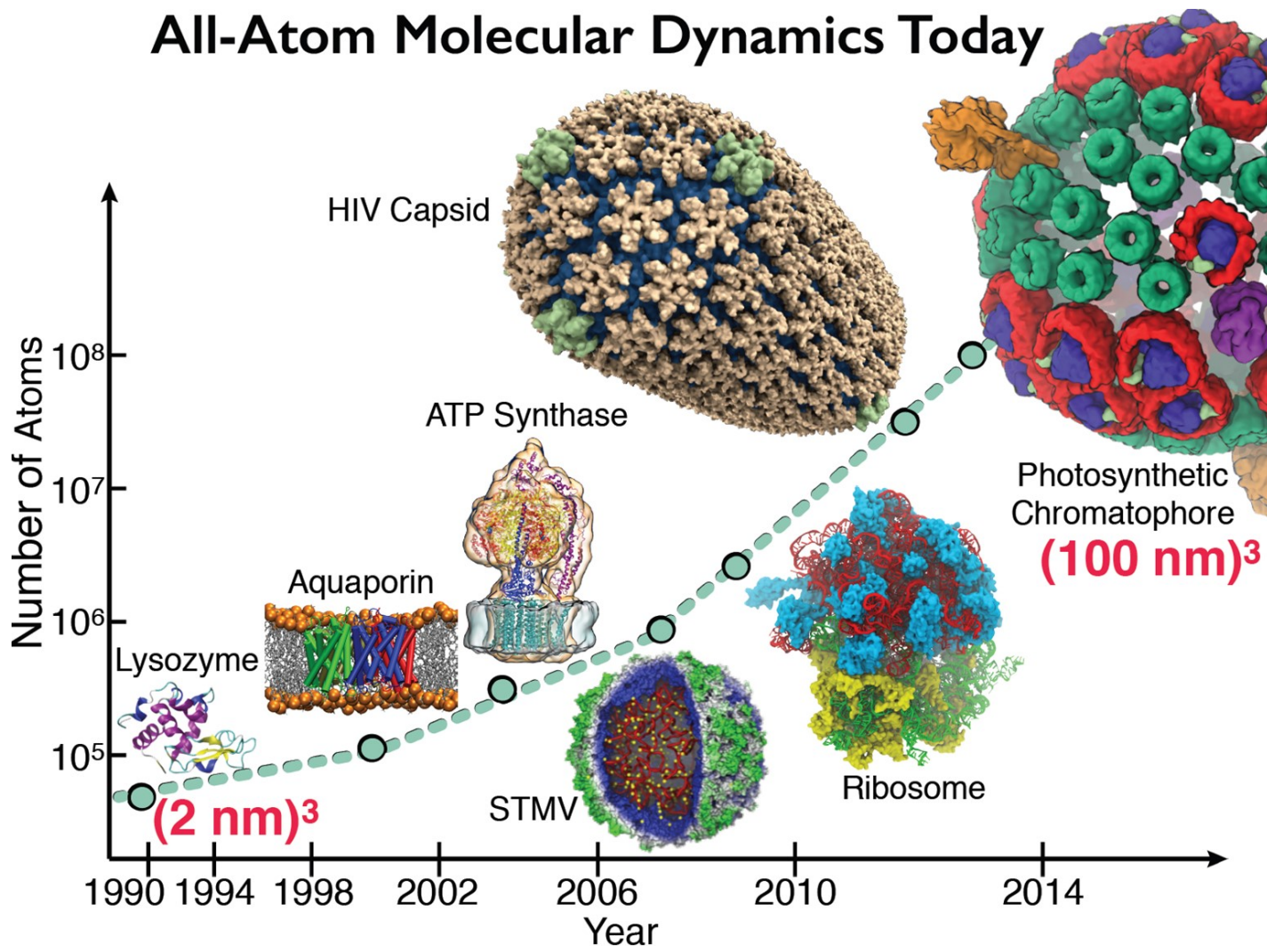
- Compute density, distance, occupancy, potential maps for a frame or averaged over a trajectory
- Example: display binding sites for diffusively bound ions as probability density isosurfaces



**tRNA magnesium ion occupancy:
VMD volmap plugin**

Large System Analysis and Visualization

All-Atom Molecular Dynamics Today



VMD Petascale Visualization and Analysis

- Combination of growing system sizes and timescales of simulation trajectories poses a major data size challenge for molecular visualization and analysis
- Parallel I/O rates up to **275 GB/sec** on 8192 Cray XE6 nodes – can read in **231 TB in 15 minutes!**
- Analyze/visualize large trajectories **too large to transfer off-site:**
 - User-defined parallel analysis operations, data types
 - Parallel rendering, movie making
- Supports GPU-accelerated compute nodes for both visualization and analysis tasks:
 - GPU accelerated trajectory analysis w/ CUDA
 - OpenGL and GPU ray tracing for visualization and movie rendering

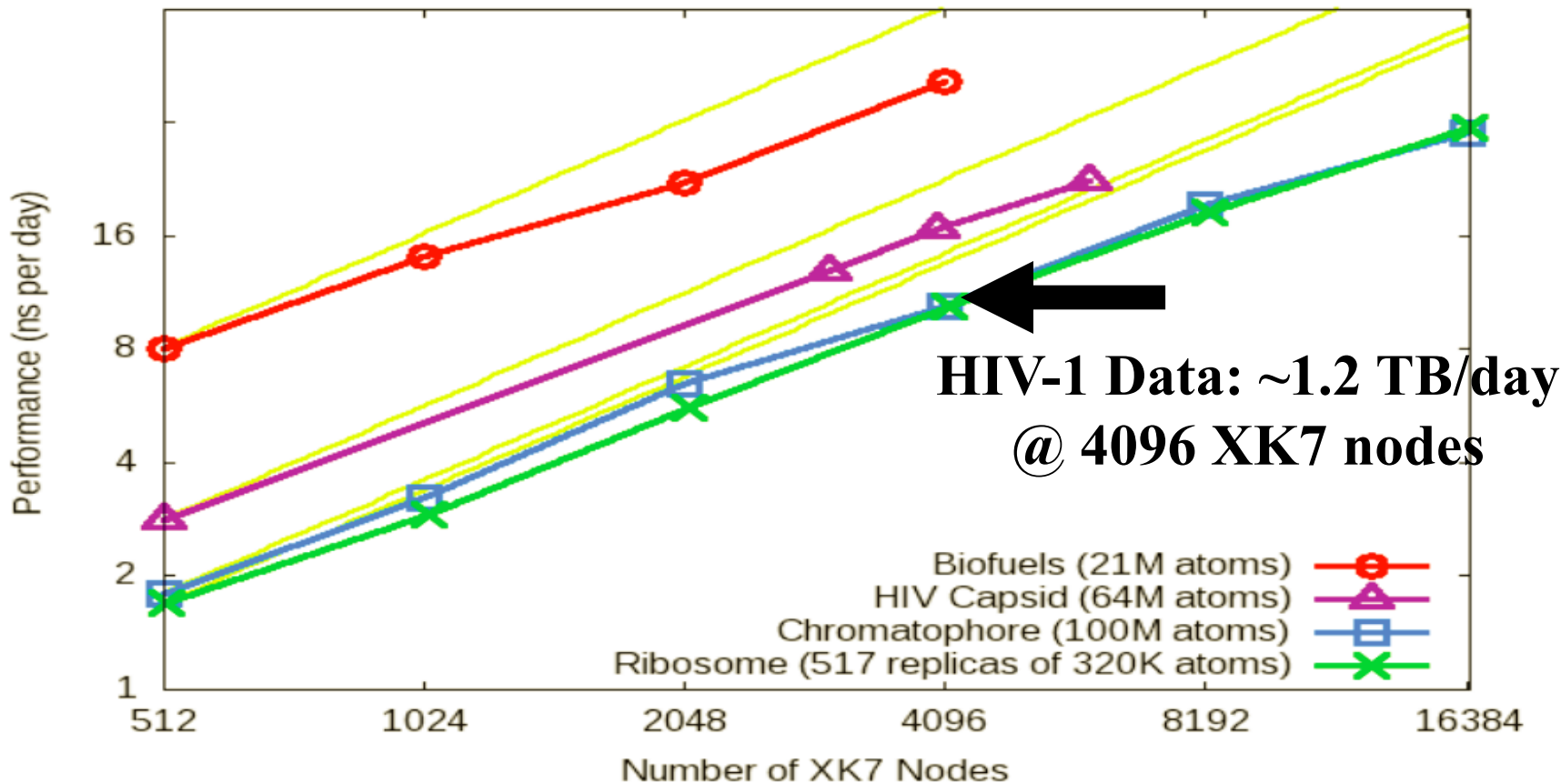


NCSA Blue Waters Hybrid Cray XE6 / XK7
22,640 XE6 dual-Opteron CPU nodes
4,224 XK7 nodes w/ Telsa K20X GPUs

Parallel VMD currently available on:
ORNL Summit and Titan, NCSA Blue
Waters, IU Big Red II, CSCS Piz Daint,
many similar systems

NAMD Titan XK7 Performance August 2013

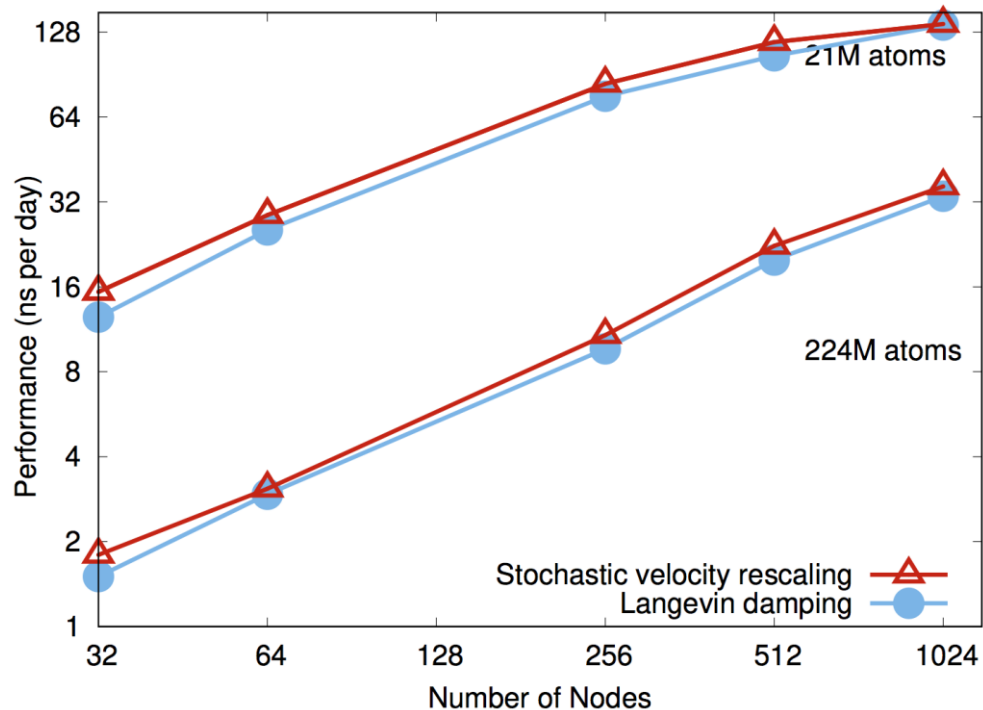
NAMD on Titan Cray XK7 (2fs timestep with PME)



Petascale Molecular Dynamics I/O and Storage Challenges

- NAMD simulations can produce up to **10TB/day @ 1024 nodes (~20%) of ORNL Summit**, more as optimizations raise NAMD performance further
- Petascale science campaigns require **months of simulation runs**
- **Long-term storage of large-fractional petabytes impractical**
- **Historical “download output files for analysis and visualization” approach is a non-starter at this scale**
- Demands visualization and analysis operate on the data **in-place** on the HPC system, **whether *post-hoc, in-transit, or in-situ***
- Analyses must **identify salient features of structure, dynamics**, cull data that don't contribute to biomolecular processes of interest

NAMD on Summit, May 2018



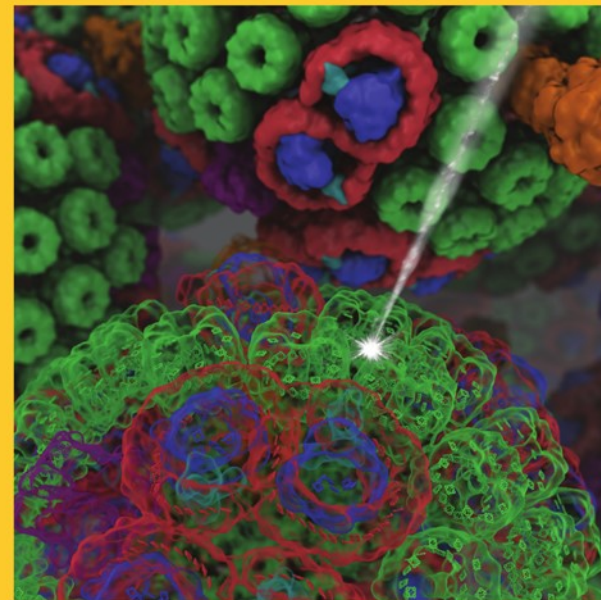
NAMD simulations can generate up to 10TB of output per day on 20% of Summit

IPCB

APRIL 20, 2017
VOLUME 121
NUMBER 15
pubs.acs.org/JPCB

THE JOURNAL OF
**PHYSICAL
CHEMISTRY**

B



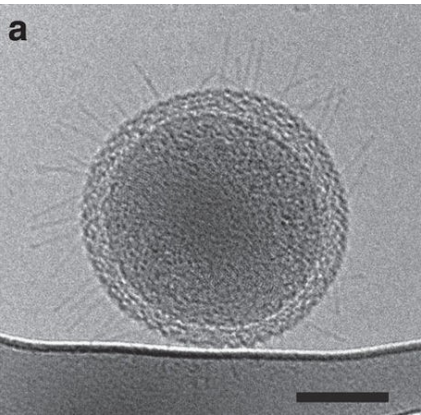
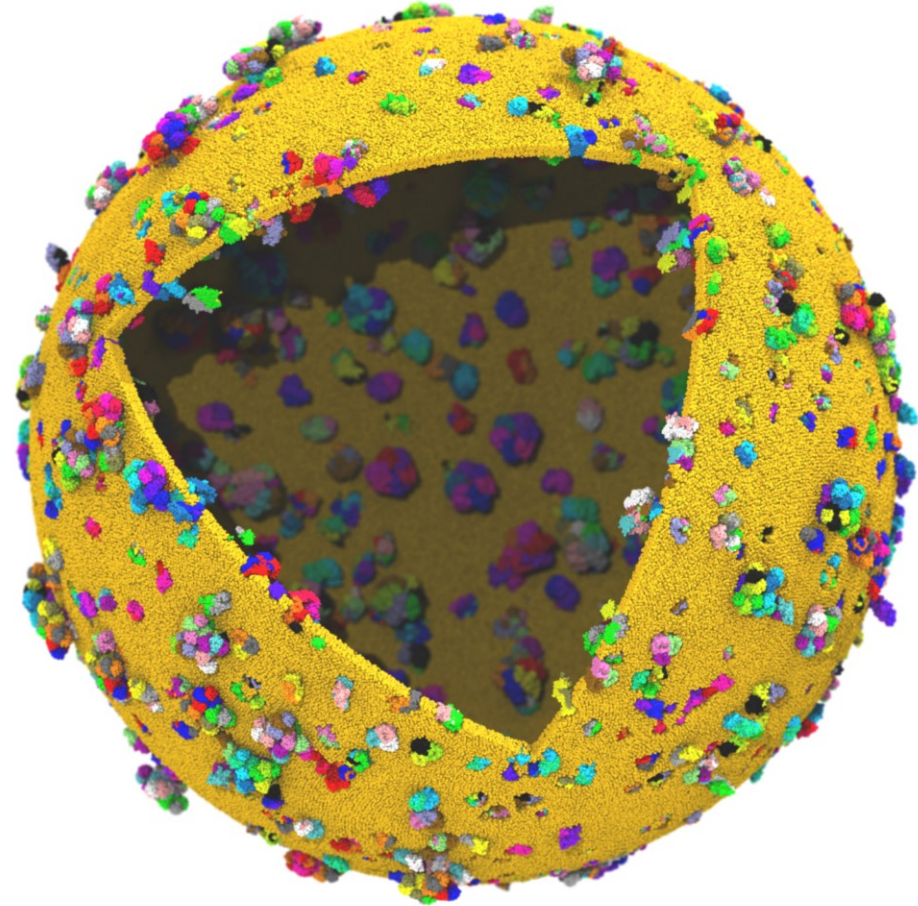
KLAUS SCHULTEN MEMORIAL ISSUE

ACS Publications
Most Trusted. Most Cited. Most Read.

www.acs.org

Next Generation: Simulating a Proto-Cell

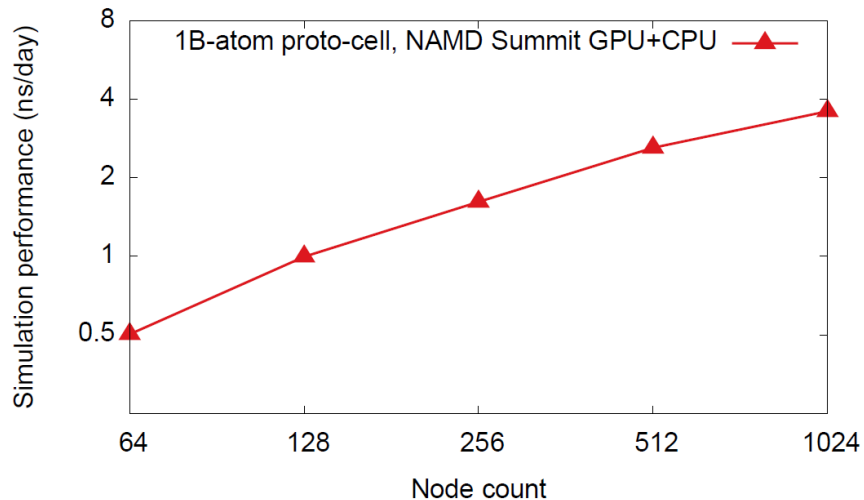
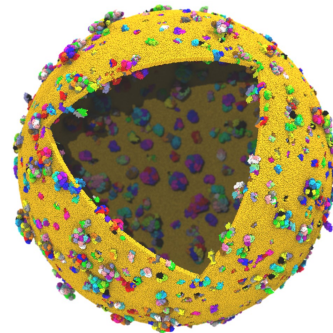
- Emulate aspects of the *Mycoplasma mycoides* bacterium
- 200nm diameter
- ~1 billion atoms w/ solvent
- ~1400 proteins in membrane



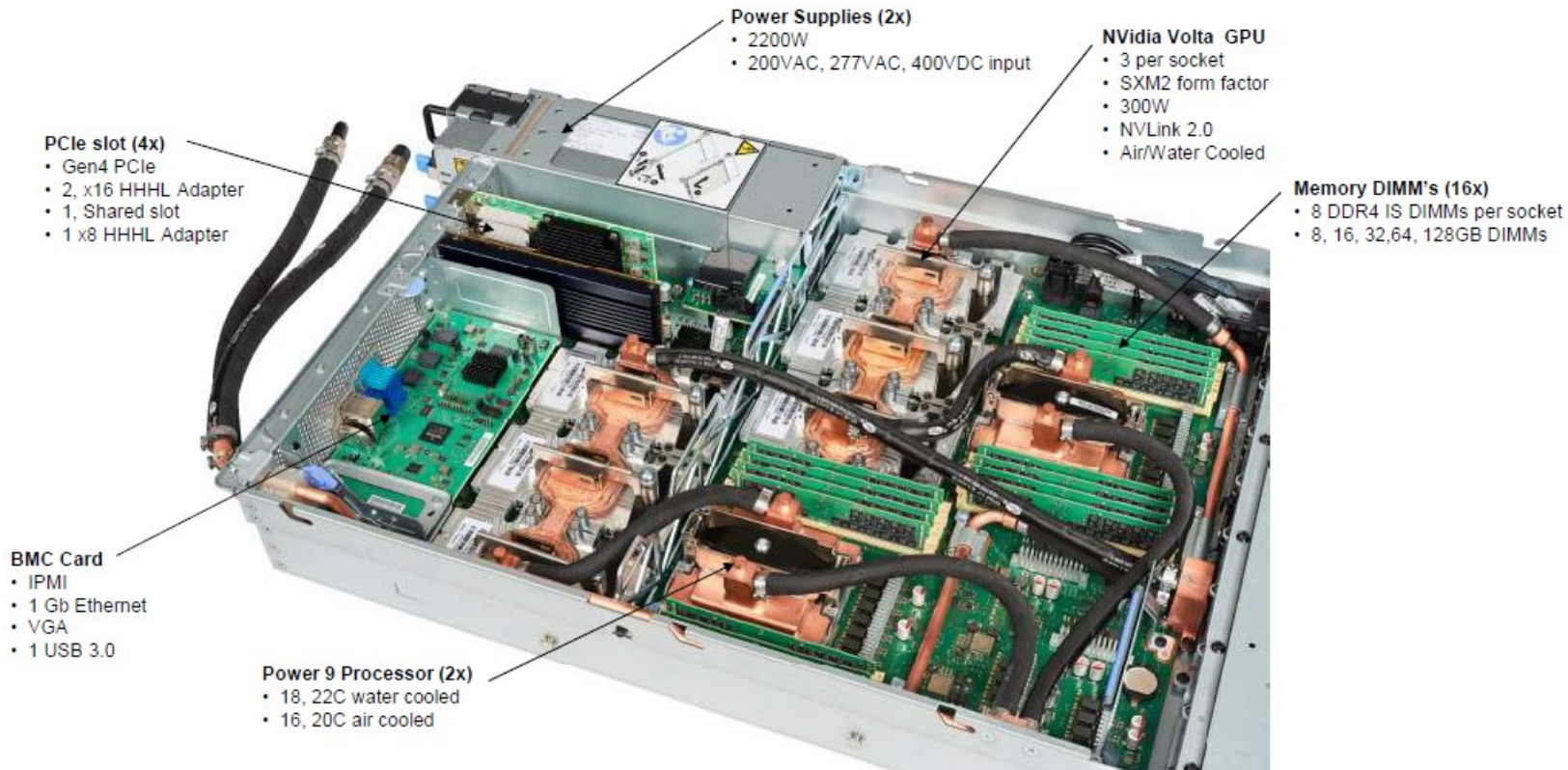
Cryo-ET image of ultra-small bacteria (scale bar 100nm)
Luef et al. Nature Comm., 6:6372, 2015.

