

REDESIGNING DRUG DESIGN WITH OPEN SOURCE SOFTWARE



John D. Chodera

MSKCC Computational Biology Program

<http://www.choderalab.org>

Slides available at <http://www.choderalab.org>

DISCLOSURES:

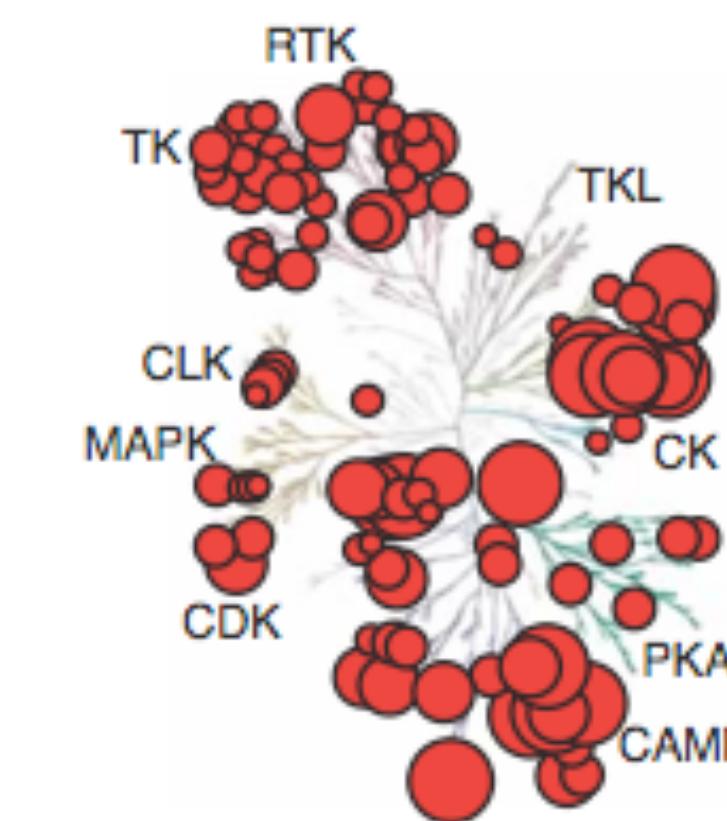
- Scientific Advisory Board, Schrödinger

SOMETIMES, DRUG DISCOVERY WORKS WELL

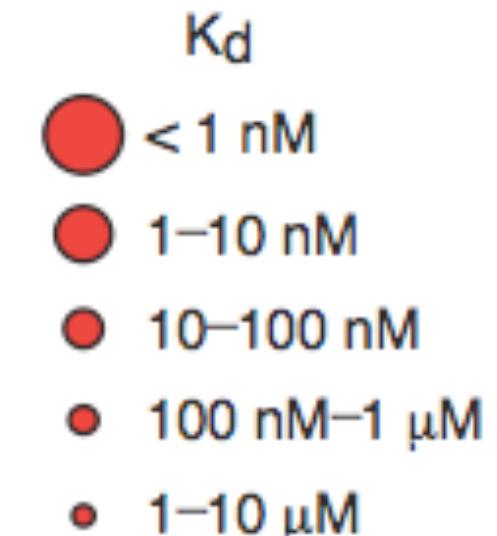
Bcr-Abl fusion
resulting in



imatinib bound to c-Abl [PDB:1IEP]



staurosporine
[toxic natural product]