Proto-Cell Data Challenges

- **1B-atom proto-cell requires nodes with more than TB RAM to build complete model...**
- **1B-atom proto-cell binary structure file: 63GB**
- **Trajectory frame atomic coordinates: 12GB, 1.2TB/ns of simulation (1 frame per 10ps)**
- **Routine modeling and visualization tasks are a big challenge at this scale**
 - **Models contain thousands of atomic-detail components** that must work together in harmony
 - **Exploit persistent memory technologies** to enable “instant on” operation on massive cell-scale models – eliminate several minutes of startup during analysis/visualization of known structure
 - **Sparse output of results at multiple timescales** will help ameliorate visualization and analysis I/O
 - **Data quantization, compression, APIs like ZFP**

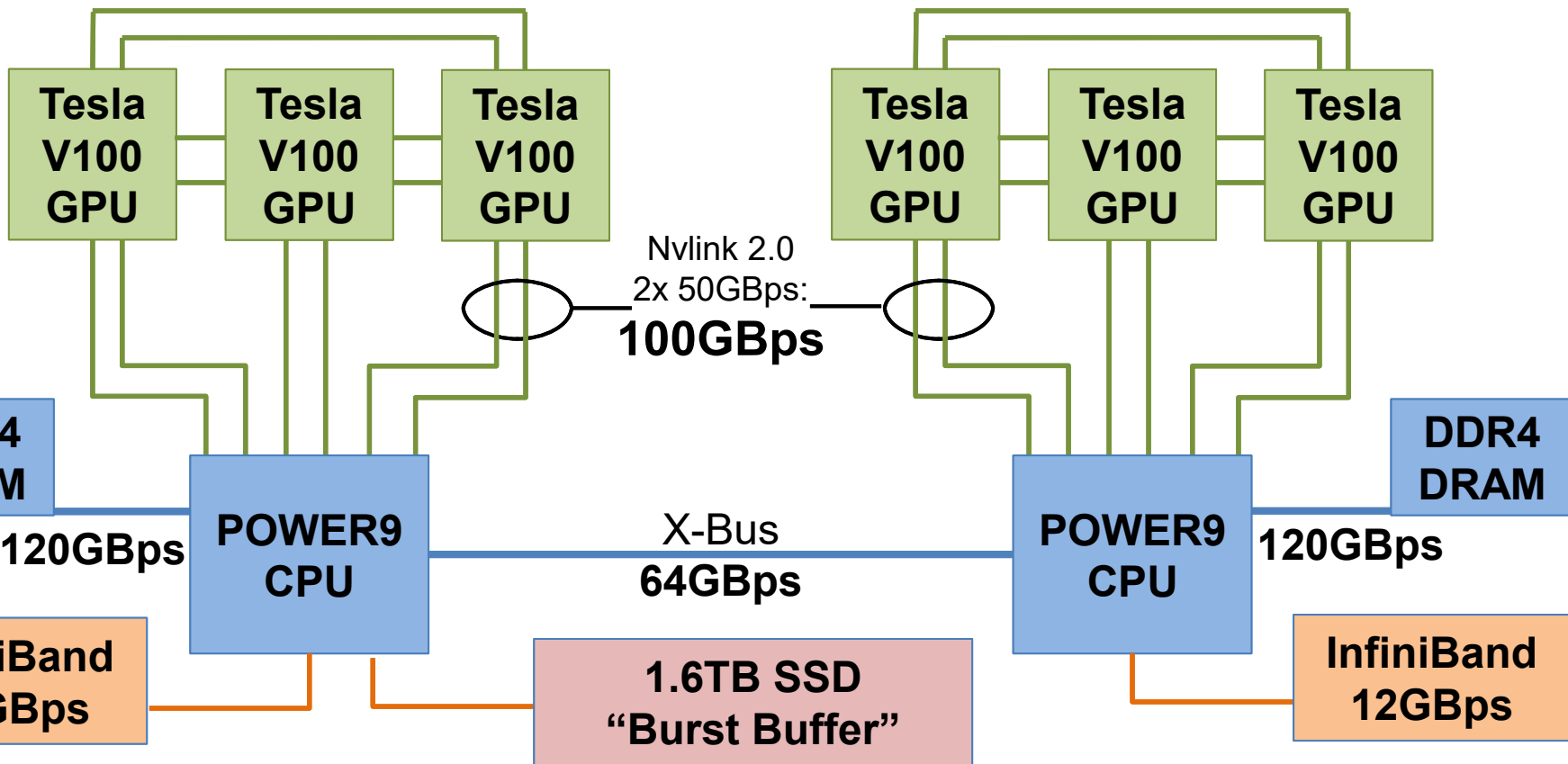


IBM AC922 w/ 6 GPUs



IBM AC922 Summit Node

3 GPUs Per CPU Socket

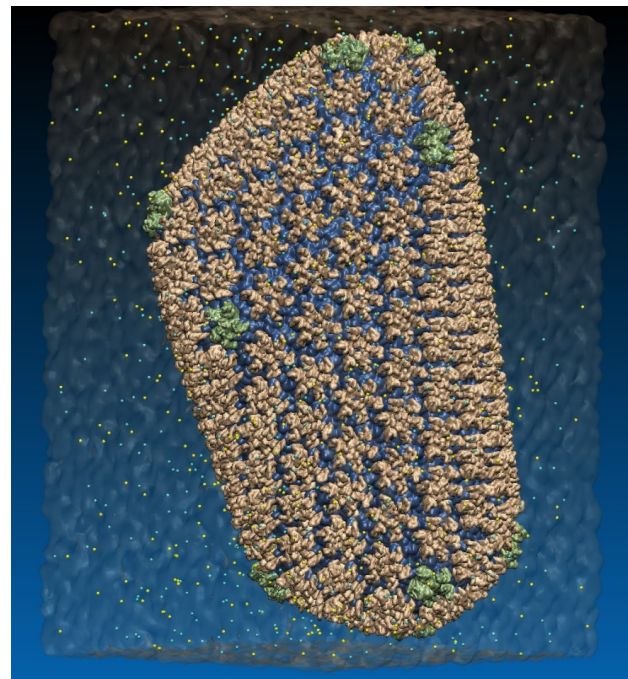
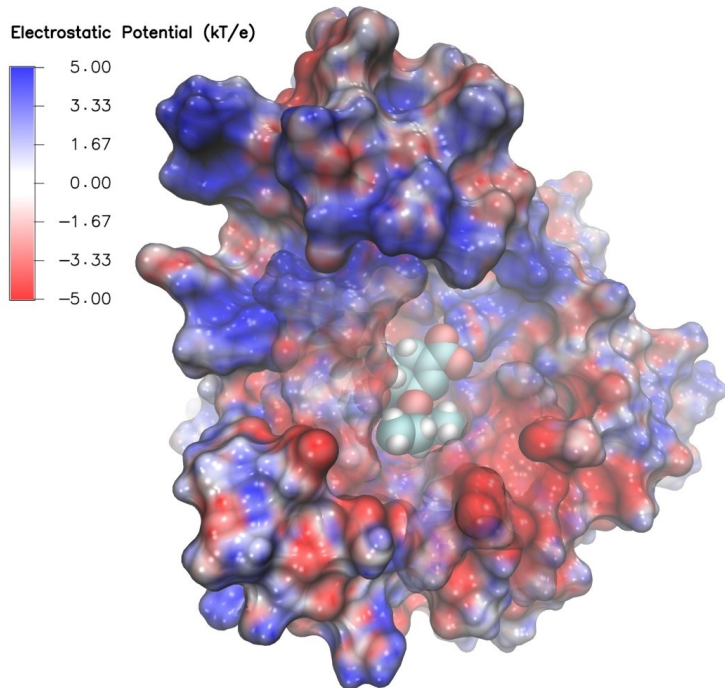


VMD Off-Screen Rendering w/ EGL

- Containers+Cloud+Workstations with recent NVIDIA drivers
- VMD on HPC systems w/ latest GPUs:
 - Cray XC50, CSCS Piz Daint
 - **ORNL Summit in progress now**
 - IBM OpenPOWER, drivers 375.66 and later support both GLX and EGL



VMD EGL Rendering: Supports full VMD GLSL shading features Vulkan support coming soon...



Swine Flu A/H1N1 neuraminidase bound to Tamiflu

64M atom HIV-1 capsid simulation

High Performance Molecular Visualization: In-Situ and Parallel Rendering with EGL.

J. E. Stone, P. Messmer, R. Sisneros, and K. Schulten. High Performance Data Analysis and Visualization Workshop, IEEE IPDPSW, pp. 1014-1023, 2016.

Running VMD on Distributed Memory Clouds, Clusters, and Supercomputers with MPI

Using VMD MPI Builds

- **See “Running VMD on Supercomputers” section in the VMD tutorial**
- Run one MPI rank (VMD process) per compute node
 - Each rank uses all CPU cores and all GPU accelerators they find
 - GPUs are shared between graphics/visualization and computing tasks
- Parallel VMD Scripting APIs:
 - “parallel” script subcommands, e.g., “parallel for ...”:
 - “nodename”, “noderank”, “nodecount”, “barrier”, “allgather”, “allreduce”, “for”
 - Available even in non-MPI builds, for single-node script compatibility
- Work scheduling:
 - Intra-node work scheduled via multithreading, CUDA, etc.
 - Inter-node distributed memory work scheduling of user analysis/viz scripts is provided by built-in load balancing implementation for VMD

Example MPI Session, CSCS Piz Daint

```
stonej@daint103> srun -C gpu -n 256 --ntasks-per-node=1 \  
/users/stonej/local/bin/vmd193 -dispdev text -e rendermovie.tcl
```

Info) VMD for CRAY_XC, version 1.9.3 (December 15, 2016)

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 256 VMD MPI nodes containing a total of 6144 CPUs and 256 GPUs:

Info) 0: 24 CPUs, 60.8GB (96%) free mem, 1 GPUs, Name: nid03072

Info) 1: 24 CPUs, 60.8GB (96%) free mem, 1 GPUs, Name: nid03073

[...example output omitted...]

Info) 254: 24 CPUs, 60.9GB (96%) free mem, 1 GPUs, Name: nid03376

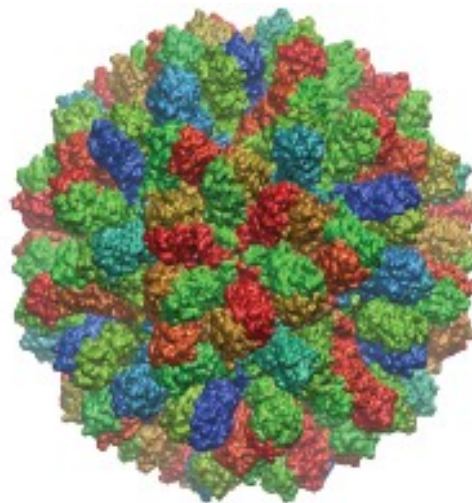
Info) 255: 24 CPUs, 60.9GB (96%) free mem, 1 GPUs, Name: nid03377

Parallel MDFF Cross Correlation Analysis on Cray XK7

Rabbit Hemorrhagic Disease Virus (RHDV)

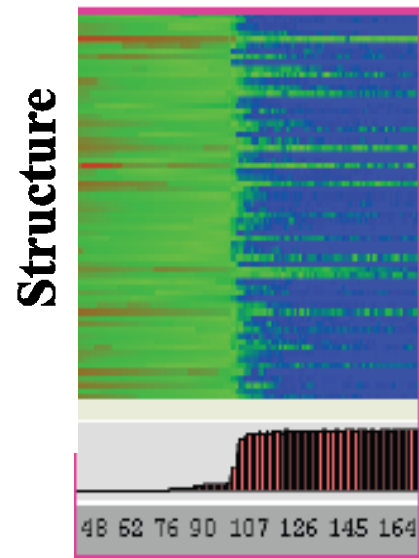
Traj. frames	10,000
Structure component selections	720
Single-node XK7 (projected)	336 hours (14 days)
128-node XK7	3.2 hours 105x speedup
2048-node XK7	19.5 minutes 1035x speedup

Calculation of 7M CCs would take **5 years** using serial CPU algorithm!



**RHDV colored
by relative CC**

Relative CC
-0.0032 0.02



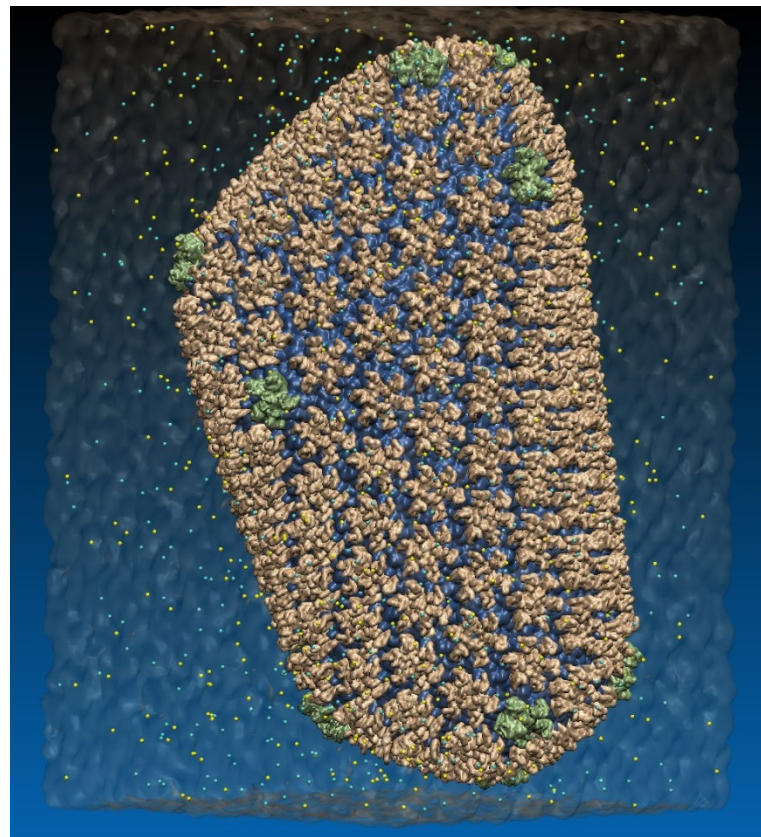
Time

VMD EGL Performance on Amazon EC2 Cloud

MPI Ranks	EC2 “G2.8xlarge” GPU Instances	HIV-1 movie rendering time (sec), (I/O %) 3840x2160 resolution
1	1	626s (10% I/O)
2	1	347s (19% I/O)
4	1	221s (31% I/O)
8	2	141s (46% I/O)
16	4	107s (64% I/O)
32	8	90s (76% I/O)

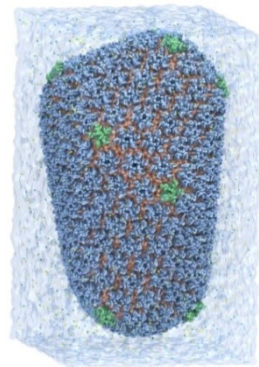
Performance at 32 nodes reaches ~48 FPS

High performance molecular visualization: In-situ and parallel rendering with EGL. J. E. Stone, P. Messmer, R. Sisneros, and K. Schulten. *2016 IEEE International Parallel and Distributed Processing Symposium Workshop (IPDPSW)*, pp. 1014-1023, 2016.



64M atom HIV-1 capsid simulation rendered via EGL

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



New VMD TachyonL-OptiX on XK7 vs. Tachyon on XE6:
K20X GPUs yield **up to twelve times** geom+ray tracing speedup

Ray Tracer Version	Node Type and Count	Script Load	State Load	Geometry + Ray Tracing	Total Time
New TachyonL-OptiX	64 XK7 Tesla K20X GPUs	2 s	39 s	435 s	476 s
New TachyonL-OptiX	128 XK7 Tesla K20X GPUs	3 s	62 s	230 s	295 s
TachyonL-OptiX [1]	64 XK7 Tesla K20X GPUs	2 s	38 s	655 s	695 s
TachyonL-OptiX [1]	128 XK7 Tesla K20X GPUs	4 s	74 s	331 s	410 s
TachyonL-OptiX [1]	256 XK7 Tesla K20X GPUs	7 s	110 s	171 s	288 s
Tachyon [1]	256 XE6 CPUs	7 s	160 s	1,374 s	1,541 s
Tachyon [1]	512 XE6 CPUs	13 s	211 s	808 s	1,032 s

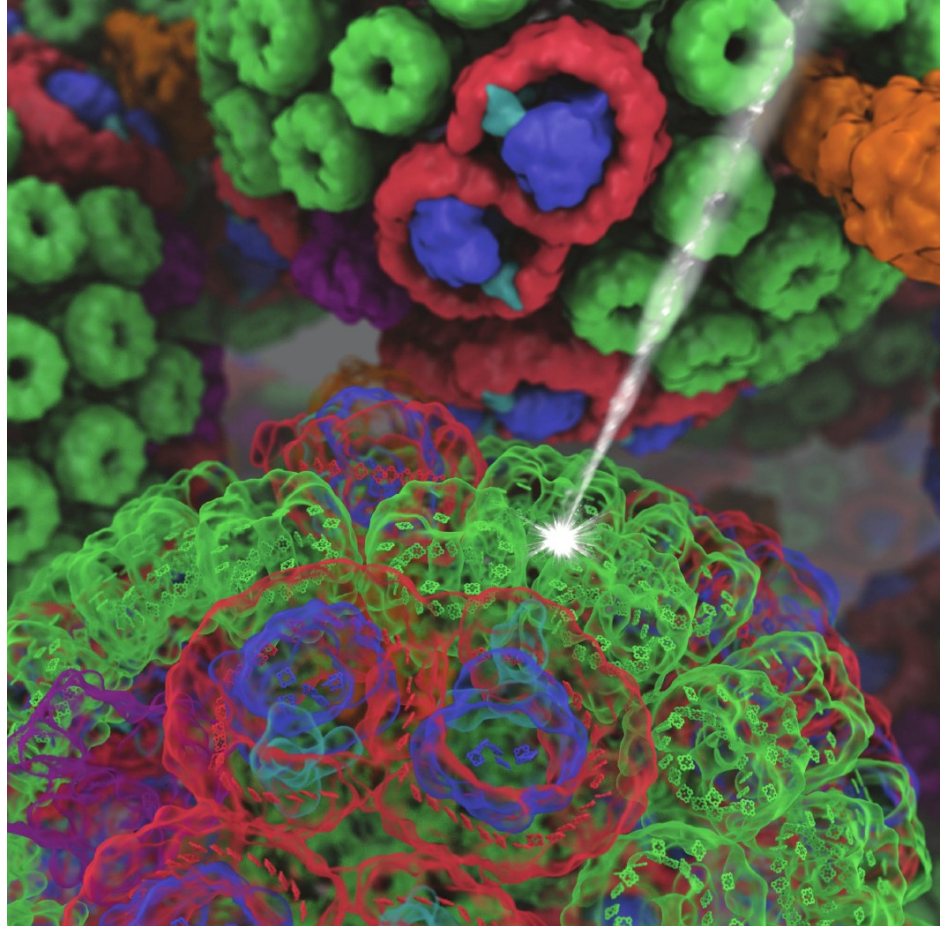
[1] GPU-Accelerated Molecular Visualization on Petascale Supercomputing Platforms.

J. E. Stone, K. L. Vandivort, and K. Schulten. UltraVis'13: Proceedings of the 8th International Workshop on Ultrascale Visualization, pp. 6:1-6:8, 2013.

High Fidelity Ray Tracing for Interactive and Cinematic Visualization

High Fidelity Ray Tracing

- Advanced rendering techniques save scientists time, produce images that are easier to interpret
- Ambient Occlusion, Depth of Field, high quality transparency, instancing,
- **Interactive RT** on laptops, desk, cloud, and **remote supercomputers**
- **Large-scale MPI parallel rendering:** in situ or post hoc visualization tasks
- Stereoscopic panorama and full-dome projections
- **Omnidirectional VR: YouTube, HMDs**
- Built-in ray tracing engines:
 - **Tachyon:** cross-platform RT
 - **NVIDIA OptiX:** GPU RTX-accelerated
 - **Intel OSPRay:** CPU x86-optimized

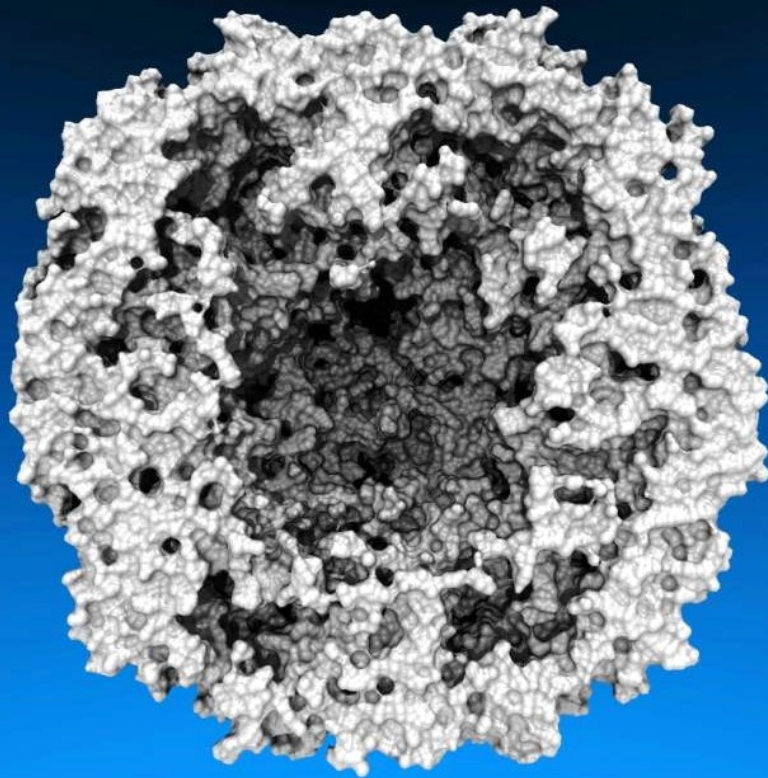
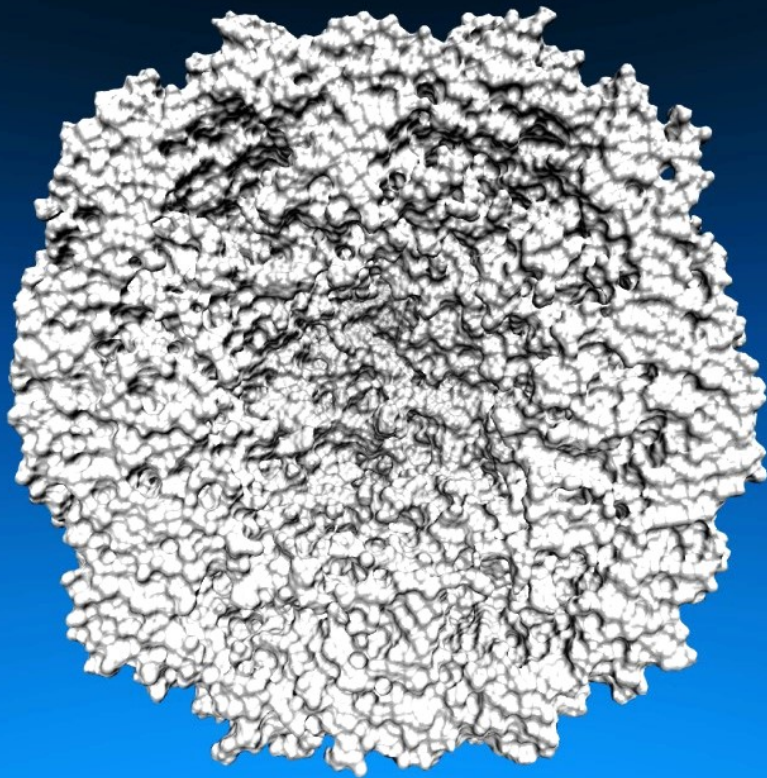


VMD/OptiX all-atom Chromatophore

Lighting Comparison, STMV Capsid

Two lights, no shadows

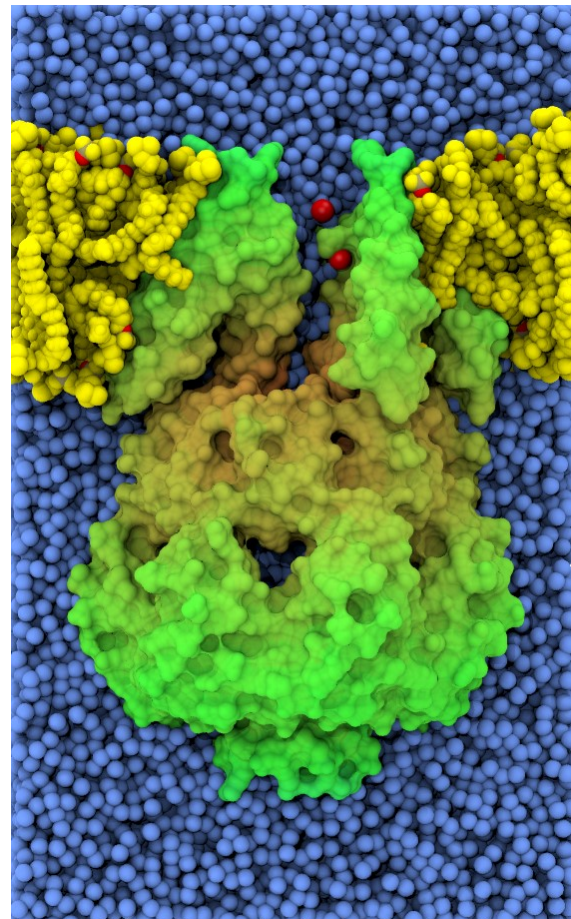
Ambient occlusion + two lights, 144 AO rays/hit



Geometrically Complex Scenes

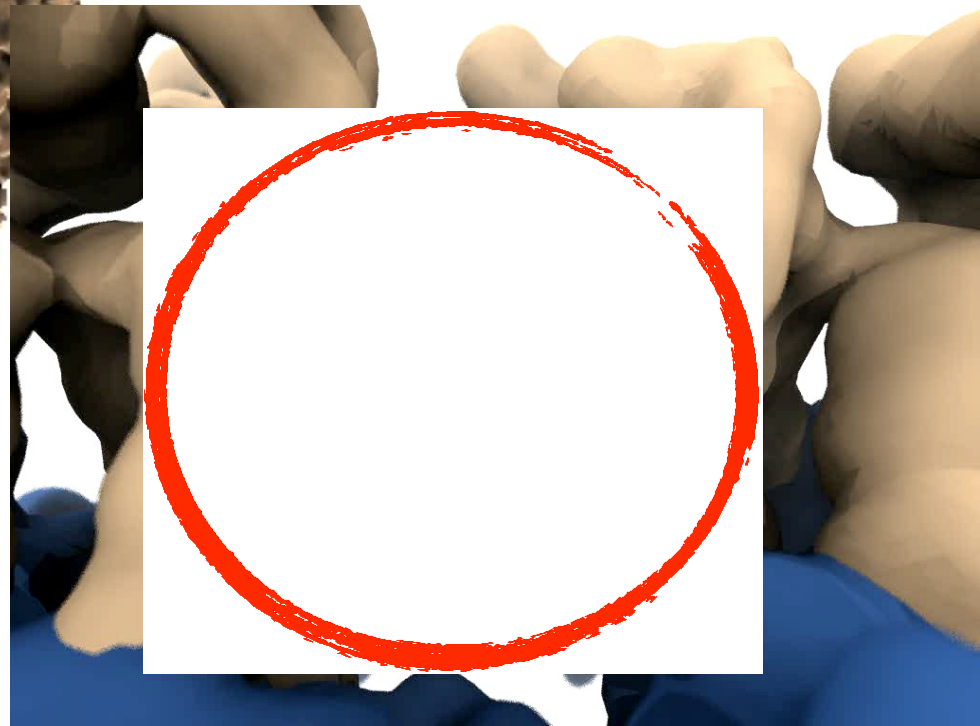
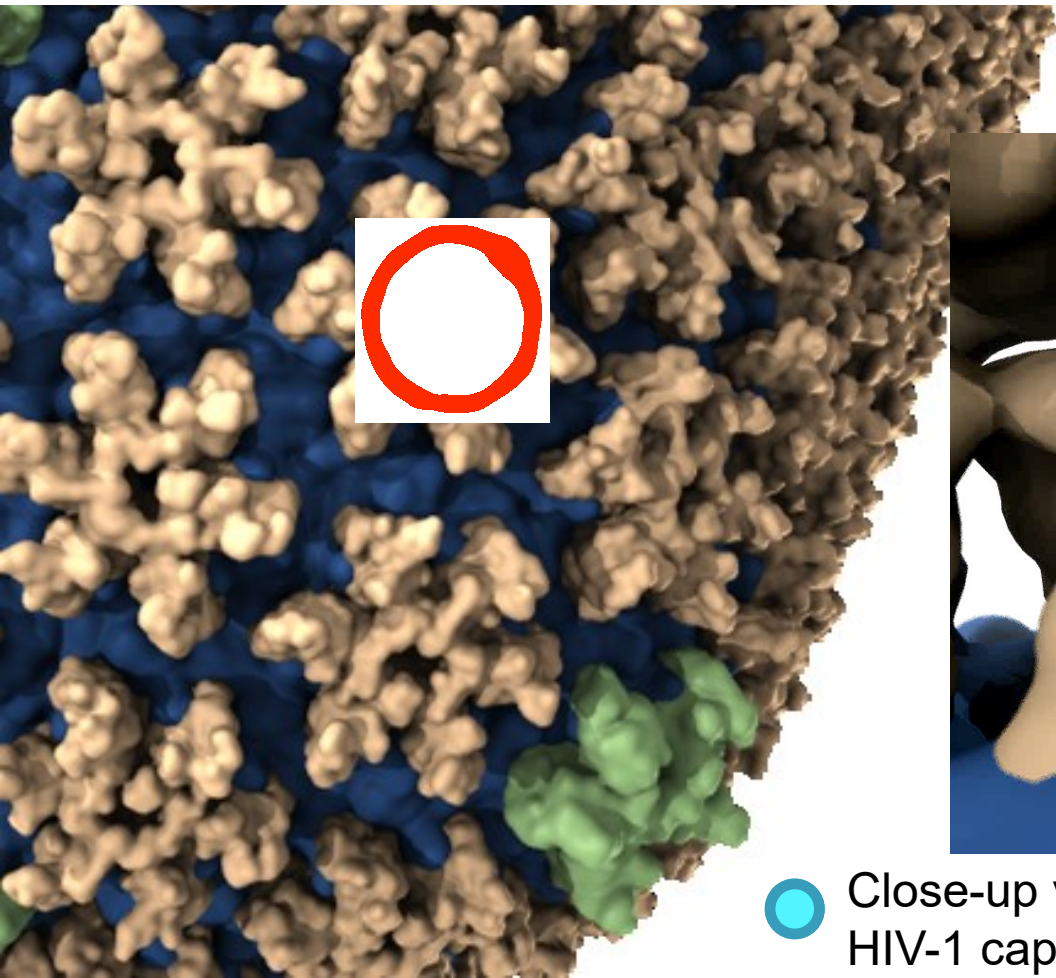
Ray tracing techniques well matched to molecular viz. needs:

- Curved geometry, e.g. spheres, cylinders, toroidal patches, easily supported
- Greatly reduced memory footprint vs. polygonalization
- Runtime scales only moderately with increasing geometric complexity
- Occlusion culling is “free”, RT acceleration algorithms do this and much more

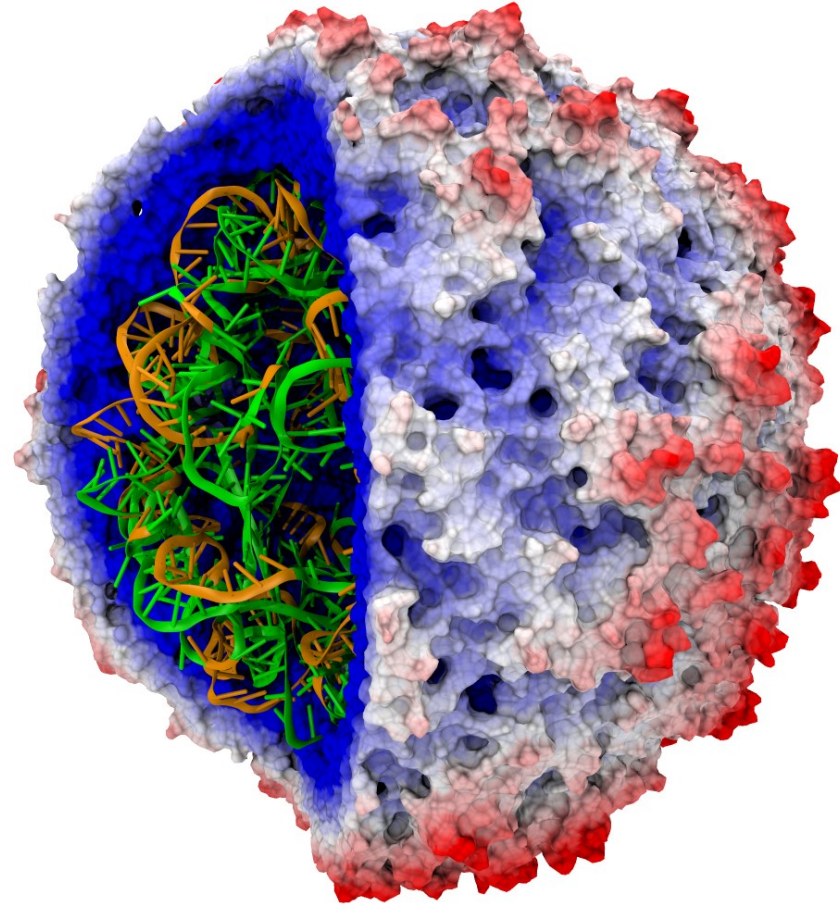
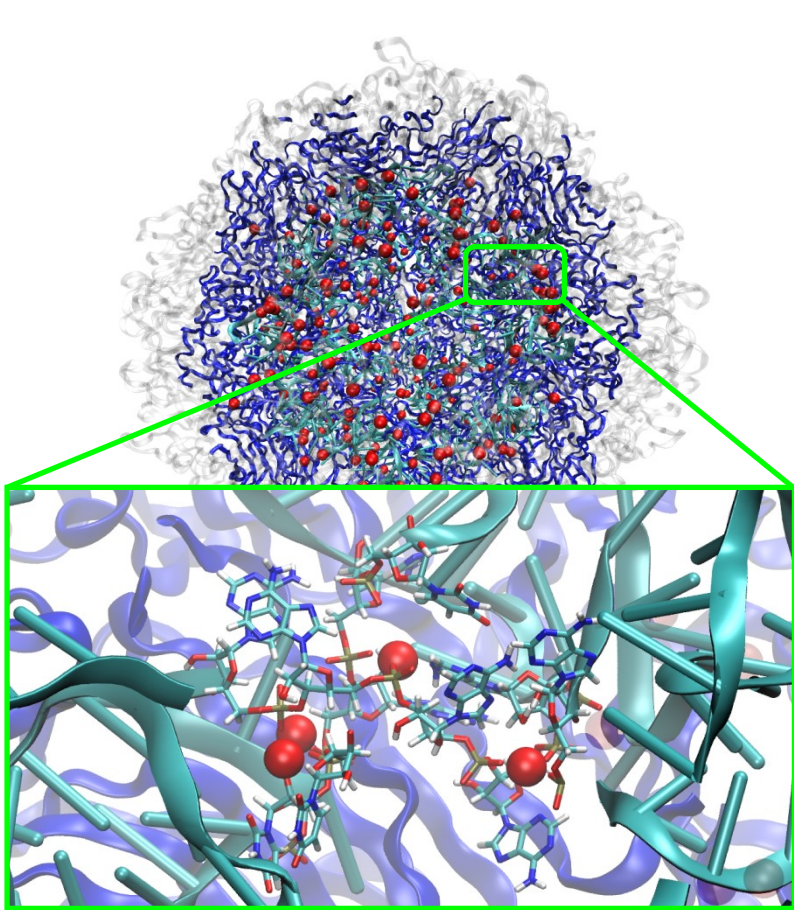


Goal: Intuitive interactive viz. in crowded molecular complexes

Results from 64 M atom, 1 μ s sim!

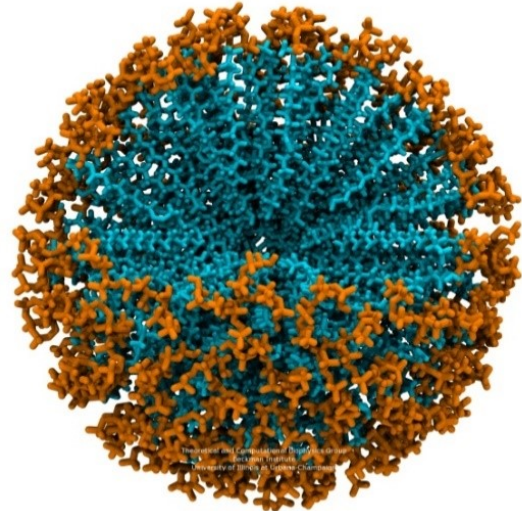
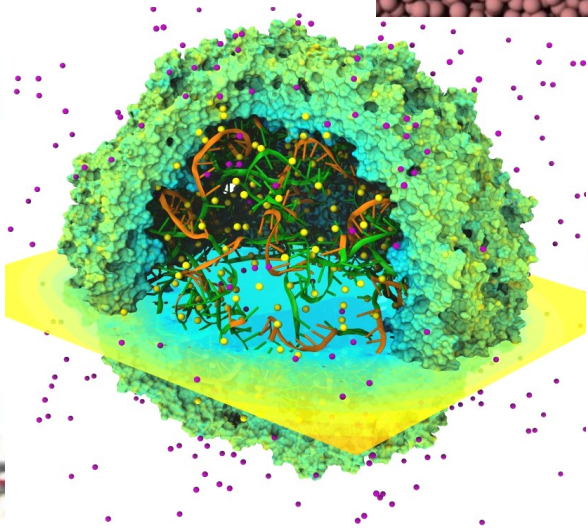
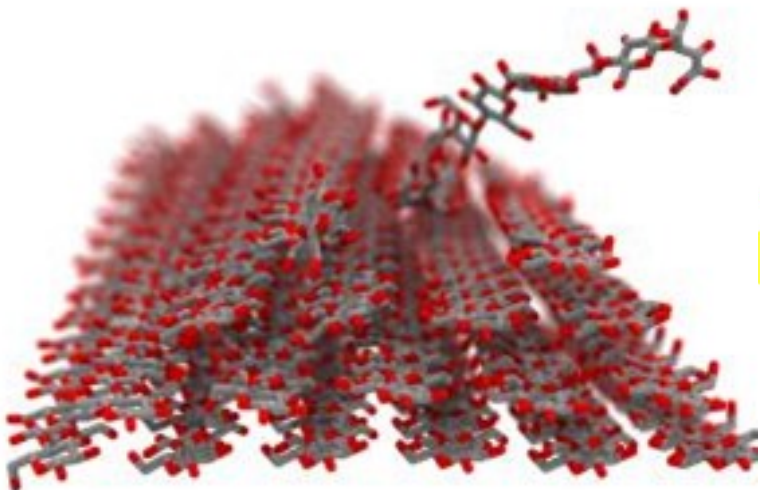
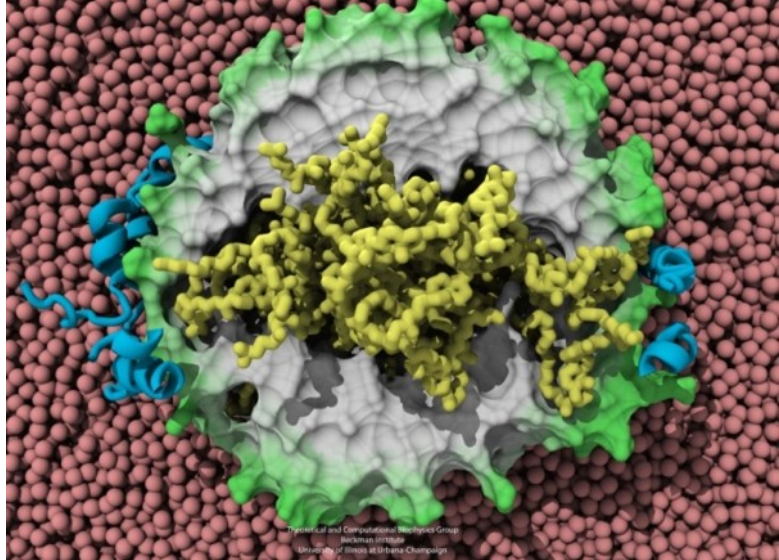
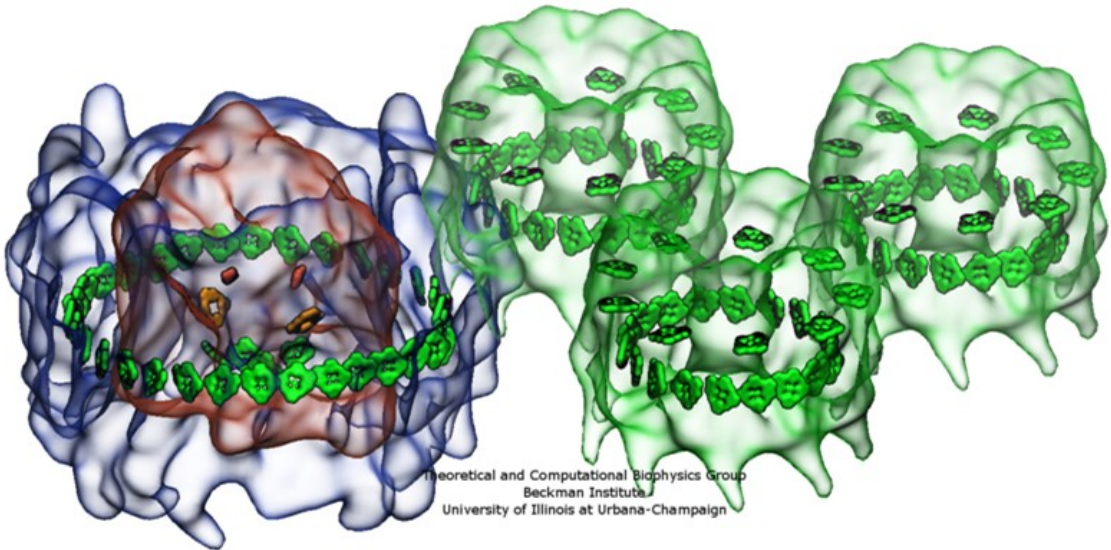


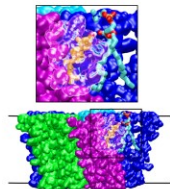
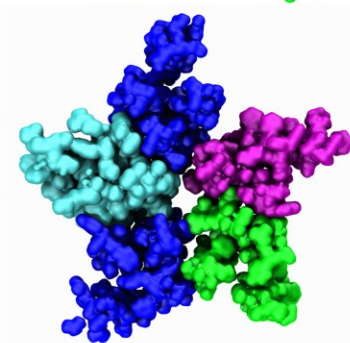
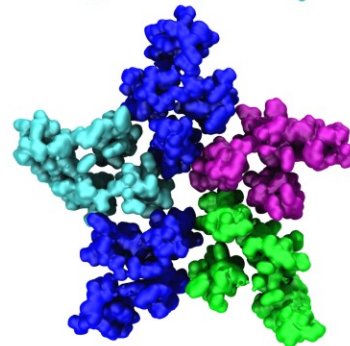
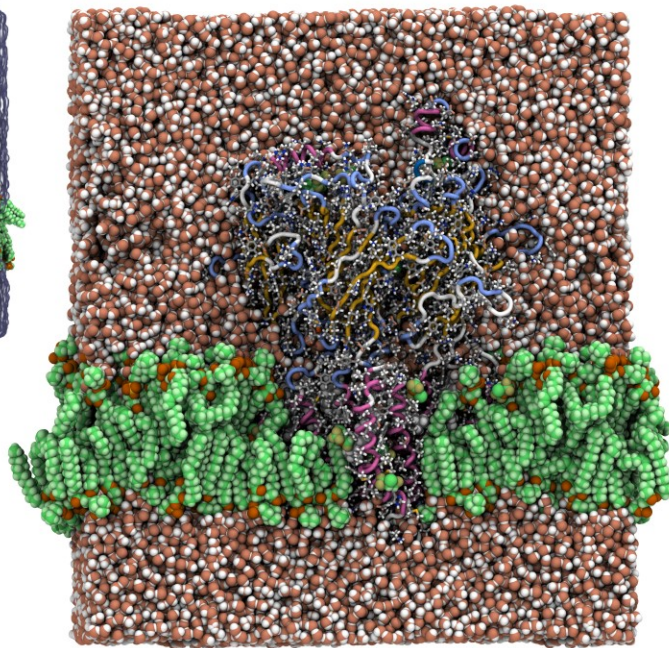
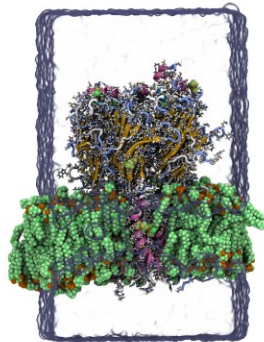
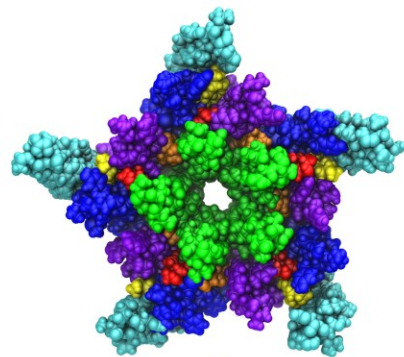
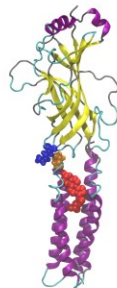
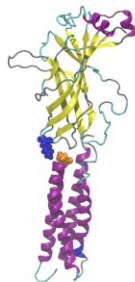
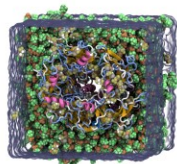
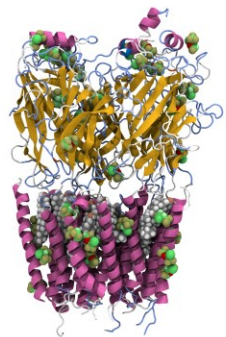
Close-up view of chloride ions permeating through HIV-1 capsid hexameric centers

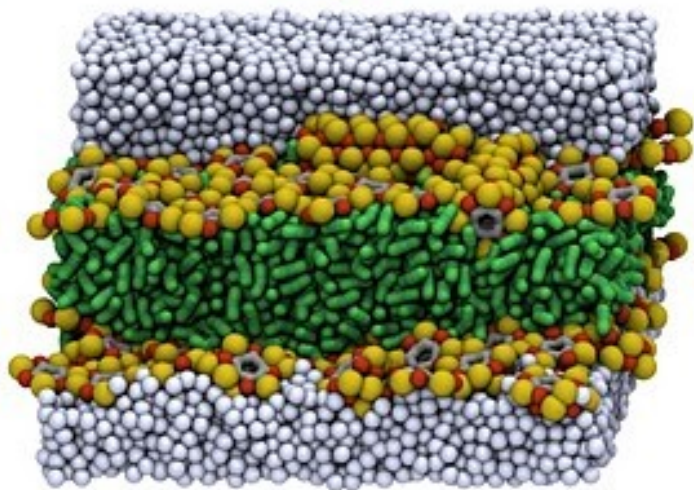
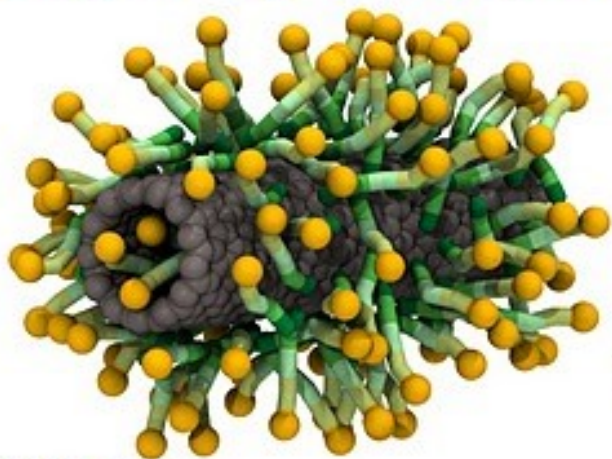
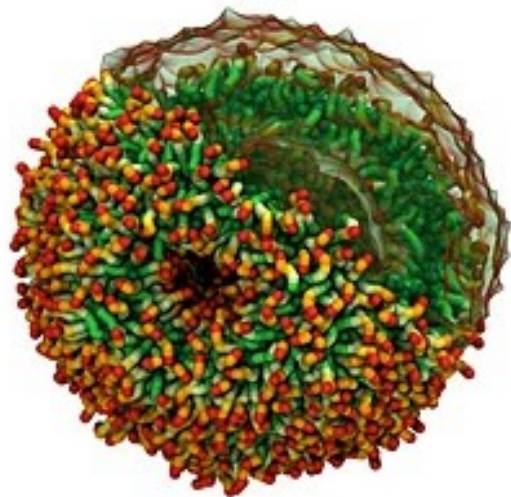
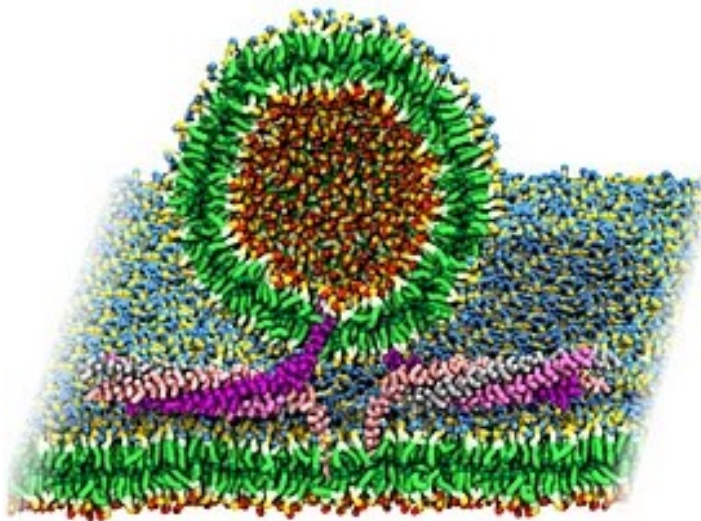