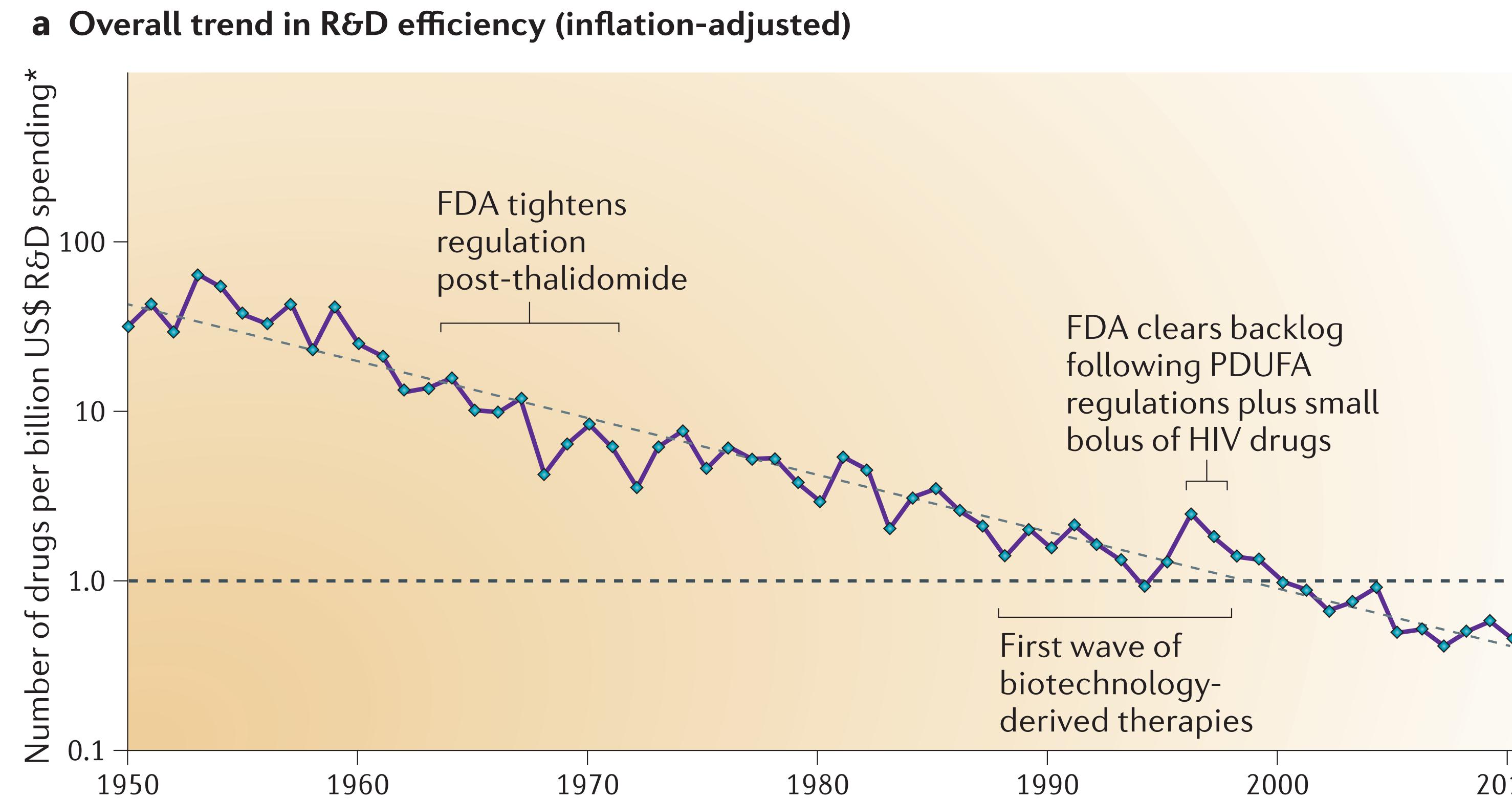
DRUG DISCOVERY USUALLY ENDS IN FAILURE

Total pharma R&D spending doubled to \$65B over 2000-2010

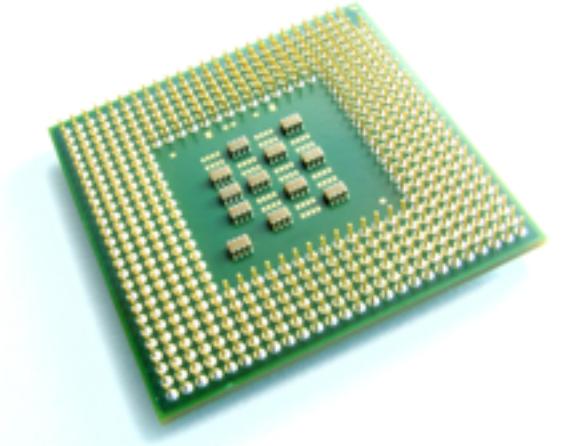
FDA approvals of new molecular entities went down by half