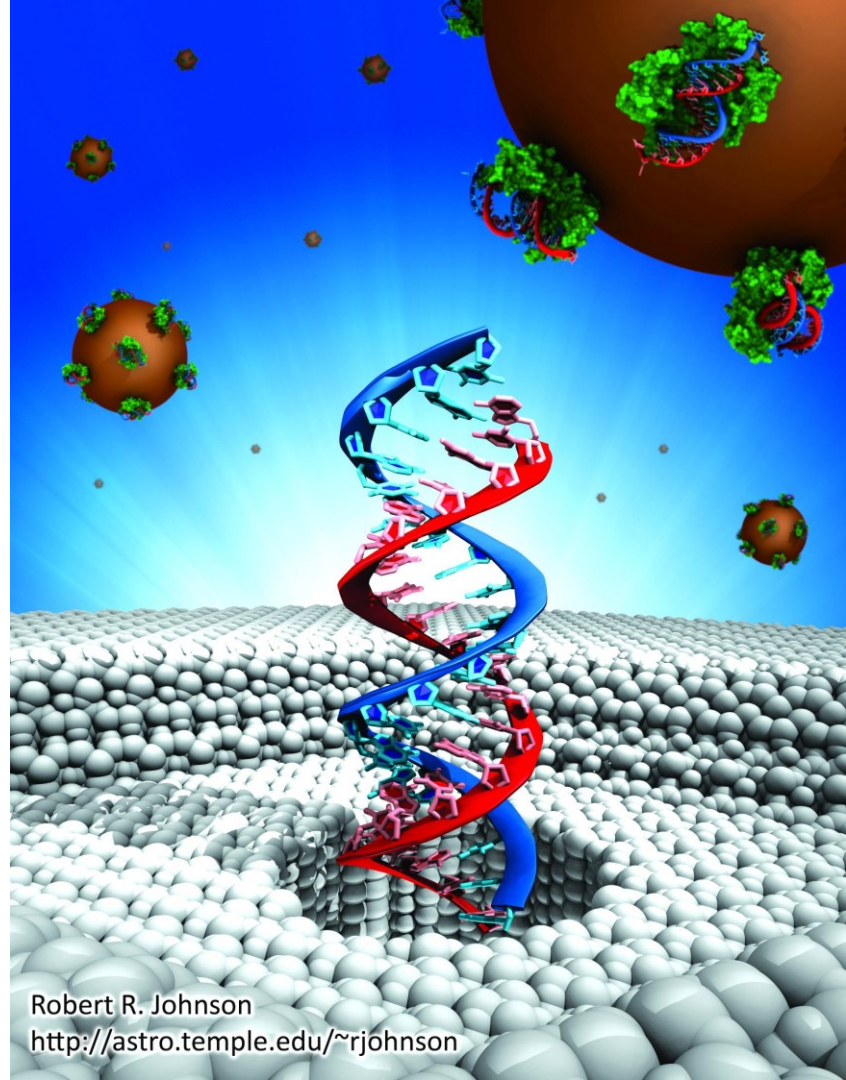
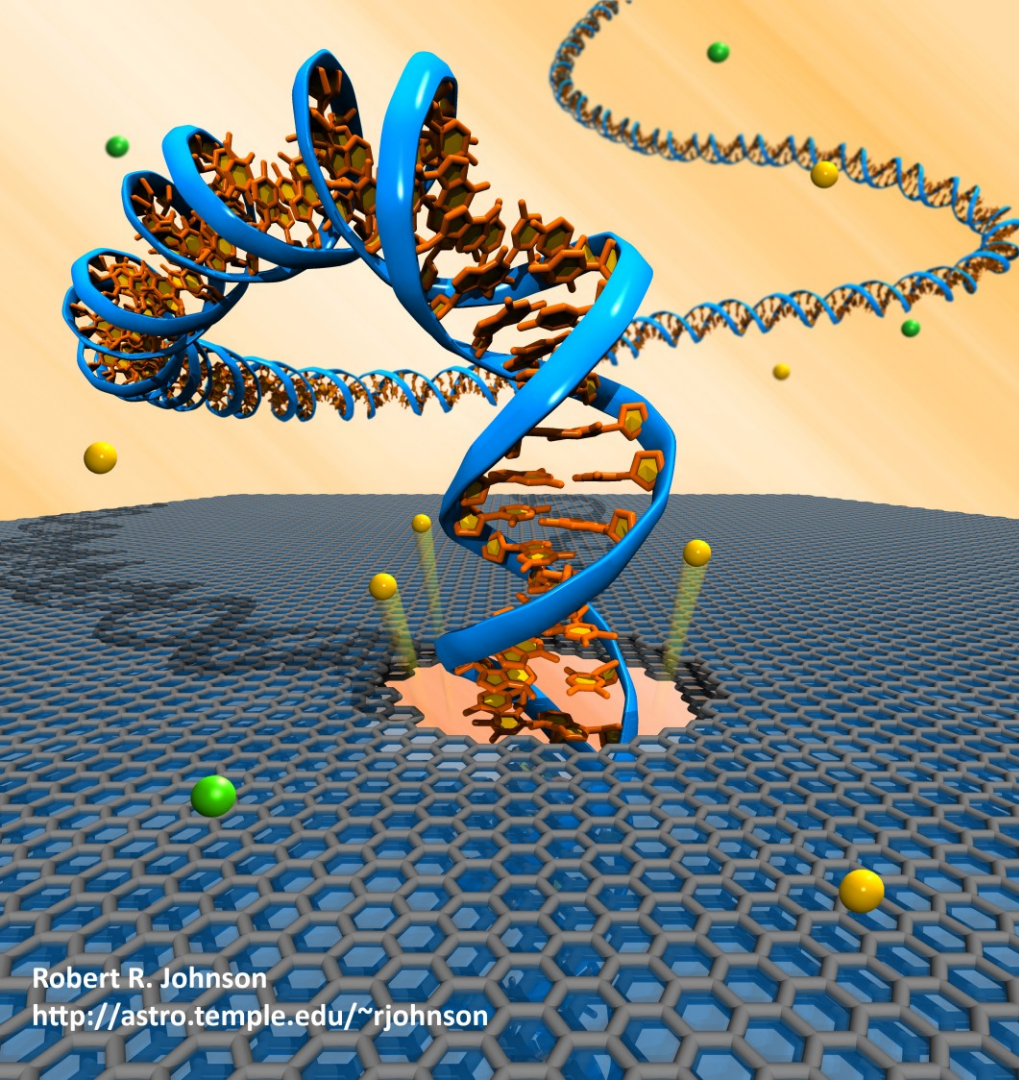
Satellite Tobacco Mosaic Virus

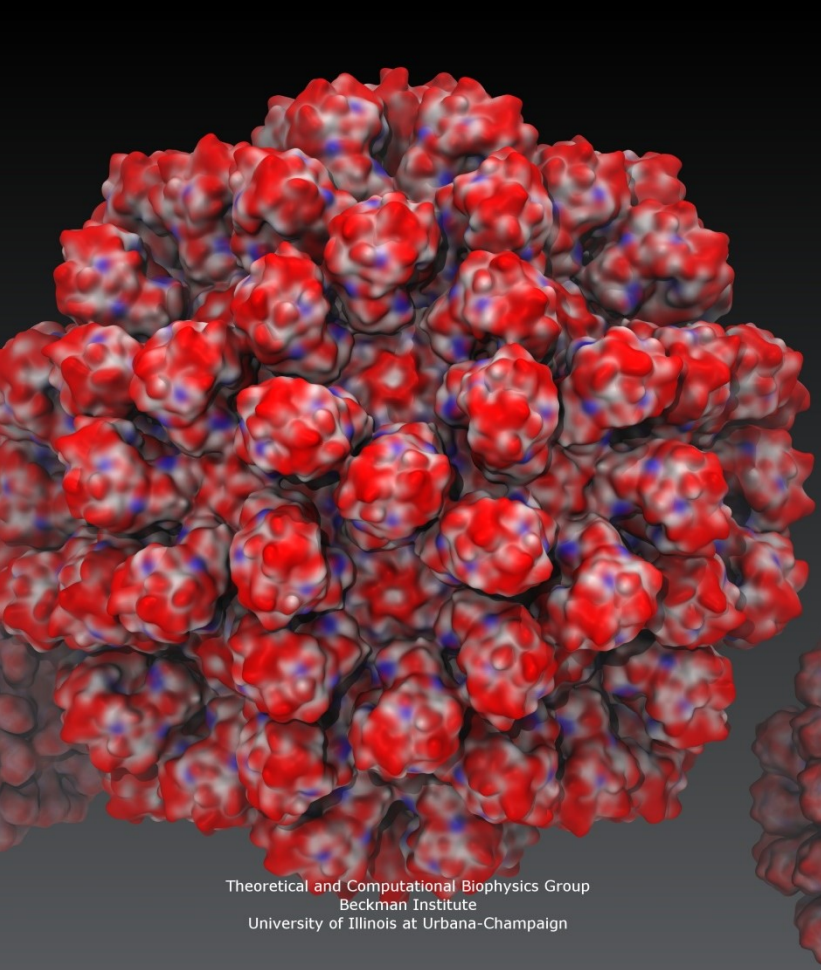
Biomedical Technology Research Center for Macromolecular Modeling and Bioinformatics
Beckman Institute, University of Illinois at Urbana-Champaign - www.ks.uiuc.edu



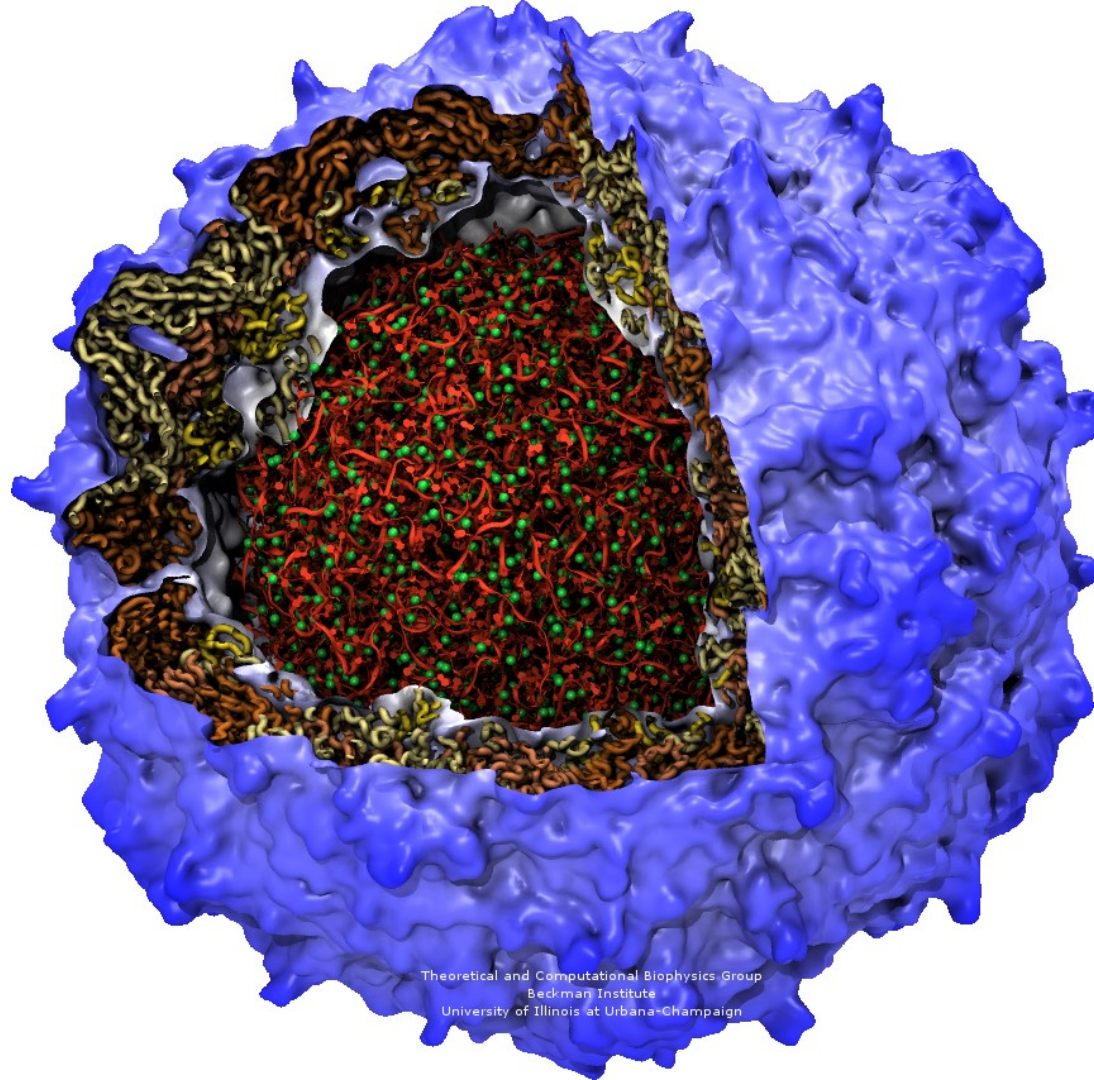








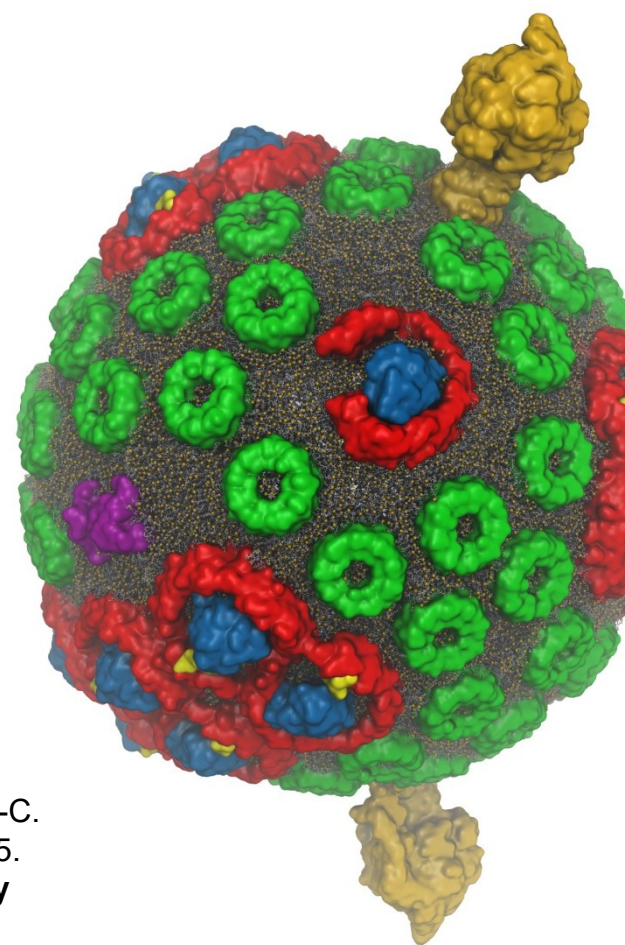
Theoretical and Computational Biophysics Group
Beckman Institute
University of Illinois at Urbana-Champaign



Theoretical and Computational Biophysics Group
Beckman Institute
University of Illinois at Urbana-Champaign

VMD w/ OptiX 6

- Interactive RT on laptops, desktops, and cloud
- Large-scale parallel rendering: in situ or post hoc visualization
- **Remote ray tracing with NvPipe video streaming**
- Stereoscopic panoramic and full-dome projections
- **Omnidirectional VR for YouTube, VR HMDs**
- VMD+OptiX NGC container: <https://ngc.nvidia.com/registry/>
- GPU memory sharing via NVLink
- **In-progress:**
 - **Denosing: faster turnaround w/ AO, DoF, etc**



GPU-Accelerated Molecular Visualization on Petascale Supercomputing Platforms.

J. E. Stone, K. L. Vandivort, and K. Schulten. UltraVis'13, pp. 6:1-6:8, 2013.

Visualization of Energy Conversion Processes in a Light Harvesting Organelle at Atomic Detail. M. Sener, et al. SC'14 Visualization and Data Analytics Showcase, 2014.

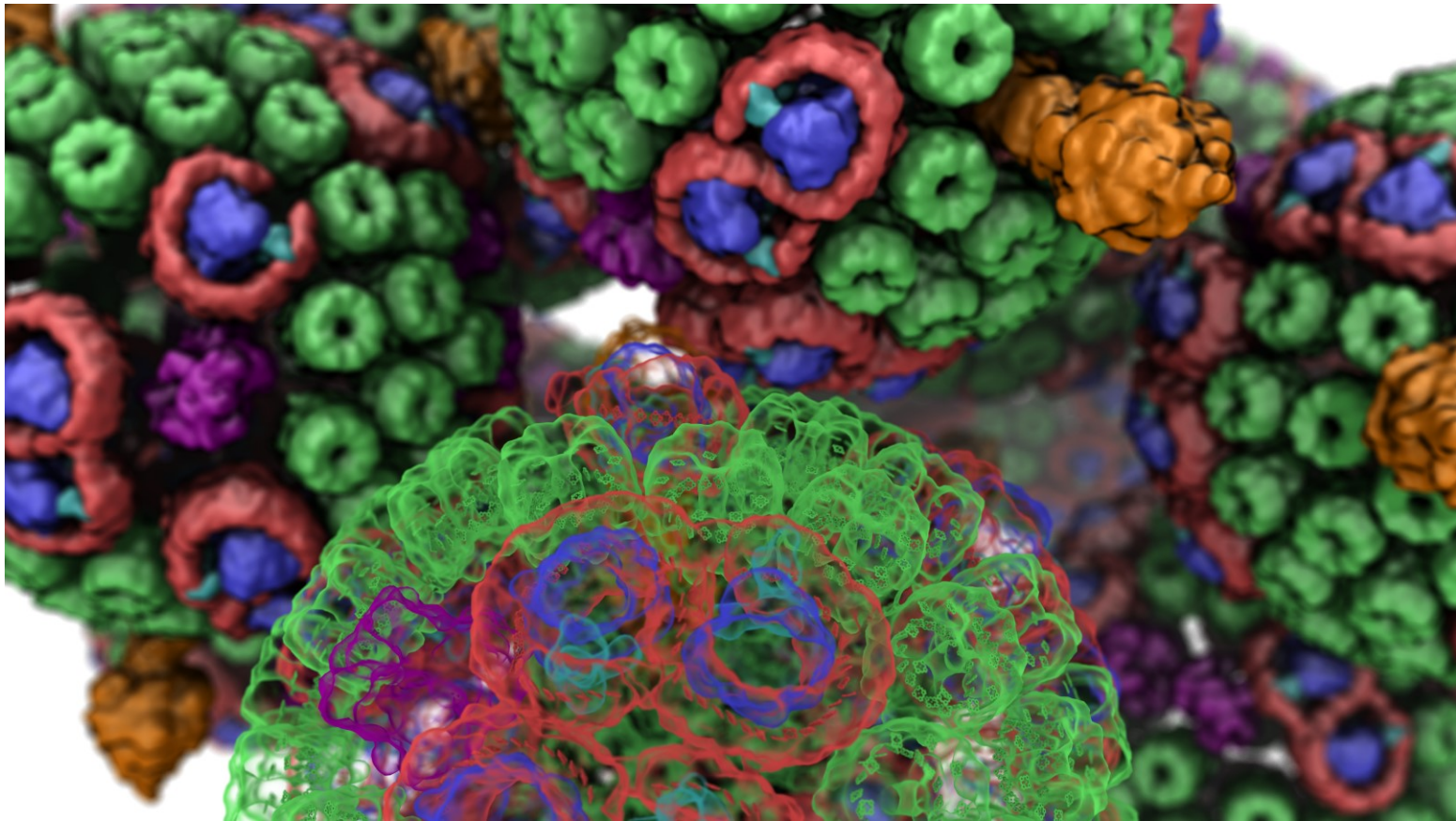
Chemical Visualization of Human Pathogens: the Retroviral Capsids. J. R. Perilla, B.-C. Goh, J. E. Stone, and K. Schulten. SC'15 Visualization and Data Analytics Showcase, 2015.

Atomic Detail Visualization of Photosynthetic Membranes with GPU-Accelerated Ray Tracing. J. E. Stone et al., J. Parallel Computing, 55:17-27, 2016.

Immersive Molecular Visualization with Omnidirectional Stereoscopic Ray Tracing and Remote Rendering J. E. Stone, W. R. Sherman, and K. HPDAV, IPDPSW, pp. 1048-1057, 2016.

VMD/OptiX GPU Ray Tracing of all-atom Chromatophore w/ lipids.

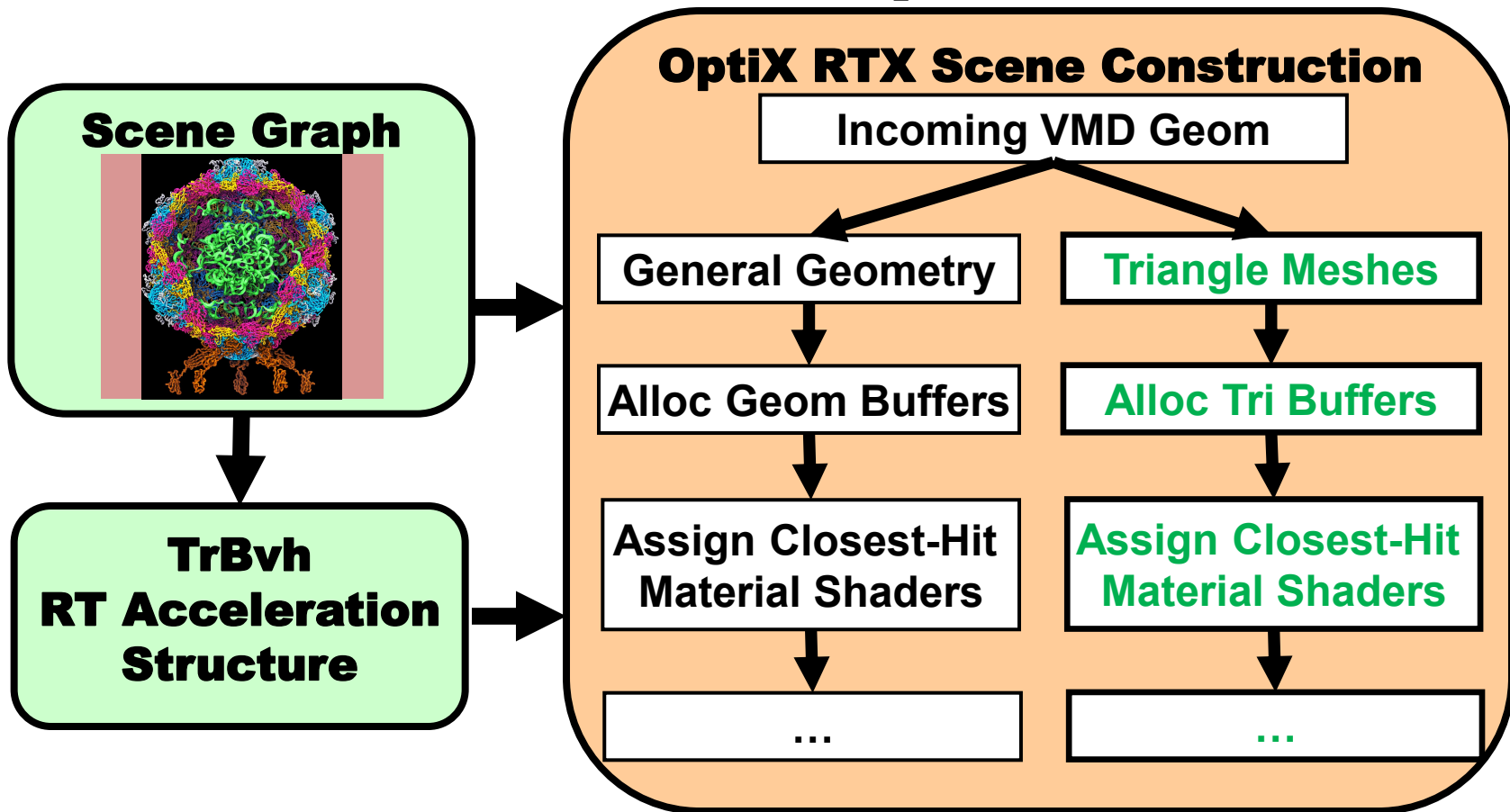
VMD/OpiX RTX Acceleration



What does RTX Acceleration do for VMD?