Number of truly innovative new molecules remained constant at 5-6/year

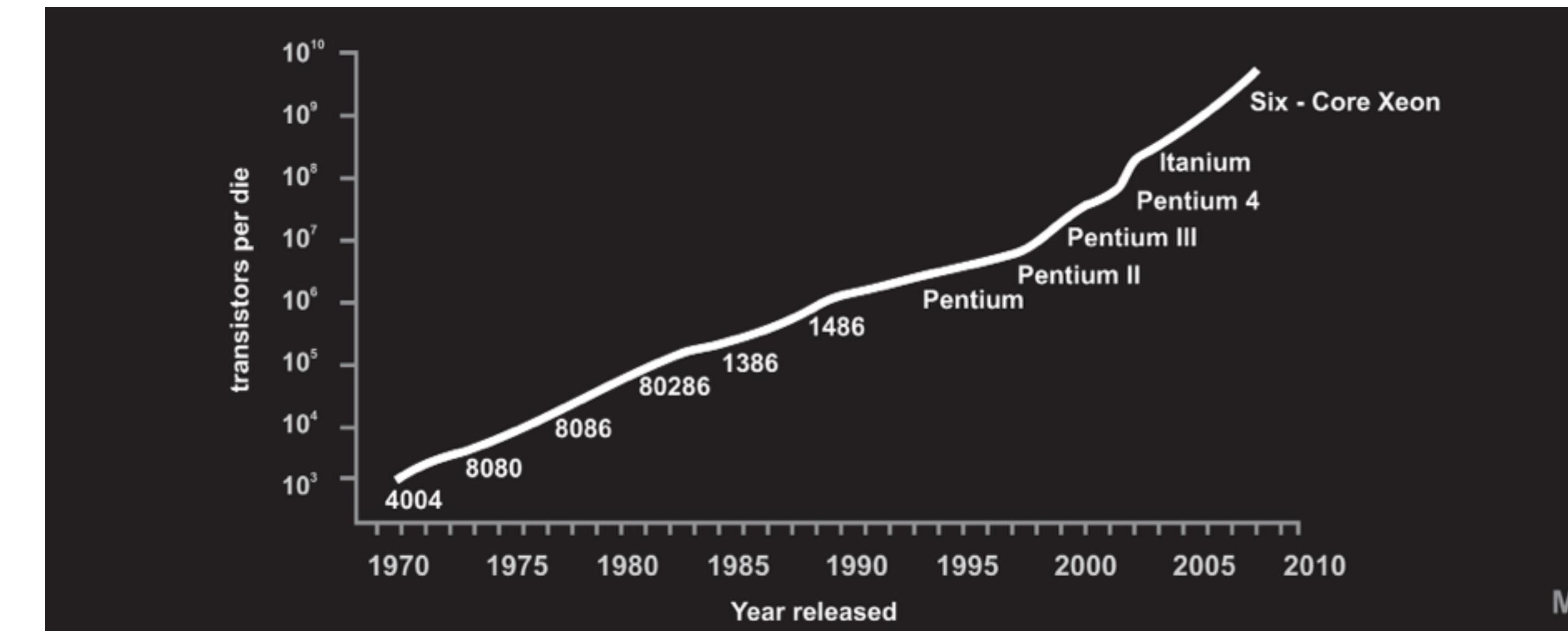
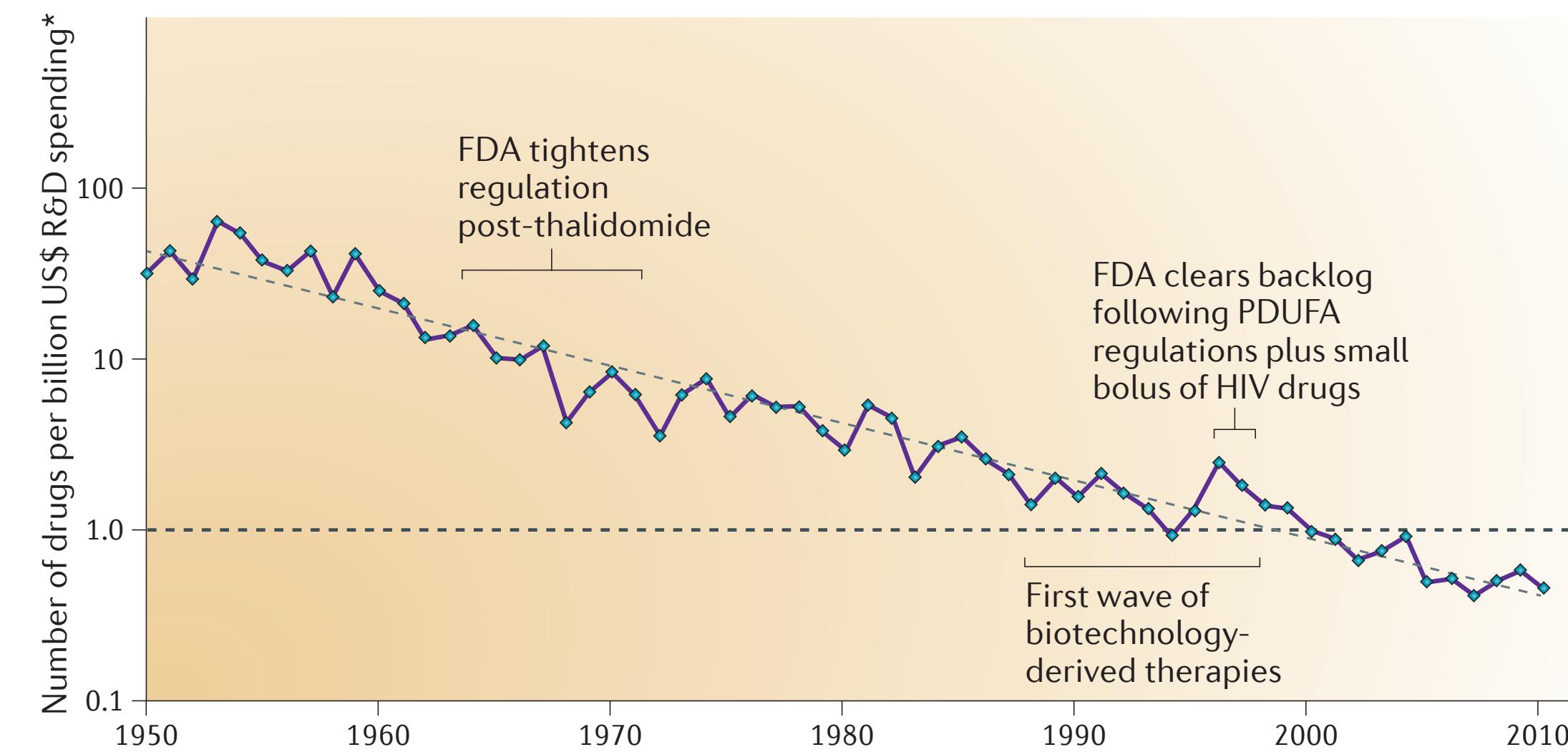
2010-2015 has seen large reductions in pharma R&D in the US