- Hardware acceleration (Turing RT cores) of:
 - BVH AS traversal
 - Ray-triangle intersection
- BVH AS can embed triangle geometry
 - Triangle geometry buffer can be ephemeral
- Often gives VMD a **~6x performance gain**

VMD Scene w/ OptiX RTX APIs

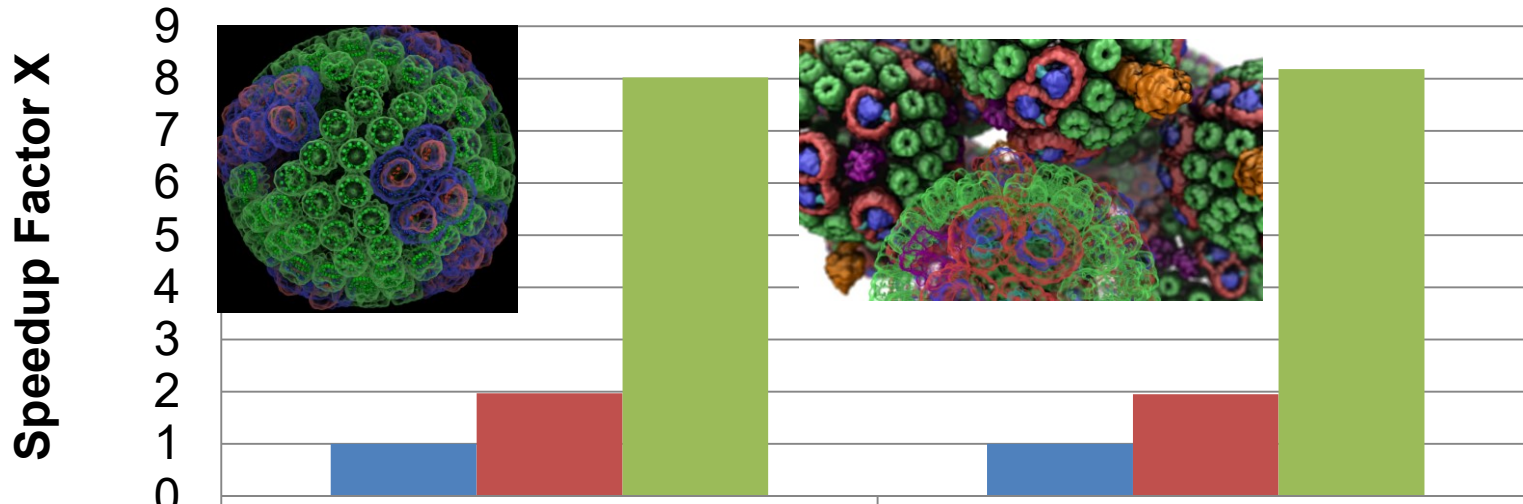


Secondary Benefits and RTX Performance Observations

- Traversal and intersection work performed by RT cores vacates GPU SMs and makes them available for other RT work
- RTX hardware traversal performance approaches GPU memory bandwidth limits
- Future RTX hardware could end up being bandwidth bound in some cases
- Start adapting geometric data representations for minimum footprint, e.g. by using compressed or quantized data representations such as Octahedron Normal Vector encoding (replace 3x 32-bit floats with a single 32-bit int)



VMD OptiX RT performance on Quadro RTX 6000

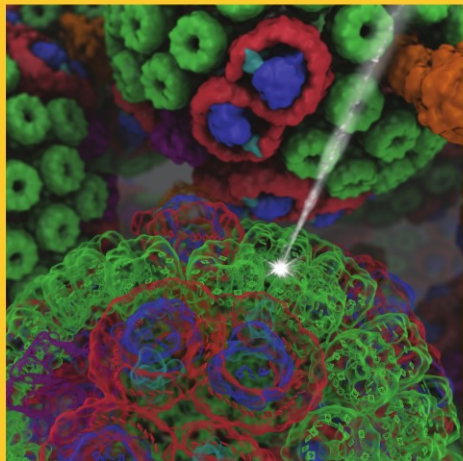


	Chromatophore @ 4Kx4K	Chrom Cell, 512x DoF @ 1080p
■ Quadro GV100	1	1
■ 2x Quadro GV100	1.97	1.95
■ Quadro RTX 6000	8.02	8.18

APRIL 20, 2017
VOLUME 121
NUMBER 15
pubs.acs.org/JPCB

THE JOURNAL OF PHYSICAL CHEMISTRY

B



KLAUS SCHULTEN MEMORIAL ISSUE

ACS Publications
Most Trusted. Most Cited. Most Read.

www.acs.org



Faraday Discussions

Volume: 169

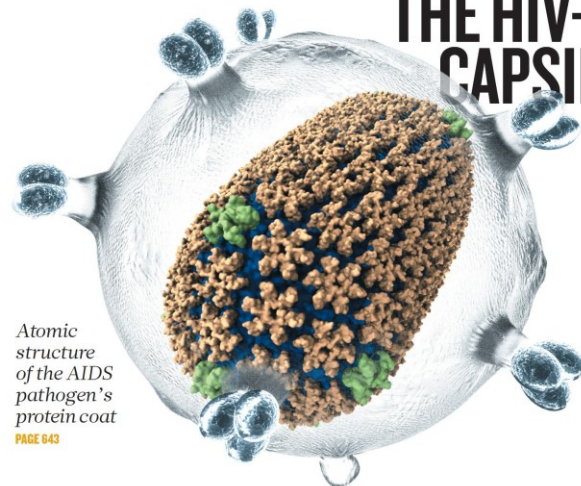
Molecular Simulations and Visualization



nature

THE INTERNATIONAL WEEKLY JOURNAL OF SCIENCE

THE HIV-1 CAPSID



Atomic
structure
of the AIDS
pathogen's
protein coat

PAGE 643

COSMOLOGY
**THE FIRST
LIGHT**
*In pursuit of the most
distant galaxies*
PAGE 554

CITATION
**CROSSING THE
BORDERS**
*International collaborations
make the most impact*
PAGE 557

ANTICANCER DRUGS
**A SITTING
TARGET**
*An indirect hit on
'undruggable' KRAS protein*
PAGES 577 & 638

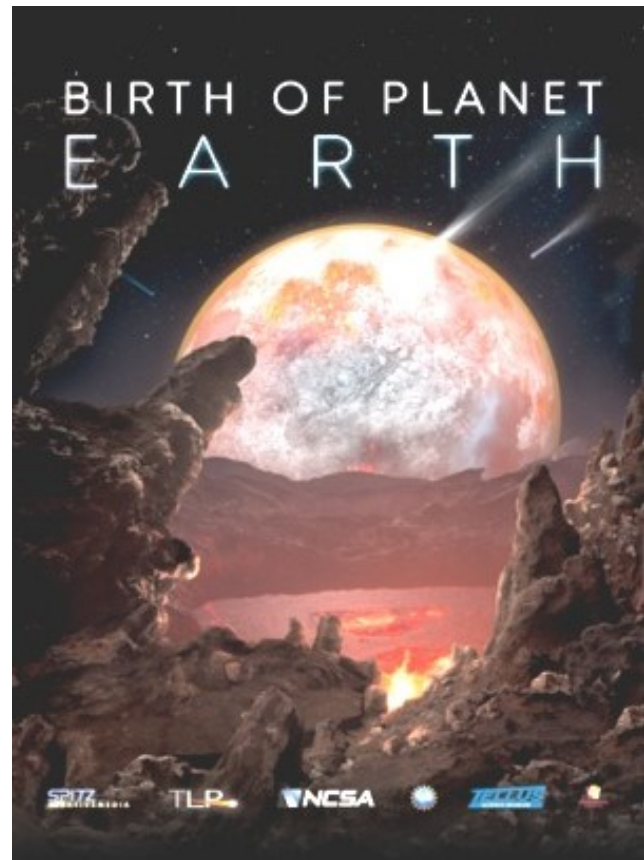
NATURE.COM/NATURE
30 May 2013



VMD Application Examples and Implementation Details

Cinematic Rendering: “Birth of Planet Earth” Fulldome Show

- Joint project with:
 - NCSA Advanced Visualization Lab
 - Thomas Lucas Productions, Inc.
 - Spitz Creative Media
 - Tellus Science Museum
- NSF Support: CADENS award ACI-1445176
- Premiered March 28, [Zeiss Großplanetarium](#), Berlin, Germany

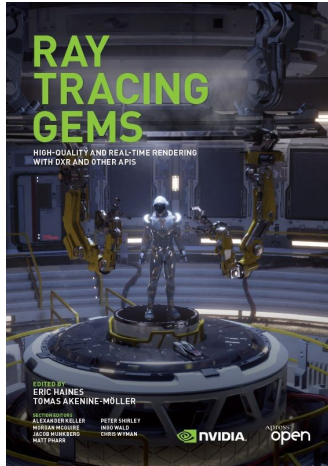


Why Use VMD for Cinematic Rendering

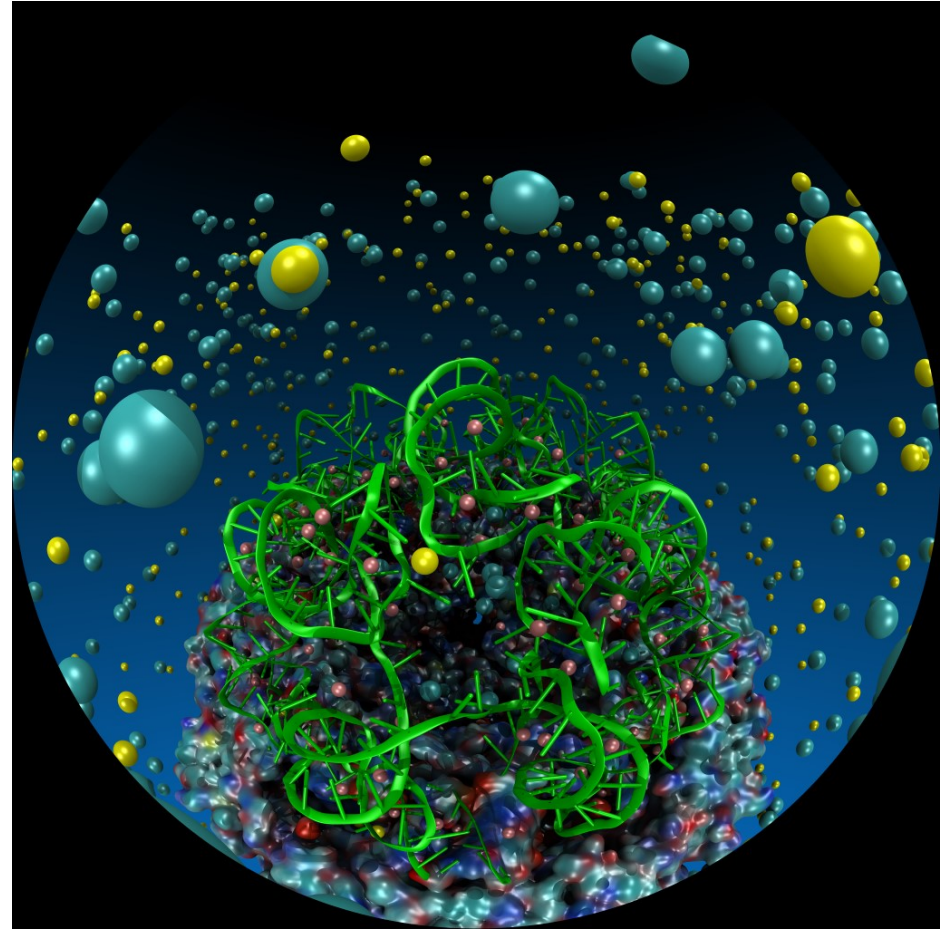
- Use all science materials where possible
- Maintain peak efficiency for calculating molecular geometries and representations
- OptiX-based RT engine and data structures designed for memory efficiency and performance specifically for molecular models
- Built-in scripting engines enabled control and incorporation within cinematic pipeline
- Avoid I/O bottlenecks

VMD Planetarium Dome Master Camera

- Fully interactive RT with ambient occlusion, shadows, depth of field, reflections, ...
- Both mono and stereoscopic
- No post-processing required

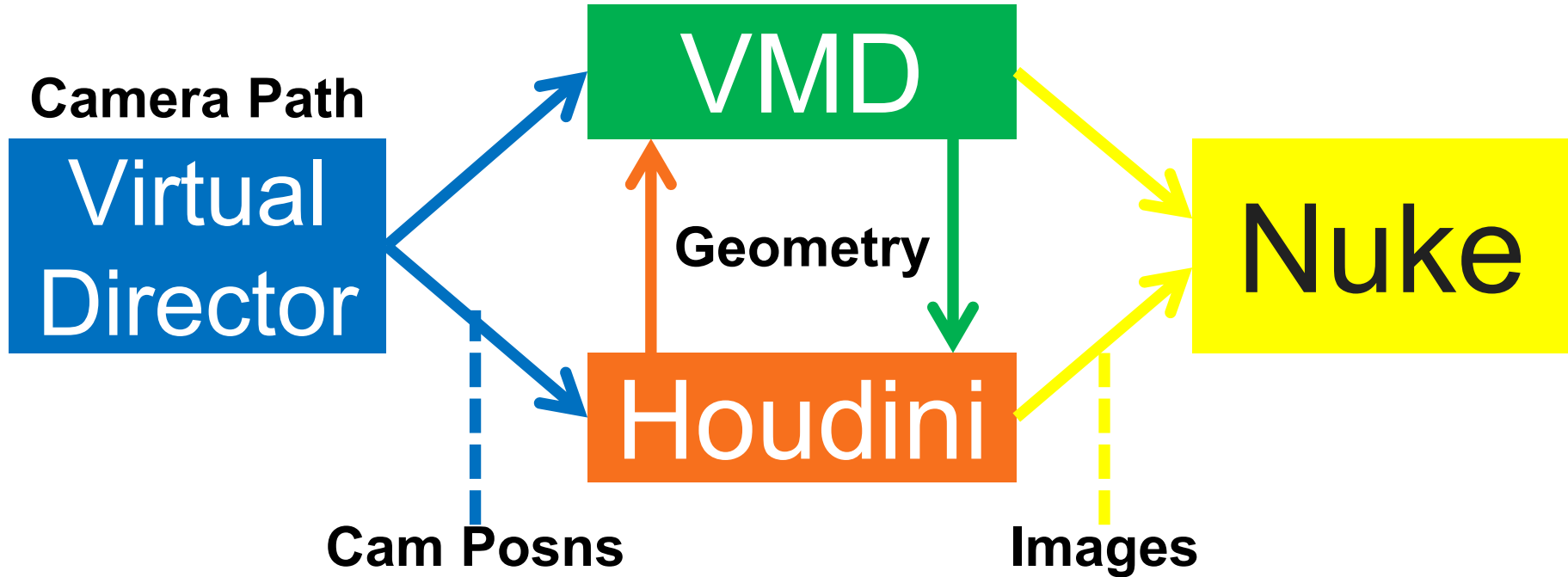


Ray Tracing Gems Ch. 4

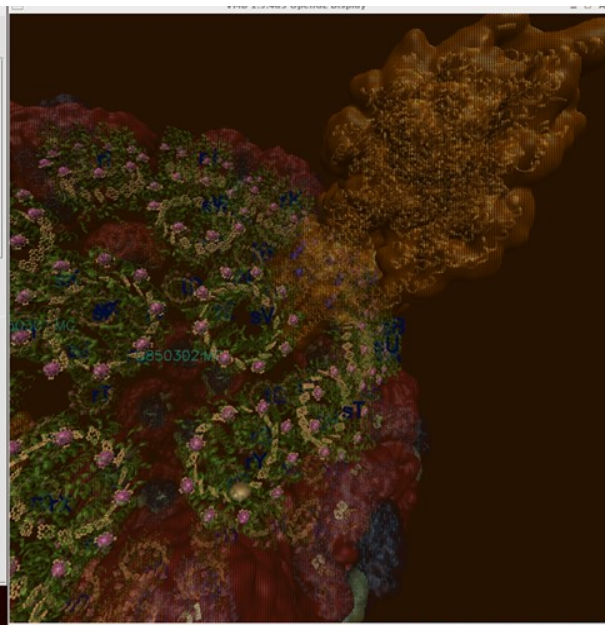
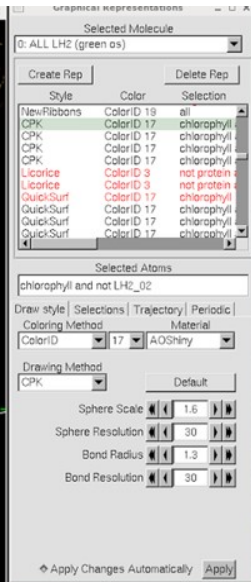
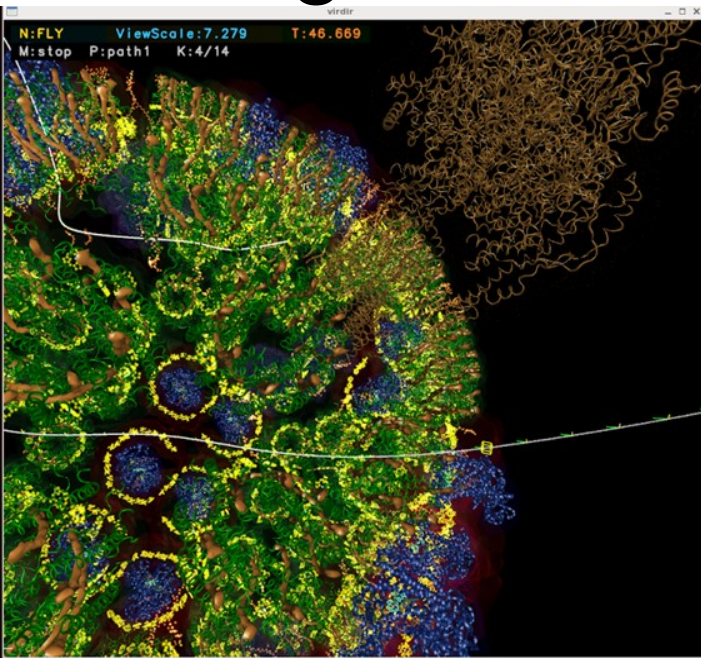


“Birth of Planet Earth” Cinematic Pipeline

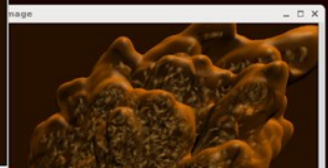
Visualization and Rendering



Integration of VMD with Virtual Director



ID	T	A	D	F	Molecule	Atoms	Frames	Val
0	A	D	F		ALL LHZ (green os)	1140604	1	0
1	A	D	F		ALL LH1 RC (red 8s)	950440	1	0
2	A	D	F		ALL BC (gold pink)	1104662	1	0
3	A	D	F		ALL ATPs (yellow turns)	151250	1	0
4	A	D	F		ALL LH1 not RC (red os 90320)		1	0
5	A	D	F		extra_ELMolaraout12031t	533147	1	0
6	A	D	F		extra_Vesicle30_LHZ_3	181056	1	0
7	A	D	F		SELECT LHZ RIBBONS	54324	1	0
8	A	D	F		SELECT LH1 RIBBONS	90040	1	0
9	A	D	F		SELECT BC RIBBONS	27608	1	0
10	A	D	F		SELECT ATP RIBBONS	75625	1	0
11	A	D	F		extra_ATPsynthase_mit	75625	1	0
12	A	D	F		SELECT NUCLEIC ACID 135	135	1	0
13	A	D	F		extra_atpase_Ves30_at	75625	1	0
14	A	D	F		SELECT NUCLEIC ACID 135	135	1	0
15	A	D	F		SELECT NUCLEIC ACID 135	135	1	0
16	A	D	F		SELECT NUCLEIC ACID 135	135	1	0
17	A	D	F		SELECT NUCLEIC ACID 135	135	1	0
18	A	D	F		extra_atpase_Ves30_at	7597	1	0
19	A	D	F		electron_1.pdb	1	1	0
20	A	D	F		electron_2.pdb	1	1	0
21	A	D	F		electron_3.pdb	1	1	0
22	A	D	F		quinoRC.xmgr	14	1	0
23	A	D	F		cytC2.pdb	1060	1	0
24	A	D	F		hydronium.pdb	4	1	0
25	A	D	F		hydronium2.pdb	4	1	0



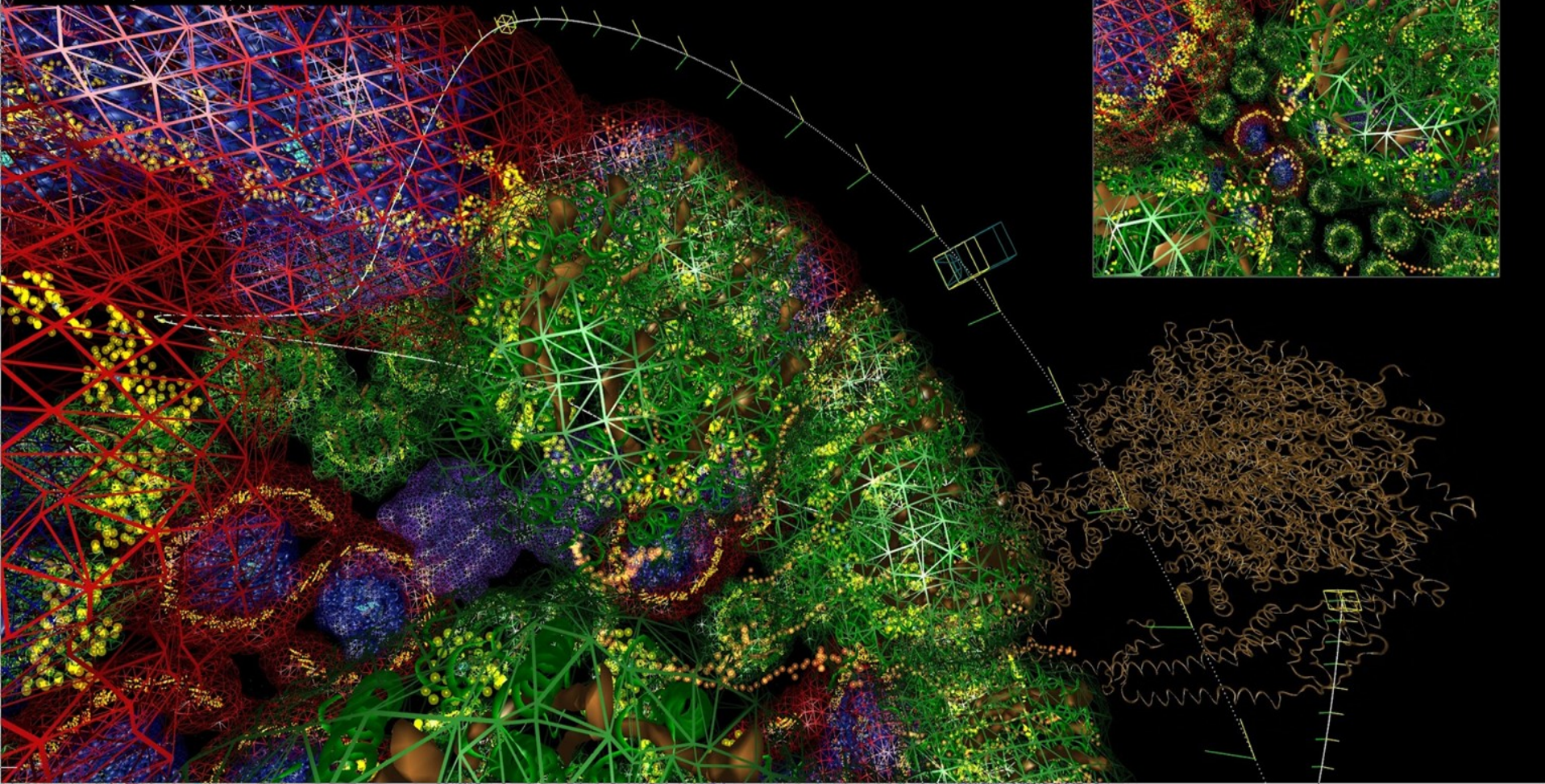
```
stevy@junco:/bdf5/bape/vtdir
File Edit View Search Terminal Help
virdir> stop
Stopped
virdir> pv gall -v every
g0:
display every 1th particle (of 0)
p1=doleshpere: (on)
p1display every 2th particle (of 750370)
p2=chloro: (on)
display every 1th particle (of 69365)
p3=carotenoid: (on)
display every 1th particle (of 10990)
p4=ribbons: (on)
display every 1th particle (of 0)
part1: gall.
virdir> pv g2 slum *10
slum 10 (var 29 dataset (0))
exit
```

```
stevy@junco:/fe2/bape/watsh/images/under/DiskA_5-25
File Edit View Search Terminal Help
[Info] User-overide of Optix shader path: /18/src/grap/vmd/vmd-1.9.4a5/LINUXAMD64/OptIXShaders.ptx
Optix(DisplayDevice) Total rendering time: 9.39 sec
[Info] Rendering complete.
New Image /tmp/rt.ppm 888x800
vmd > rt
[Info] Rendering current scene to '/tmp/rt.ppm' ...
Optix(DisplayDevice) Total rendering time: 9.31 sec
[Info] Rendering complete.
New Image /tmp/rt.ppm 888x800
vmd > rt
[Info] Rendering current scene to '/tmp/rt.ppm' ...
Optix(DisplayDevice) Total rendering time: 9.66 sec
[Info] Rendering complete.
New Image /tmp/rt.ppm 888x800
vmd > rt
[Info] Rendering current scene to '/tmp/rt.ppm' ...
Optix(DisplayDevice) Total rendering time: 9.21 sec
[Info] Rendering complete.
New Image /tmp/rt.ppm 888x800
vmd > rt
```

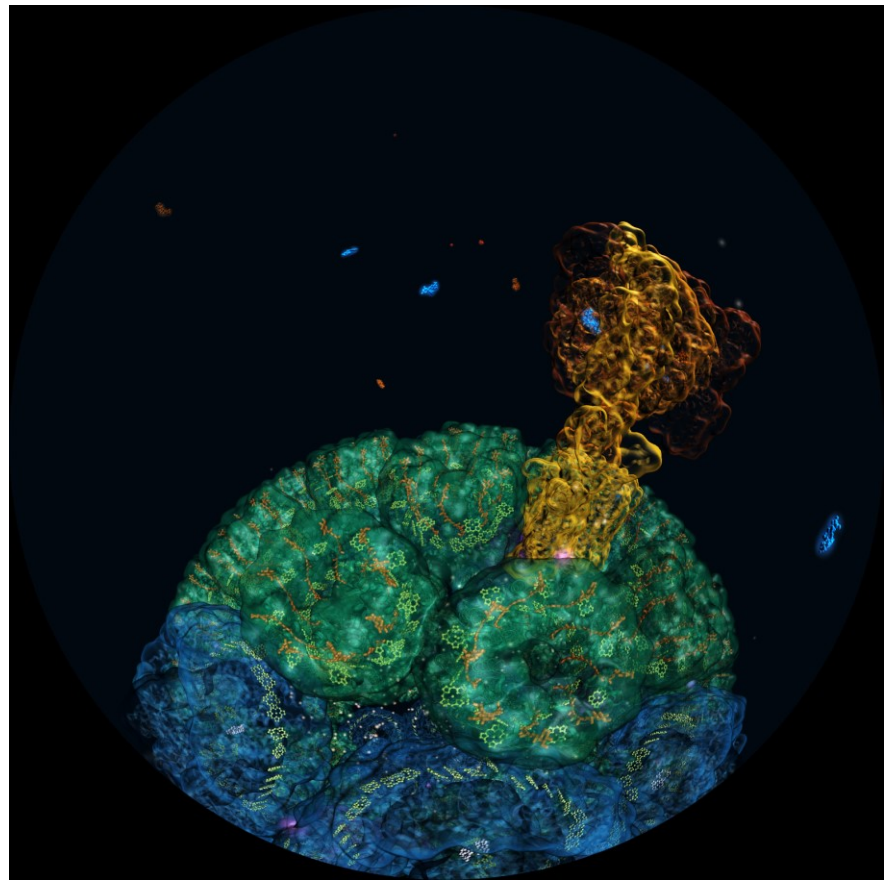
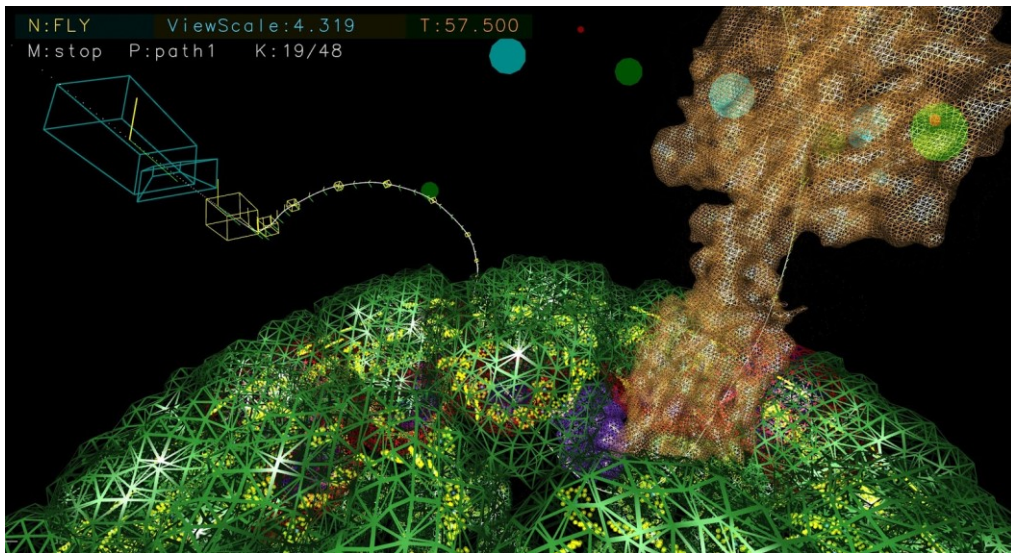


N:FLY ViewScale:3.666 T:56.000

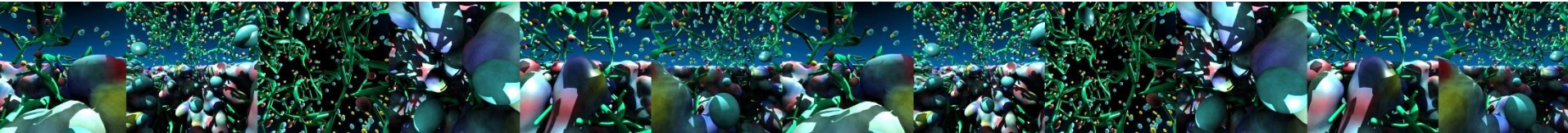
M:stop P:path1 K:5/14



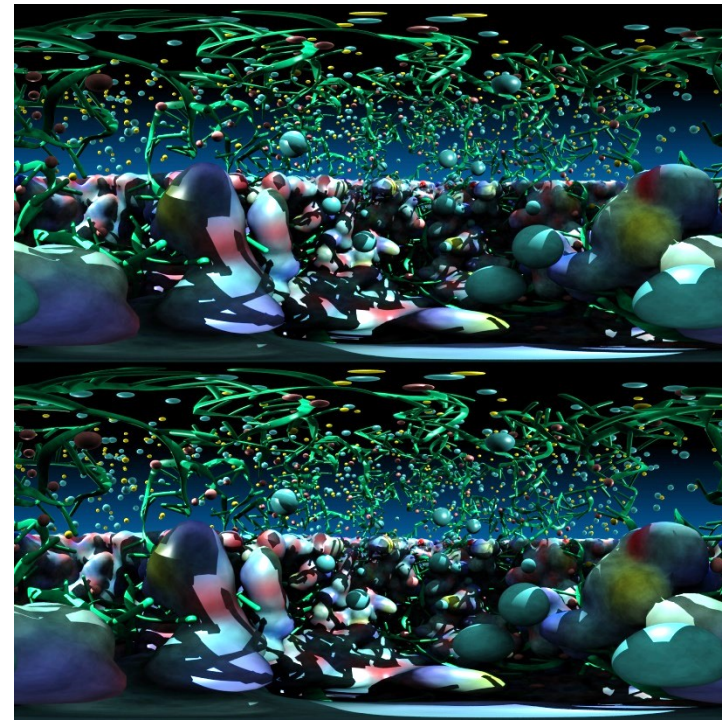
Camera Path in Virtual Director and VMD



Stereoscopic Panorama Ray Tracing

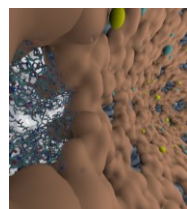
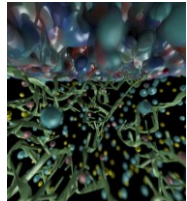
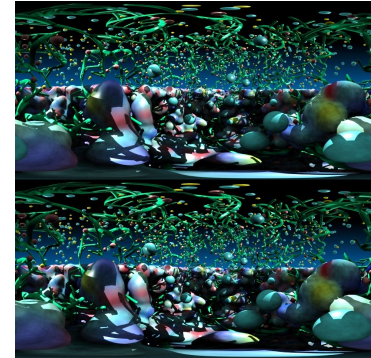
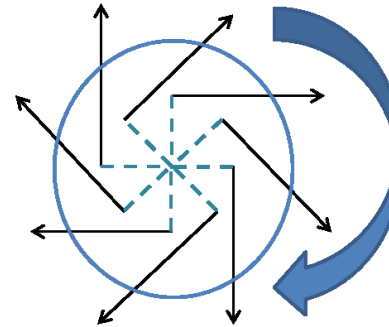
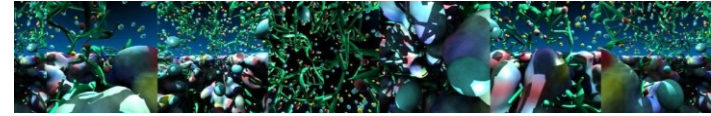


- **Render 360° images and movies for VR headsets such as Oculus Rift, Google Cardboard, YouTube VR**
- Ray trace omnidirectional stereo spheremaps or cubemaps for very high-frame-rate reprojection and display via OpenGL texturing
- Stereo requires spherical camera projections **poorly suited to rasterization**
- Benefits from OptiX multi-GPU rendering and load balancing, **remote visualization**



Omnidirectional Stereoscopic Ray Tracing

- **Ray trace 360° images and movies for Desk and VR HMDs: Oculus, Vive, Cardboard**
- Stereo spheremaps or cubemaps allow very high-frame-rate interactive OpenGL display
- **AO lighting, depth of field, shadows, transparency, curved geometry, ...**
- **Summit 6x Tesla V100 GPU nodes:**
 - Render many omni-stereo viewpoints, no acceleration structure rebuilds, tens of frames/sec per-node!
 - OptiX multi-GPU rendering, NVLink compositing and data distribution, etc...
 - Future: AI for warping between views



Atomic Detail Visualization of Photosynthetic Membranes with GPU-Accelerated Ray Tracing. J. E. Stone, et al. J. Parallel Computing, 55:17-27, 2016.

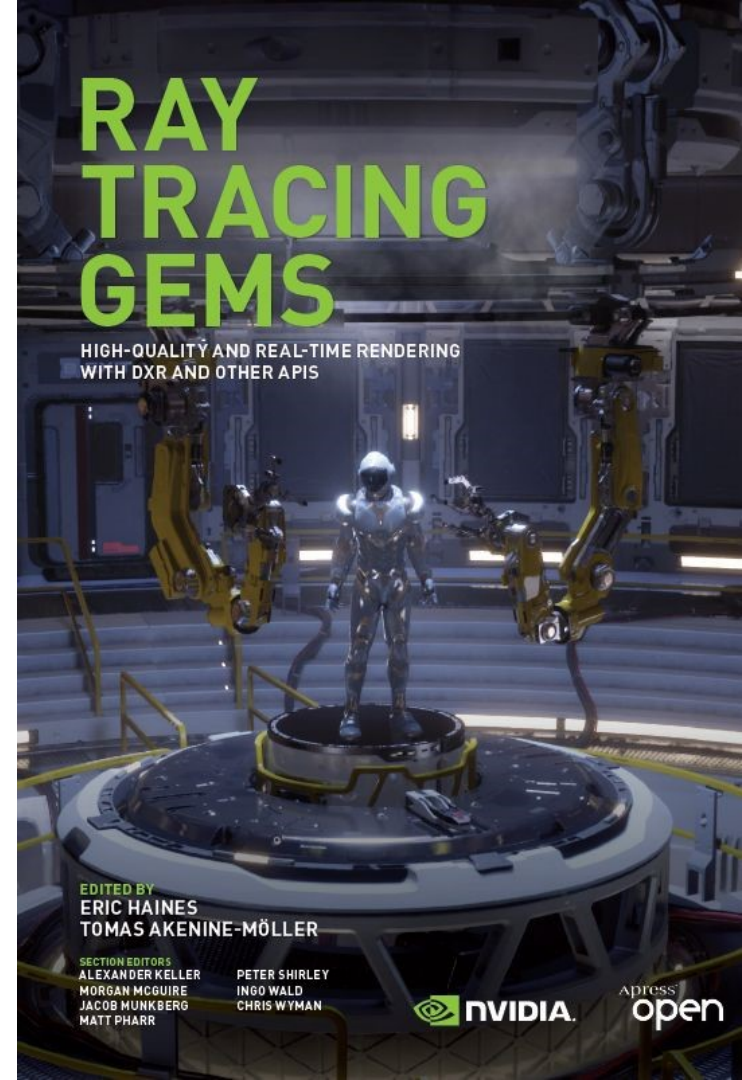
Immersive Molecular Visualization with Omnidirectional Stereoscopic Ray Tracing and Remote Rendering. J. E. Stone, W. R. Sherman, and K. Schulten. High Performance Data Analysis and Visualization Workshop, IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), pp. 1048-1057, 2016.



Immersive Molecular Visualization with Omnidirectional Stereoscopic Ray Tracing and Remote Rendering. J. E. Stone, W. R. Sherman, and K. Schulten. High Performance Data Analysis and Visualization Workshop, IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), pp. 1048-1057, 2016.

Ray Tracing Gems

- Ch. 4, “A Planetarium Dome Master Camera”
- Ch. 27, “Interactive Ray Tracing Techniques for High-Fidelity Scientific Visualization”
- Tons of great material and code samples!



Molecular Dynamics Flexible Fitting (MDFF)

X-ray crystallography



APS at Argonne

MDFF

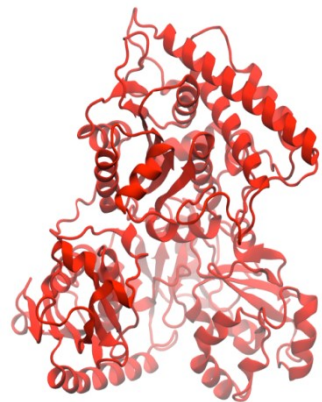


Electron microscopy

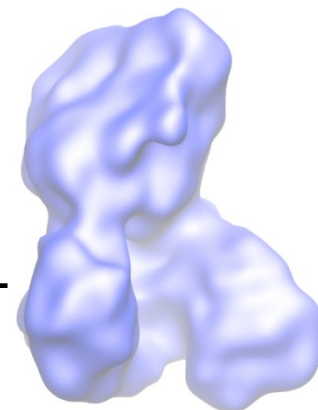


FEI microscope

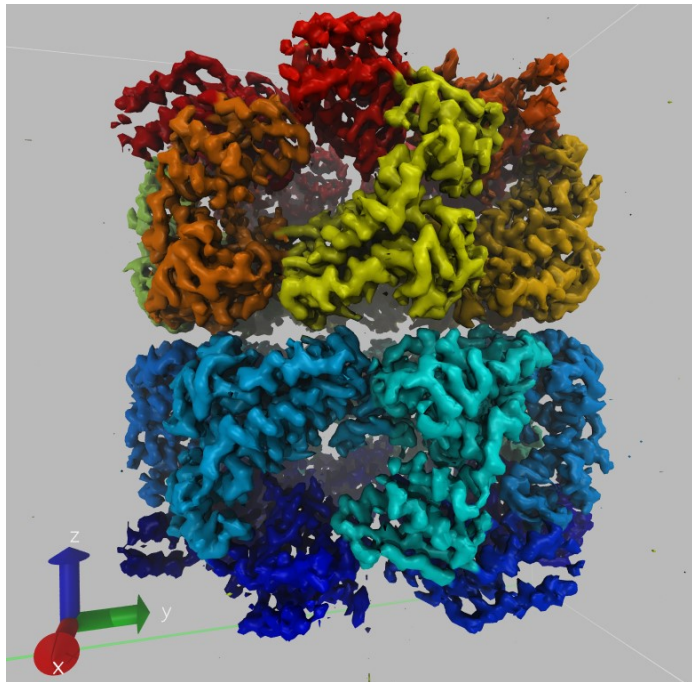
ORNL Titan



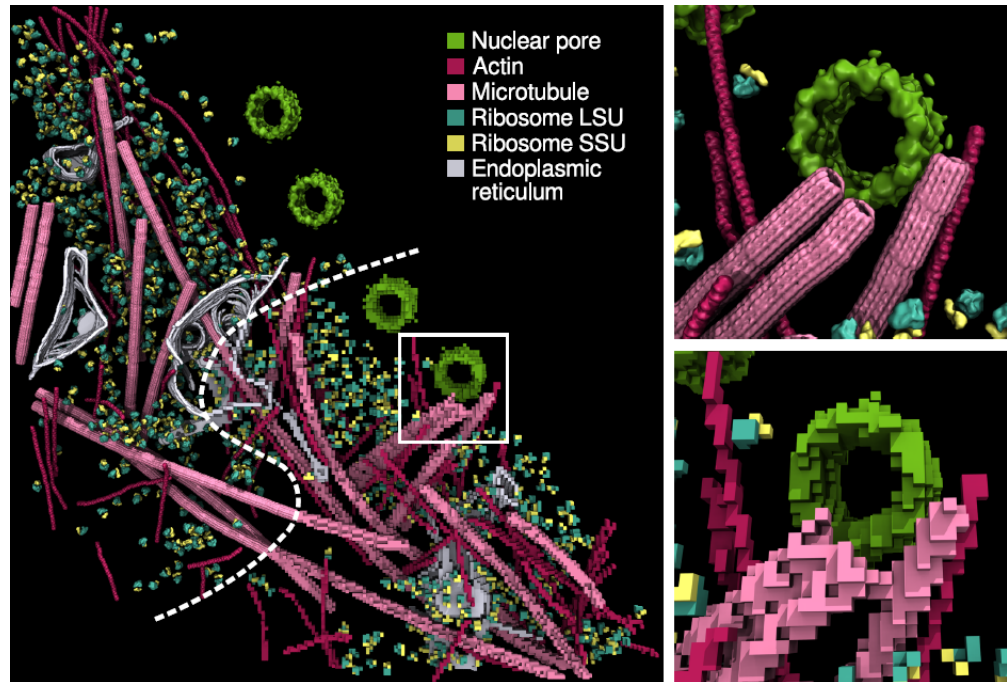
Molecular dynamics-based refinement and validation for sub-5Å cryo-electron microscopy maps. A. Singharoy, I. Teo, R. McGreevy, J. E. Stone, J. Zhao, and K. Schulten. *eLife* 2016;10.7554/eLife.16105



Density Map Segmentation



VMD GPU-accelerated density map segmentation of GroEL

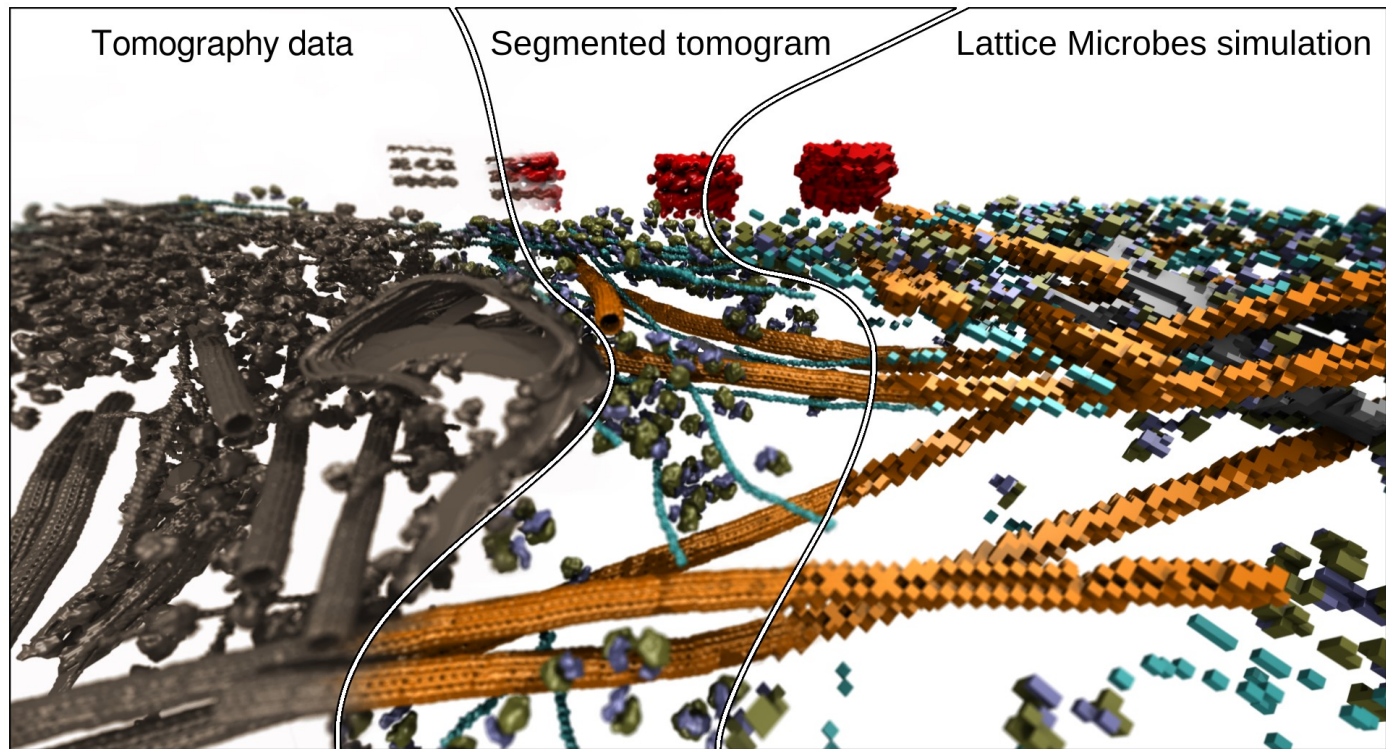


Earnest, et al. *J. Physical Chemistry B*, 121(15): 3871-3881, 2017.

Cryo-EM / Cryo-ET Density Map Segmentation

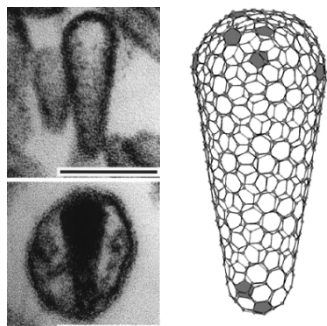
Evaluate 3-D volumetric electron density maps and segment them, to identify key structural components

Index/label components so they can be referred to, colored, analyzed, and simulated...

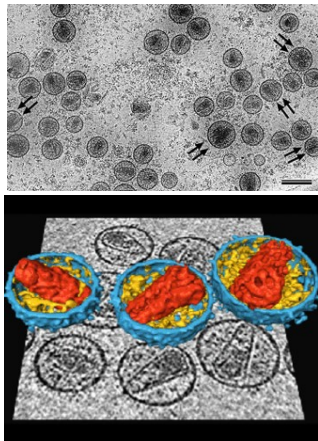


Structural Route to the all-atom HIV-1 Capsid

1st TEM (1999) 1st tomography (2003)

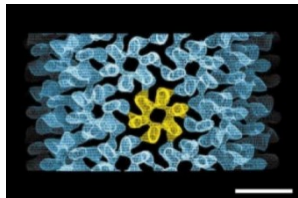


Ganser et al. *Science*, 1999
Briggs et al. *EMBO J*, 2003
Briggs et al. *Structure*, 2006

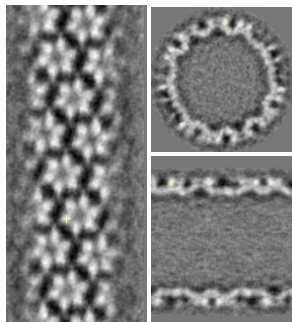


cryo-ET (2006)

hexameric tubule

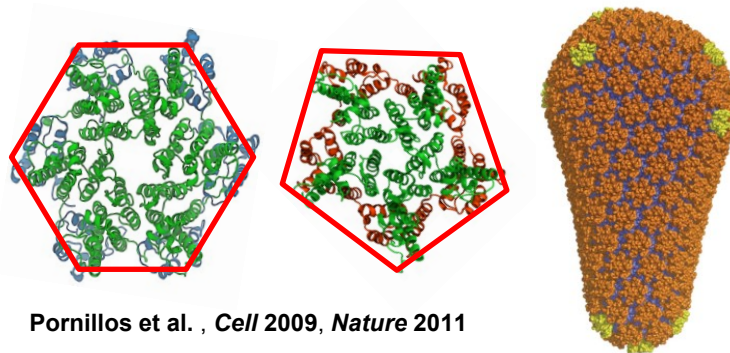


Li et al., *Nature*, 2000



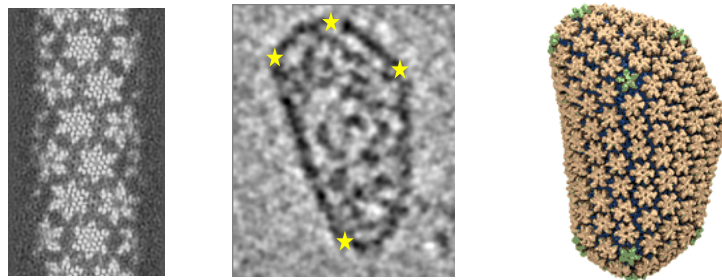
Byeon et al., *Cell* 2009

Crystal structures of separated hexamer and pentamer



Pornillos et al., *Cell* 2009, *Nature* 2011

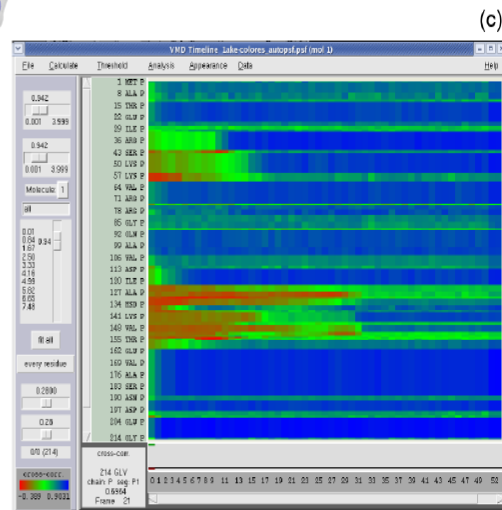
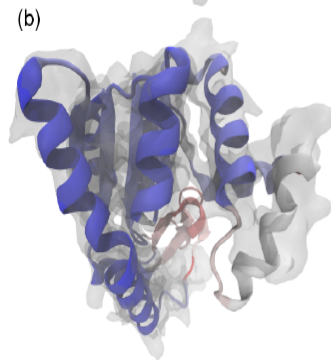
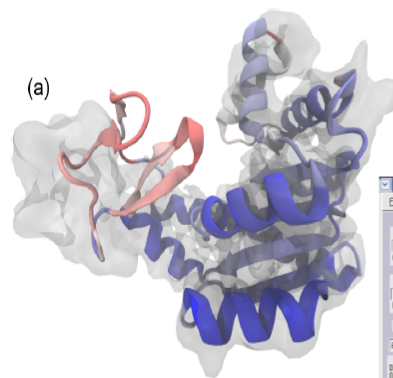
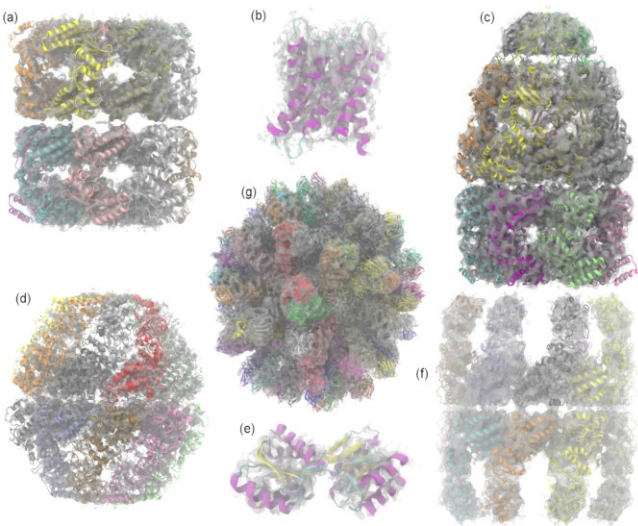
High res. EM of hexameric tubule, tomography of capsid,
all-atom model of capsid by MDFF w/ NAMD & VMD,
NSF/NCSA Blue Waters computer at Illinois



Zhao et al., *Nature* 497: 643-646 (2013)

Evaluating Quality-of-Fit for Structures Solved by Hybrid Fitting Methods

Compute Pearson correlation to evaluate quality-of-fit between a reference cryo-EM density map and a **simulated density map** from an **all-atom structure**.



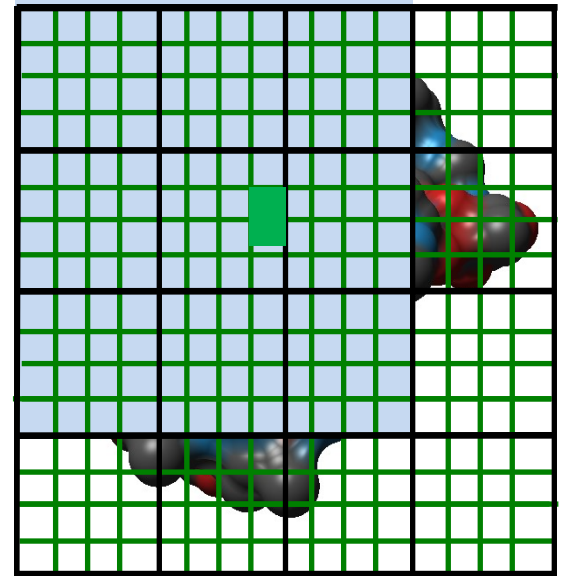
MDFF Cross Correlation Timeline
Regions with poor fit **Regions with good fit**

MDFFF Density Map Algorithm

- Build spatial acceleration data structures, optimize data for GPU
- Compute 3-D density map:

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

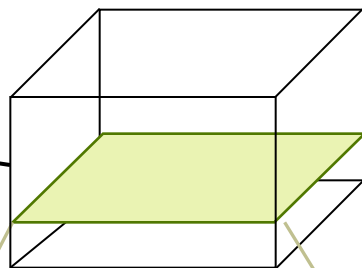
- Truncated Gaussian and spatial acceleration grid ensure linear time-complexity



3-D density map lattice point and the neighboring spatial acceleration cells it references

Single-Pass MDFF GPU Cross-Correlation

3-D density map decomposes into 3-D grid of 8x8x8 tiles containing CC partial sums and local CC values

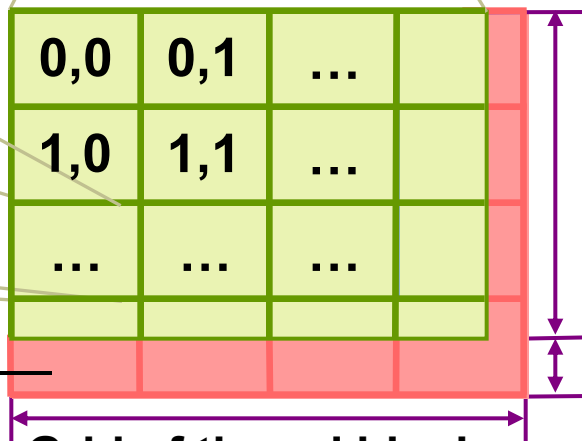
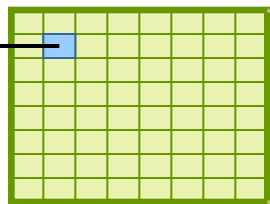


Fusion of density and CC calculations into a single CUDA kernel!!!

Spatial CC map and overall CC value computed in a single pass

Small 8x8x2 CUDA thread blocks afford large per-thread register count, shared memory

Each thread computes 4 z-axis density map lattice points and associated CC partial sums



Threads producing results that are used

Inactive threads, region of discarded output

Padding optimizes global memory performance, guaranteeing coalesced global memory accesses

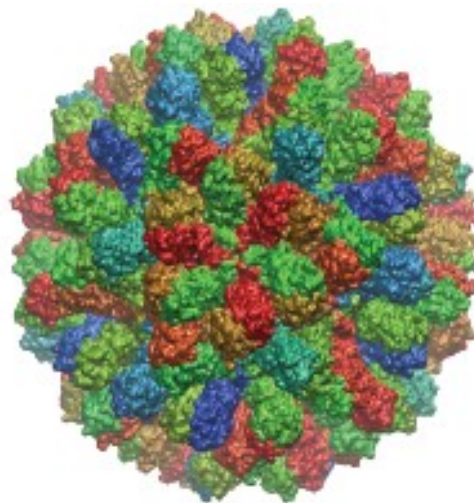
Grid of thread blocks

Parallel MDFF Cross Correlation Analysis on Cray XK7

Rabbit Hemorrhagic Disease Virus (RHDV)

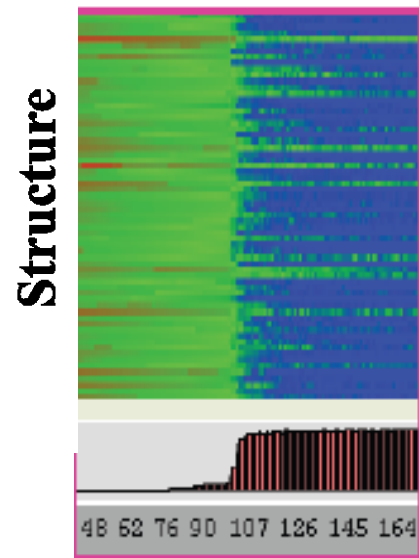
Traj. frames	10,000
Structure component selections	720
Single-node XK7 (projected)	336 hours (14 days)
128-node XK7	3.2 hours 105x speedup
2048-node XK7	19.5 minutes 1035x speedup

Calculation of 7M CCs would take **5 years** using serial CPU algorithm!



**RHDV colored
by relative CC**

Relative CC
-0.0032 0.02



Time

VMD Tesla V100 Cross Correlation Performance

Rabbit Hemorrhagic Disease Virus: 702K atoms, 6.5Å resolution

Volta GPU architecture almost 2x faster than previous gen Pascal:

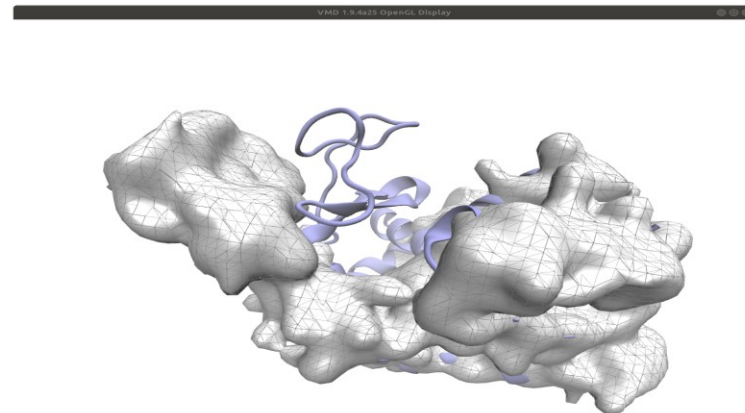
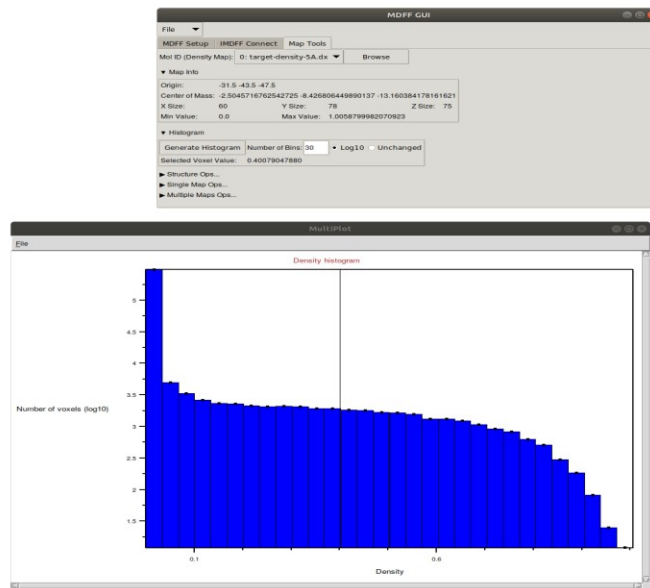
Application and Hardware platform	Runtime, Speedup vs. Chimera, VMD+GPU
Chimera Xeon E5-2687W (2 socket) [1]	15.860s, 1x
VMD-CUDA IBM Power8 + 1x Tesla K40 [2]	0.488s, 32x 0.9x
VMD-CUDA Intel Xeon E5-2687W + 1x Quadro K6000 [1,2]	0.458s, 35x 1.0x
VMD-CUDA Intel Xeon E5-2698v3 + 1x Tesla P100	0.090s, 176x 5.1x
VMD-CUDA IBM Power8 “Minsky” + 1x Tesla P100	0.080s, 198x 5.7x
VMD-CUDA Intel Xeon E5-2697Av4 + 1x Tesla V100	0.050s, 317x 9.2x
VMD-CUDA IBM Power9 “Newell” + 1x Tesla V100	0.049s, 323x 9.3x

[1] GPU-Accelerated Analysis and Visualization of Large Structures Solved by Molecular Dynamics Flexible Fitting. J. E. Stone, R. McGreevy, B. Isralewitz, and K. Schulten. Faraday Discussions 169:265-283, 2014.

[2] Early Experiences Porting the NAMD and VMD Molecular Simulation and Analysis Software to GPU-Accelerated OpenPOWER Platforms. J. E. Stone, A.-P. Hynninen, J. C. Phillips, K. Schulten. International Workshop on OpenPOWER for HPC (IWOPH'16), LNCS 9945, pp. 188-206, 2016.

New VMD MDFF Density Map Tools

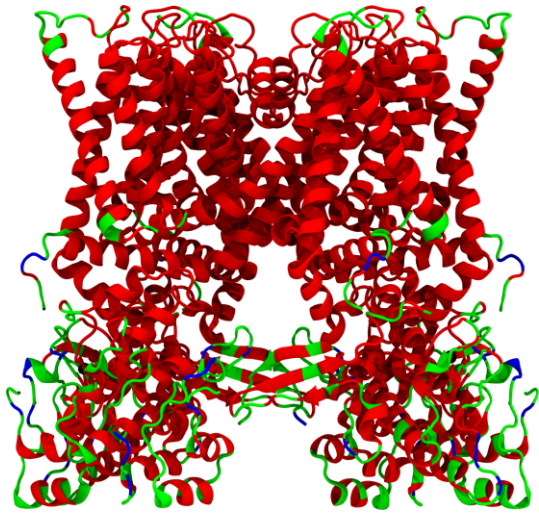
- **New** Map Tools tab of MDFF GUI provides wide array of density map manipulation tools including:
 - **New** Rigid Body Fitting
 - **New** Interactive Histogram
 - Trim, Crop, Clamp, Smooth...
 - **Easy Masking routine**
- **New** Density Segmentation
- Add, subtract, multiply maps
- Cross correlation and potential calculations for MDFF



New Density Map Tools - Masking

Easily select and mask density map regions with VMD selection language

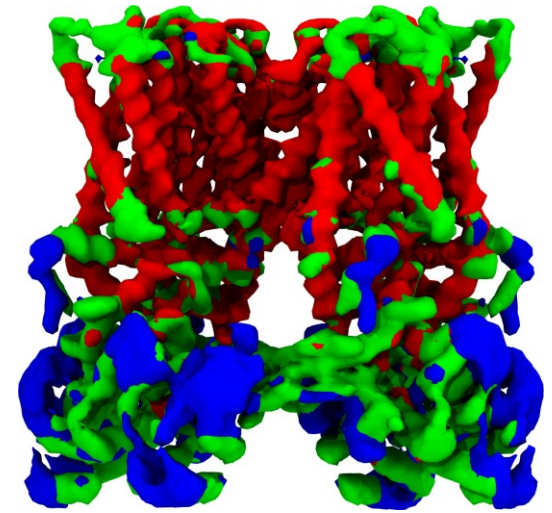
TRPV1 structure (3J5P) and cryo-EM density (emd-5778) colored by local resolution obtained by ResMap



High Res
(~3 Å)

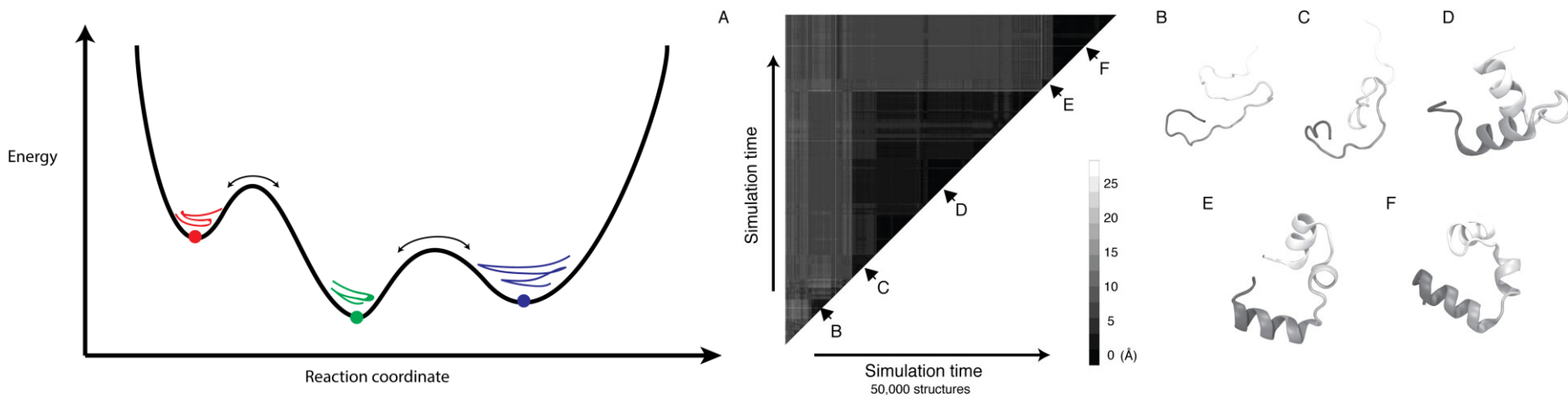
Med Res
(~4 Å)

Low Res
(~5 Å)



A. Kucukelbir, F.J. Sigworth, H.D. Tagare, Quantifying the Local Resolution of Cryo-EM Density Maps, Nature Methods, Volume 11, Issue 1, Pages 63-65, 2014.

Clustering Analysis of Molecular Dynamics Trajectories: Requires I/O+Memory for All-Pairs of Trajectory Frames



GPU-Accelerated Molecular Dynamics Clustering Analysis with OpenACC. J.E. Stone, J.R. Perilla, C. K. Cassidy, and K. Schulten. In, Robert Farber, ed., *Parallel Programming with OpenACC*, Morgan Kaufmann, Chapter 11, pp. 215-240, 2016.

Use of Node-Local Burst Buffers and Non-Volatile Memory DIMMs

- **Perform viz+analysis in-transit in node-local SSDs, persistent memory NVDIMMs**
- ORNL Summit I/O:
 - Parallel FS: 2.5 TB/s
 - **Node-local PCIe “burst buffer” SSDs: 10+ TB/sec, 7PB capacity**
- Plenty of capacity for full-detail MD trajectories, **could enable ~100x increase in temporal resolution** in cases where it would be valuable to the science
- **Enable all-pairs trajectory clustering analyses and resulting visualizations**
- Future systems with NVDIMMs (3D Xpoint, phase change memory) could eventually provide bandwidths approaching DRAM
- Use NVDIMMs w/ **mmap()**, **APIs like PMDK** to perform formerly-out-of-core calculations using persistent memory:
<https://github.com/pmem/pmdk>
- **Imagine future Summit-like machines w/ NVLink-connected GPUs w/ access to high-bandwidth persistent memory on each node**

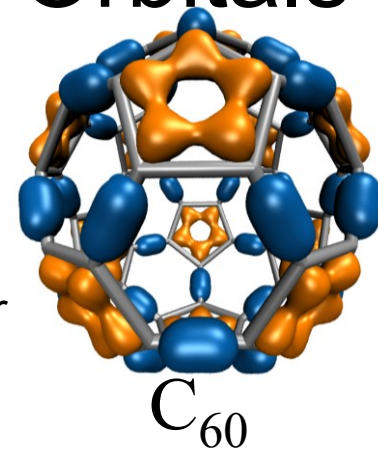
Trade FLOPS for Reduced I/O

ORNL Summit compute node:

- 6x Tesla V100 GPUs, 2x POWER9 CPUs
- GPUs Peak: ~46 DP TFLOPS, ~96 SP TFLOPS
- Peak IB rate per node: ~23GB/sec
- **Ratio of FLOPS vs. I/O:**
 - ~2,000 DP FLOPS/byte, ~4000 SP FLOPS/byte
 - ~16K FLOPS per FP word**

Unconventional approach: Recompute to avoid I/O

Computing+Visualizing Molecular Orbitals



- Movies of simulation trajectories provides insight into results
- QM, and hybrid (QM/MM) MO visualizations historically done from huge “cube” files, impractical
- Store QM wavefunctions + Gaussian basis set, only 10s of KB per stored timestep compared to 100s of MB
- **Recompute MO grid on-the-fly from QM basis set, huge decrease in RAM+I/O in exchange for heavy FP arithmetic**

NAMD goes quantum: An integrative suite for hybrid simulations. Melo, M. C. R.; Bernardi, R. C.; Rudack T.; Scheurer, M.; Riplinger, C.; Phillips, J. C.; Maia, J. D. C.; Rocha, G. D.; Ribeiro, J. V.; Stone, J. E.; Neese, F.; Schulten, K.; Luthey-Schulten, Z.; **Nature Methods**, 2018.

<http://dx.doi.org/10.1038/nmeth.4638>

High Performance Computation and Interactive Display of Molecular Orbitals on GPUs and Multi-core CPUs. J. E. Stone, J. Saam, D. Hardy, K. Vandivort, W. Hwu, K. Schulten, *2nd Workshop on General-Purpose Computation on Graphics Processing Units (GPGPU-2)*, ACM International Conference Proceeding Series, volume 383, pp. 9-18, 2009.

MO Kernel for One Grid Point (Naive C)

```
...
for (at=0; at<numatoms; at++) {
    int prim_counter = atom_basis[at];
    calc_distances_to_atom(&atompos[at], &xdist, &ydist, &zdist, &dist2, &xdiv);
    for (contracted_gto=0.0f, shell=0; shell < num_shells_per_atom[at]; shell++) {
        int shell_type = shell_symmetry[shell_counter];
        for (prim=0; prim < num_prim_per_shell[shell_counter]; prim++) {
            float exponent      = basis_array[prim_counter    ];
            float contract_coeff = basis_array[prim_counter + 1];
            contracted_gto += contract_coeff * expf(-exponent*dist2);
            prim_counter += 2;
        }
        for (tmpshell=0.0f, j=0, zdp=1.0f; j<=shell_type; j++, zdp*=zdist) {
            int imax = shell_type - j;
            for (i=0, ydp=1.0f, xdp=pow(xdist, imax); i<=imax; i++, ydp*=ydist, xdp*=xdiv)
                tmpshell += wave_f[ifunc++] * xdp * ydp * zdp;
        }
        value += tmpshell * contracted_gto;
        shell_counter++;
    }
}
}.....
```

Loop over atoms

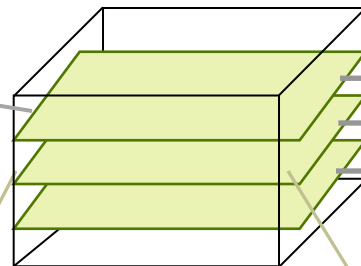
Loop over shells

Loop over primitives:
largest component of
runtime, due to **expf()**

Loop over angular
momenta
(unrolled in real code)

MO GPU Parallel Decomposition

MO 3-D lattice decomposes into 2-D slices (CUDA grids)

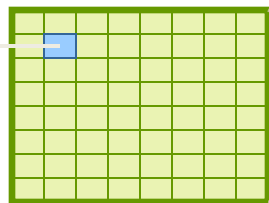


...
GPU 2
GPU 1
GPU 0

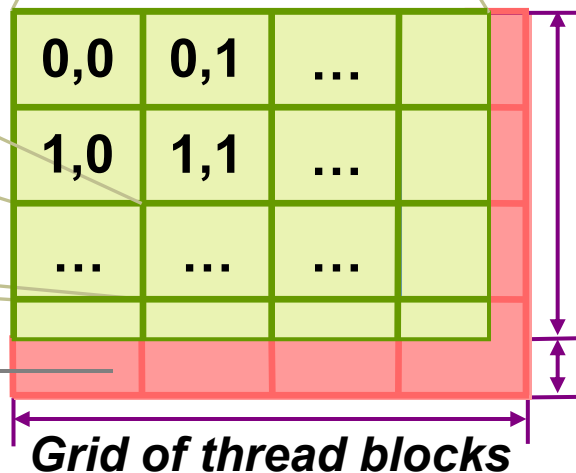
Lattice computed using multiple GPUs

Small 8x8 thread blocks afford large per-thread register count, shared memory

Each thread computes one MO lattice point.



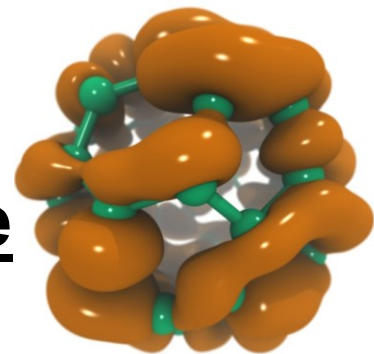
Padding optimizes global memory performance, guaranteeing coalesced global memory accesses



Threads producing results that are used

Threads producing results that are discarded

VMD C₆₀ MO Viz. Perf, 516x519x507 Grid: @ .13s/frame, avoids 3.8GB/s I/O per-node



Hardware platform	Runtime,	Speedup
IBM Power8 (ORNL 'crest') + 1x Tesla K40 [1]	3.49s,	1.0x
Intel Xeon E5-2697Av4 + 1x Tesla V100	0.610s,	5.7x
Intel Xeon E5-2697Av4 + 2x Tesla V100	0.294s,	11.8x
Intel Xeon E5-2697Av4 + 3x Tesla V100	0.220s,	15.9x
IBM Power9 "Newell" + 1x Tesla V100	0.394s,	8.8x
IBM Power9 "Newell" + 2x Tesla V100	0.207s,	16.8x
IBM Power9 "Newell" + 3x Tesla V100	0.151s,	23.1x
IBM Power9 "Newell" + 4x Tesla V100	0.130s,	26.8x

NVLink perf.
boost w/ no
code tuning
(YET)

[1] Early Experiences Porting the NAMD and VMD Molecular Simulation and Analysis Software to GPU-Accelerated OpenPOWER Platforms. J. E. Stone, A.-P. Hynninen, J. C. Phillips, K. Schulten. International Workshop on OpenPOWER for HPC (IWOPH'16), LNCS 9945, pp. 188-206, 2016.

MO Kernel Structure, Opportunity for NVRTC JIT...

Data-driven execution, but representative loop trip counts in (...)

Loop over atoms (1 to ~200) {

Loop over electron shells for this atom type (1 to ~6) {

Loop over primitive functions for this shell type (1 to ~6) {

Small loop trip counts result in significant loop overhead. **Runtime kernel generation and NVRTC JIT compilation can achieve in a large (1.8x!) speed boost via loop unrolling, constant folding, elimination of array accesses, ...**

Loop over angular momenta for this shell type (1 to ~15) {}

}

}

Molecular Orbital Computation and Display Process

Runtime Kernel Generation, NVRTC Just-In-Time (JIT) Compilation

**One-time
initialization**

**Initialize Pool of GPU
Worker Threads**

Read QM simulation log file, trajectory

Preprocess MO coefficient data
eliminate duplicates, sort by type, etc...

Generate/compile basis set-specific CUDA kernel

For current frame and MO index,
retrieve MO wavefunction coefficients

**Compute 3-D grid of MO wavefunction amplitudes
using basis set-specific CUDA kernel**

Extract isosurface mesh from 3-D MO grid

Render the resulting surface

**For each trj frame,
for each MO shown**

```
for (shell=0; shell < maxshell; shell++) {  
    float contracted_gto = 0.0f;  
  
    // Loop over the Gaussian primitives of CGTO  
    int maxprim = const_num_prim_per_shell[shell_counter];  
    int shell_type = const_shell_symmetry[shell_counter];  
    for (prim=0; prim < maxprim; prim++) {  
        float exponent = const_basis_array[prim_counter];  
        float contract_coeff = const_basis_array[prim_counter + 1];  
        contracted_gto += contract_coeff * expf(-exponent*dist2);  
        prim_counter += 2;  
    }  
  
    contracted_gto = 1.832937 * expf(-7.868272*dist2);  
    contracted_gto += 1.405380 * expf(-1.881289*dist2);  
    contracted_gto += 0.701383 * expf(-0.544249*dist2);  
}
```

General loop-based
data-dependent MO
CUDA kernel



Runtime-generated data-
specific MO CUDA kernel
compiled via **CUDA**
NVRTC JIT...



1.8x Faster

```
for (shell=0; shell < maxshell; shell++) {
```

```
float contracted_gto = 0.0f;
```

```
// Loop over the Gaussian primitives of CGTO
```

```
int maxprim = const_num_prim_per_shell[shell_counter];
```

```
int shell_type = const_shell_symmetry[shell_counter];
```

```
for (prim=0; prim < maxprim; prim++) {
```

```
float exponent = const_basis_array[prim_counter];
```

```
float contract_coeff = const_basis_array[prim_counter + 1];
```

```
contracted_gto += contract_coeff * expf(-exponent*dist2);
```

```
prim_counter += 2;
```

```
}
```

```
float tmpshell=0;
```

```
switch (shell_type) {
```

```
case S_SHELL:
```

```
value += const_wave_f[ifunc++] * contracted_gto;
```

```
break;
```

```
[.....]
```

```
case D_SHELL:
```

```
tmpshell += const_wave_f[ifunc++] * xdist2;
```

```
tmpshell += const_wave_f[ifunc++] * ydist2;
```

```
tmpshell += const_wave_f[ifunc++] * zdist2;
```

```
tmpshell += const_wave_f[ifunc++] * xdist * ydist;
```

General loop-based
data-dependent MO

CUDA kernel



Runtime-generated data-
specific MO CUDA
kernel compiled via
CUDA NVRTC JIT...



1.8x Faster

```
contracted_gto = 1.832937 * expf(-7.868272*dist2);  
contracted_gto += 1.405380 * expf(-1.881289*dist2);  
contracted_gto += 0.701383 * expf(-0.544249*dist2);
```

```
// P_SHELL
```

```
tmpshell = const_wave_f[ifunc++] * xdist;
```

```
tmpshell += const_wave_f[ifunc++] * ydist;
```

```
tmpshell += const_wave_f[ifunc++] * zdist;
```

```
value += tmpshell * contracted_gto;
```

```
contracted_gto = 0.187618 * expf(-0.168714*dist2);
```

```
// S_SHELL
```

```
value += const_wave_f[ifunc++] * contracted_gto;
```

```
contracted_gto = 0.217969 * expf(-0.168714*dist2);
```

```
// P_SHELL
```

```
tmpshell = const_wave_f[ifunc++] * xdist;
```

```
tmpshell += const_wave_f[ifunc++] * ydist;
```

```
tmpshell += const_wave_f[ifunc++] * zdist;
```

```
value += tmpshell * contracted_gto;
```

```
contracted_gto = 3.858403 * expf(-0.800000*dist2);
```

```
// D_SHELL
```

```
tmpshell = const_wave_f[ifunc++] * xdist2;
```

```
tmpshell += const_wave_f[ifunc++] * ydist2;
```

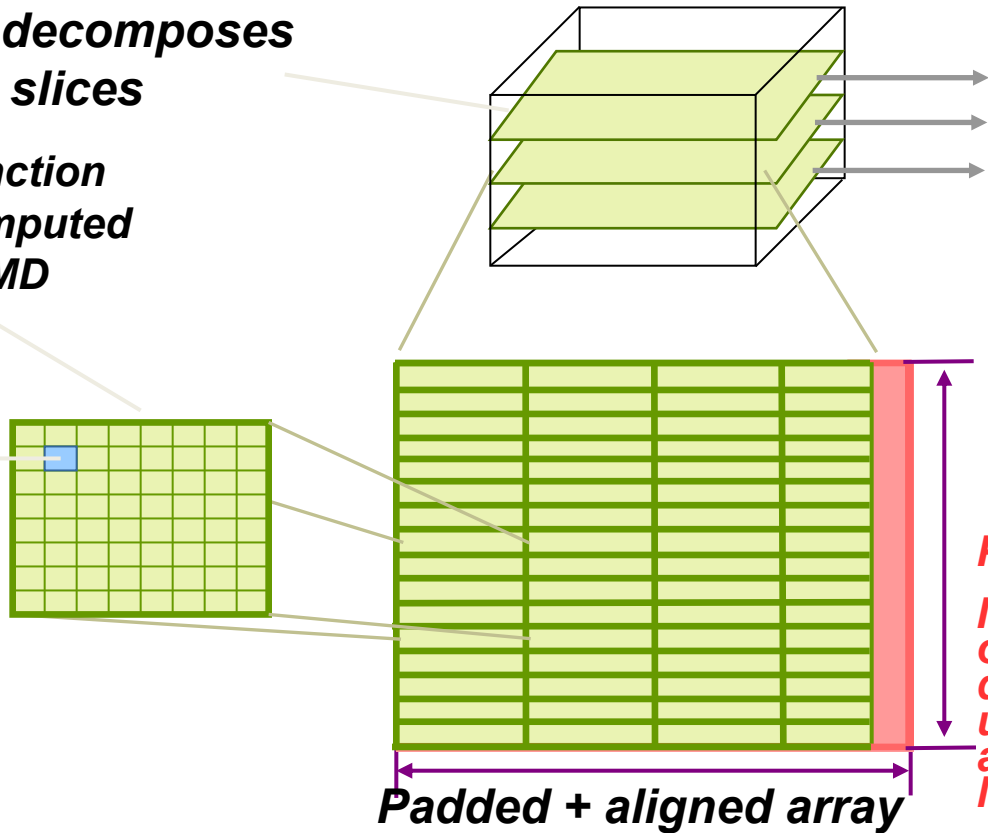
MO CPU Parallel Decomposition

AVX-512 Kernels on KNL

MO 3-D lattice decomposes into 2-D slices

Vectors of wavefunction amplitudes are computed using hardware SIMD instructions

Each CPU thread computes 1, 4, 8, 16 MO lattice points per loop iteration: C, SSE, AVX2 or AVX-512ER



...
Thread 2
Thread 1
Thread 0

Lattice decomposed across many CPU threads

SIMD lanes producing results that are used

Padding:
Inactive SIMD lanes or region of discarded output used to guarantee aligned vector loads+stores

Padded + aligned array

AVX-512ER MO CGTO Loop

```
int maxprim = num_prim_per_shell[shell_counter];
int shelltype = shell_types[shell_counter];
for (prim=0; prim<maxprim; prim++) {
    float exponent      = basis_array[prim_counter    ];
    float contract_coeff = basis_array[prim_counter + 1];

    // contracted_gto += contract_coeff * exp(exponent*dist2);
    __m512 expval = _mm512_mul_ps(_mm512_set1_ps(exponent * MLOG2EF), dist2);

    // expf() approximation required, use (base-2) AVX-512ER instructions...
    __m512 retval = _mm512_exp2a23_ps(expval);
```


Performance of AVX-512ER Intrinsics vs. Autovectorization on KNL: Small 172x173x169 Grid

- Intel C++ '15 autovectorization (**fail**): **220+ sec**
- Hand-coded SSE2 w/ existing thread scheme: **48.5 sec**
- Hand-coded AVX-512ER w/ existing thread scheme: **6.3 sec**
- Hand-coded AVX-512ER, refactoring thread pool: **0.2 sec**
- **Hand-coded AVX-512ER tuned thread pool: 0.131 sec**
- **Hand-coded AVX-512ER+FMA tweaks: 0.107 sec**

Further improvement will require attention to details of cache behaviour and further tuning of low-level threading constructs for Xeon Phi/KNL

Convenient Packaging of HPC- Optimized Versions of VMD and NAMD

Making Our Research Tools Easily Accessible

- Docker “container” images available in NVIDIA NGC registry
 - Users obtain Docker images via registry, download and run on the laptop, workstation, cloud, or supercomputer of their choosing
 - <https://ngc.nvidia.com/registry/>
 - <https://ngc.nvidia.com/registry/hpc-vmd>
- Cloud based deployment
 - Full virtual machines (known as “AMI” in Amazon terminology)
 - Amazon AWS EC2 GPU-accelerated instances:
<http://www.ks.uiuc.edu/Research/cloud/>



Clusters, Supercomputers

Workstations,
Servers,
Cloud



Molecular dynamics-based refinement and validation for sub-5 Å cryo-electron microscopy

maps. Abhishek Singharoy, Ivan Teo, Ryan McGreevy, John E. Stone, Jianhua Zhao, and Klaus Schulten. *eLife*, 10.7554/eLife.16105, 2016. (66 pages).

QwikMD-integrative molecular dynamics toolkit for novices and experts. Joao V. Ribeiro, Rafael C. Bernardi, Till Rudack, John E. Stone, James C. Phillips, Peter L. Freddolino, and Klaus Schulten. *Scientific Reports*, 6:26536, 2016.

High performance molecular visualization: In-situ and parallel rendering with EGL. John E. Stone, Peter Messmer, Robert Sisneros, and Klaus Schulten. *2016 IEEE International Parallel and Distributed Processing Symposium Workshop (IPDPSW)*, pp. 1014-1023, 2016.



VMD / NAMD / LM, NGC Containers



Registry

Configuration

Documentation

User Forum

System Status

Registry

Get API Key



Documentation

How to use NGC containers on supported platforms >

Repositories

nvidia ^

hpc v

- candle
- gamess
- gromacs
- lammps
- lattice-microbes
- namd
- relion
- vmd

nvidia-hpcvis v

- index
- paraview-holodeck
- paraview-index
- paraview-optim

hpc/vmd

```
docker pull nvcr.io/hpc/vmd:cuda9-ubuntu1604-egl-1.9.4a17
```

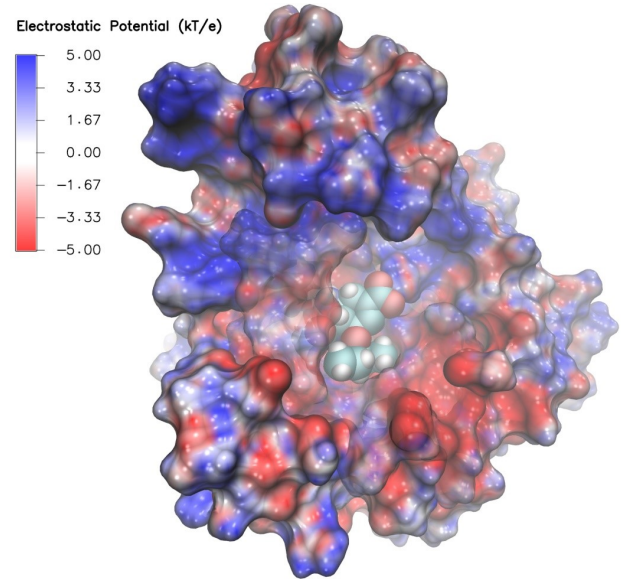
VMD

VMD is designed for modeling, visualization, and analysis of biomolecular systems such as proteins, nucleic acids, lipid membranes, carbohydrate structures, etc. VMD provides a wide variety of graphical representations for visualizing and coloring molecular structures: molecular surfaces, space-filling CPK spheres and cylinders, licorice bonds, backbone tubes and ribbons, secondary structure cartoons, and others.

VMD can be used to animate and analyze the trajectory of a molecular dynamics (MD) simulation. In particular, VMD can act as a graphical front end for an external MD program by

VMD OptiX/EGL NGC Container

- <https://ngc.nvidia.com/registry/>
- **CUDA-accelerated** viz+analysis
- **EGL off-screen rendering** – no windowing system needed
- **OptiX high-fidelity GPU ray tracing engine** built in
- All dependencies included
- **Easy to deploy on a wide range of GPU accelerated platforms**



High performance molecular visualization: In-situ and parallel rendering with EGL. J. E. Stone, P. Messmer, R. Sisneros, and K. Schulten. *2016 IEEE International Parallel and Distributed Processing Symposium Workshop (IPDPSW)*, pp. 1014-1023, 2016.

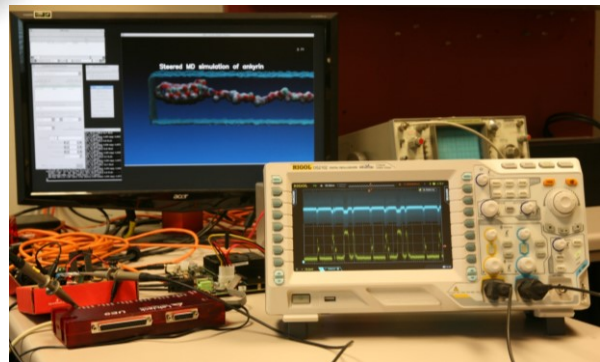
Technology Opportunities and Collaborations

- Supercomputer Centers, Cray, IBM
 - Remote visualization
 - Performance, power profiling and optimization
- NVIDIA
 - **GPU computing**
 - **Ray tracing**
 - **Remote visualization**
 - ARM, Tablets, power profiling and optimization
- Intel
 - **x86, Xeon Phi optimization, ray tracing**
- Amazon
 - Cloud deployment of VMD/NAMD, related tools
 - Remote visualization
- Universities:
 - **G. Fiorin, J. Henin, Toni Giorgino, collective variables**
 - **T. Ertl, U. Stuttgart: visualization algorithms**
 - M. Kuttel, U. Cape Town: visualization and analysis
 - W. Sherman, Indiana U.: VR HMDs, visualization

GPU computing,
Ray tracing,
Remote viz.



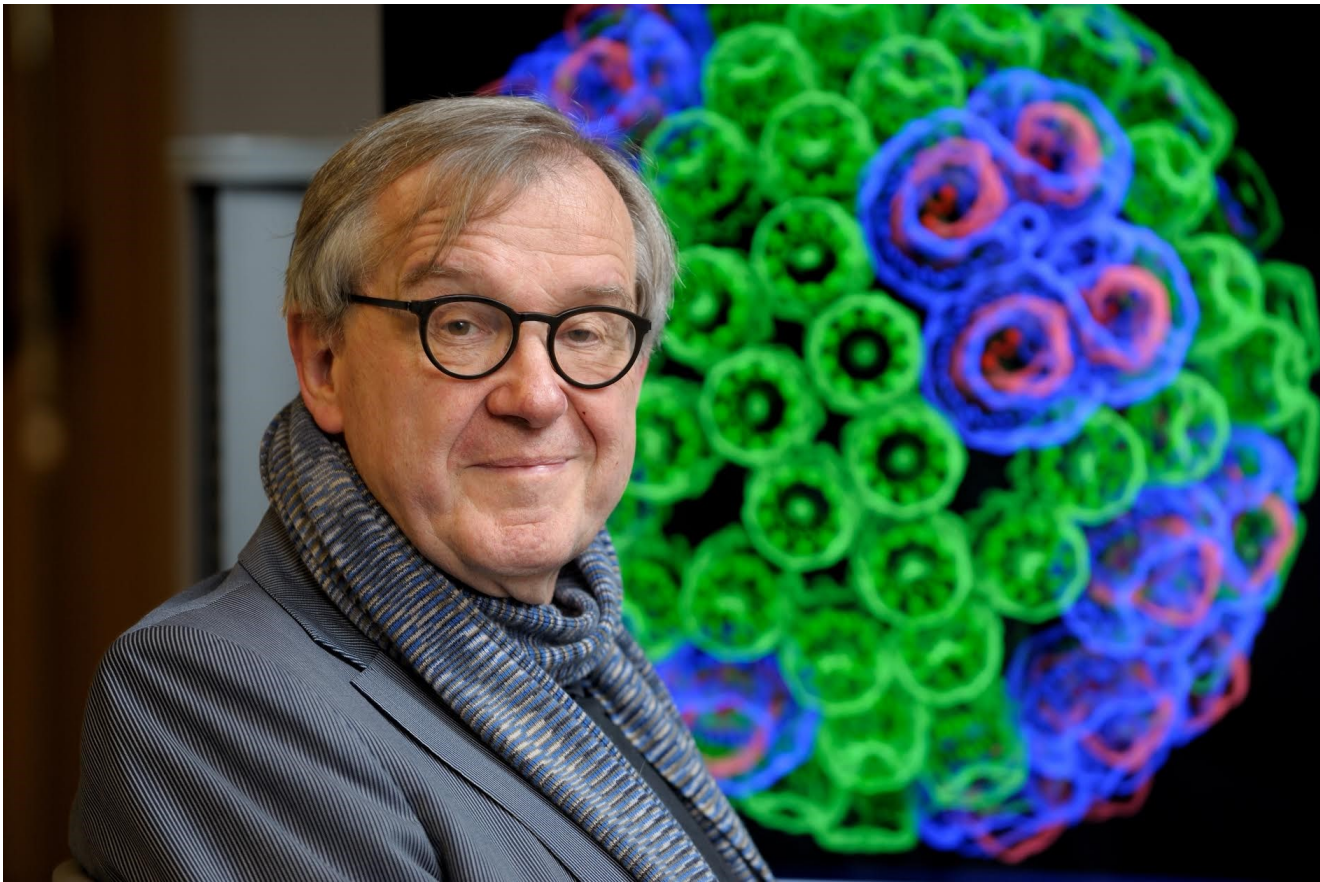
VR HMDs, 6DoF input devices



Energy efficiency: ARM+GPU

Acknowledgements

- Theoretical and Computational Biophysics Group, University of Illinois at Urbana-Champaign
- CUDA Center of Excellence, University of Illinois at Urbana-Champaign
- NVIDIA CUDA and OptiX teams
- Intel OSPRay team
- Funding:
 - DOE INCITE, ORNL Titan: DE-AC05-00OR22725
 - NSF Blue Waters:
NSF OCI 07-25070, PRAC “The Computational Microscope”,
ACI-1238993, ACI-1440026
 - NIH support: 9P41GM104601, 5R01GM098243-02



“When I was a young man, my goal was to look with mathematical and computational means at the inside of cells, one atom at a time, to decipher how living systems work. That is what I strived for and I never deflected from this goal.” – Klaus Schulten

Related Publications

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

- **NAMD goes quantum: An integrative suite for hybrid simulations.** Melo, M. C. R.; Bernardi, R. C.; Rudack T.; Scheurer, M.; Riplinger, C.; Phillips, J. C.; Maia, J. D. C.; Rocha, G. D.; Ribeiro, J. V.; Stone, J. E.; Neese, F.; Schulten, K.; Luthey-Schulten, Z.; Nature Methods, 2018. (In press)
- **Challenges of Integrating Stochastic Dynamics and Cryo-electron Tomograms in Whole-cell Simulations.** T. M. Earnest, R. Watanabe, J. E. Stone, J. Mahamid, W. Baumeister, E. Villa, and Z. Luthey-Schulten. J. Physical Chemistry B, 121(15): 3871-3881, 2017.
- **Early Experiences Porting the NAMD and VMD Molecular Simulation and Analysis Software to GPU-Accelerated OpenPOWER Platforms.** J. E. Stone, A.-P. Hynninen, J. C. Phillips, and K. Schulten. International Workshop on OpenPOWER for HPC (IWOPH'16), LNCS 9945, pp. 188-206, 2016.
- **Immersive Molecular Visualization with Omnidirectional Stereoscopic Ray Tracing and Remote Rendering.** J. E. Stone, W. R. Sherman, and K. Schulten. High Performance Data Analysis and Visualization Workshop, IEEE International Parallel and Distributed Processing Symposium Workshop (IPDPSW), pp. 1048-1057, 2016.
- **High Performance Molecular Visualization: In-Situ and Parallel Rendering with EGL.** J. E. Stone, P. Messmer, R. Sisneros, and K. Schulten. High Performance Data Analysis and Visualization Workshop, IEEE International Parallel and Distributed Processing Symposium Workshop (IPDPSW), pp. 1014-1023, 2016.
- **Evaluation of Emerging Energy-Efficient Heterogeneous Computing Platforms for Biomolecular and Cellular Simulation Workloads.** J. E. Stone, M. J. Hallock, J. C. Phillips, J. R. Peterson, Z. Luthey-Schulten, and K. Schulten. 25th International Heterogeneity in Computing Workshop, IEEE International Parallel and Distributed Processing Symposium Workshop (IPDPSW), pp. 89-100, 2016.

Related Publications

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

- **Atomic Detail Visualization of Photosynthetic Membranes with GPU-Accelerated Ray Tracing.** J. E. Stone, M. Sener, K. L. Vandivort, A. Barragan, A. Singharoy, I. Teo, J. V. Ribeiro, B. Isralewitz, B. Liu, B.-C. Goh, J. C. Phillips, C. MacGregor-Chatwin, M. P. Johnson, L. F. Kourkoutis, C. Neil Hunter, and K. Schulten. *J. Parallel Computing*, 55:17-27, 2016.
- **Chemical Visualization of Human Pathogens: the Retroviral Capsids.** Juan R. Perilla, Boon Chong Goh, John E. Stone, and Klaus Schulten. *SC'15 Visualization and Data Analytics Showcase*, 2015.
- **Visualization of Energy Conversion Processes in a Light Harvesting Organelle at Atomic Detail.** M. Sener, J. E. Stone, A. Barragan, A. Singharoy, I. Teo, K. L. Vandivort, B. Isralewitz, B. Liu, B. Goh, J. C. Phillips, L. F. Kourkoutis, C. N. Hunter, and K. Schulten. *SC'14 Visualization and Data Analytics Showcase*, 2014.
***Winner of the **SC'14 Visualization and Data Analytics Showcase**
- **Runtime and Architecture Support for Efficient Data Exchange in Multi-Accelerator Applications.** J. Cabezas, I. Gelado, J. E. Stone, N. Navarro, D. B. Kirk, and W. Hwu. *IEEE Transactions on Parallel and Distributed Systems*, 26(5):1405-1418, 2015.
- **Unlocking the Full Potential of the Cray XK7 Accelerator.** M. D. Klein and J. E. Stone. Cray Users Group, Lugano Switzerland, May 2014.
- **GPU-Accelerated Analysis and Visualization of Large Structures Solved by Molecular Dynamics Flexible Fitting.** J. E. Stone, R. McGreevy, B. Isralewitz, and K. Schulten. *Faraday Discussions*, 169:265-283, 2014.
- **Simulation of reaction diffusion processes over biologically relevant size and time scales using multi-GPU workstations.** M. J. Hallock, J. E. Stone, E. Roberts, C. Fry, and Z. Luthey-Schulten. *Journal of Parallel Computing*, 40:86-99, 2014.

Related 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.

Related 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.

Related 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. E. Stone, J. Saam, D. Hardy, K. Vandivort, W. Hwu, K. Schulten, *2nd Workshop on General-Purpose Computation on Graphics Processing 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.

Related 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.

Related 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. E. Stone, J. Saam, D. Hardy, K. Vandivort, W. Hwu, K. Schulten, *2nd Workshop on General-Purpose Computation on Graphics Processing 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.

Related 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.