DRUG DISCOVERY USUALLY ENDS IN FAILURE



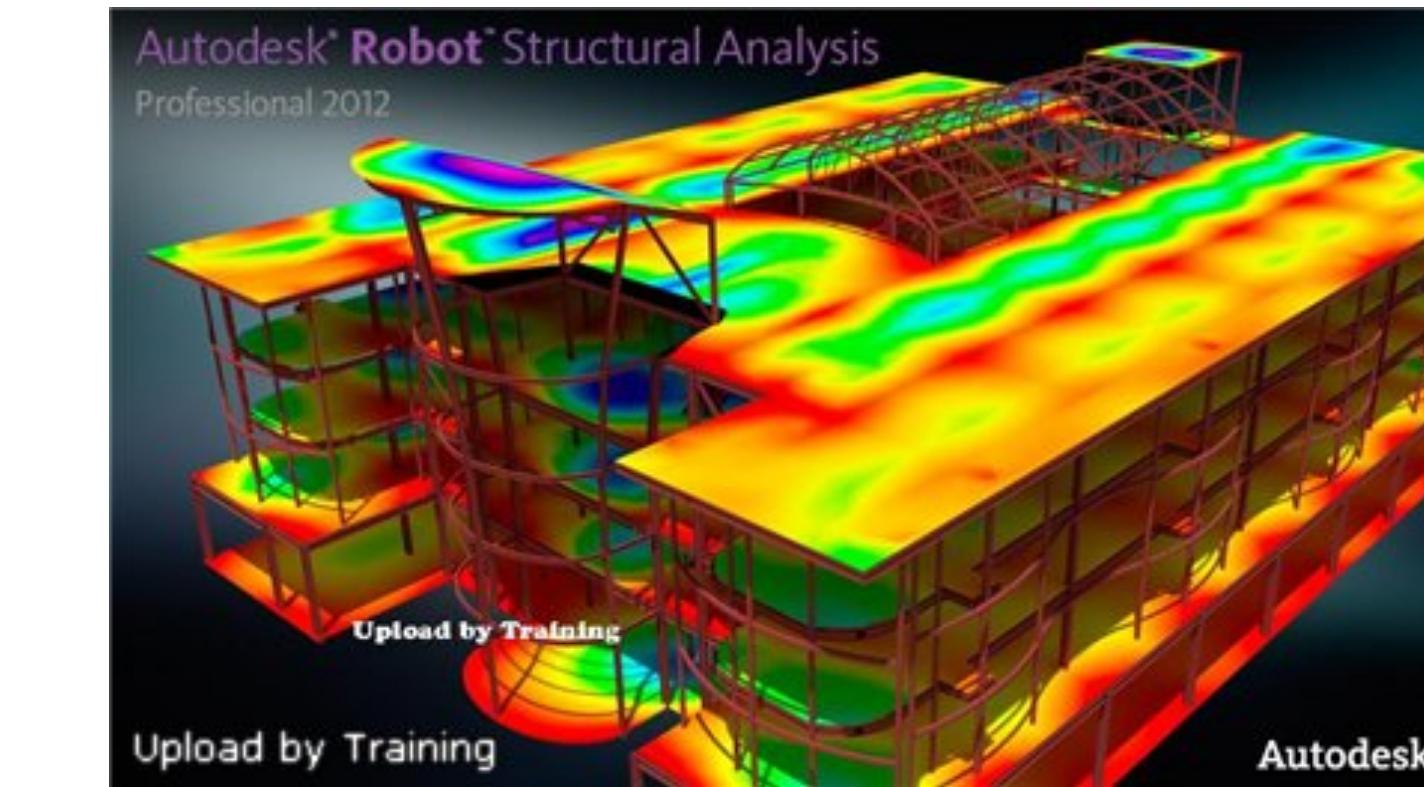
a Overall trend in R&D efficiency (inflation-adjusted)



EROOM'S LAW

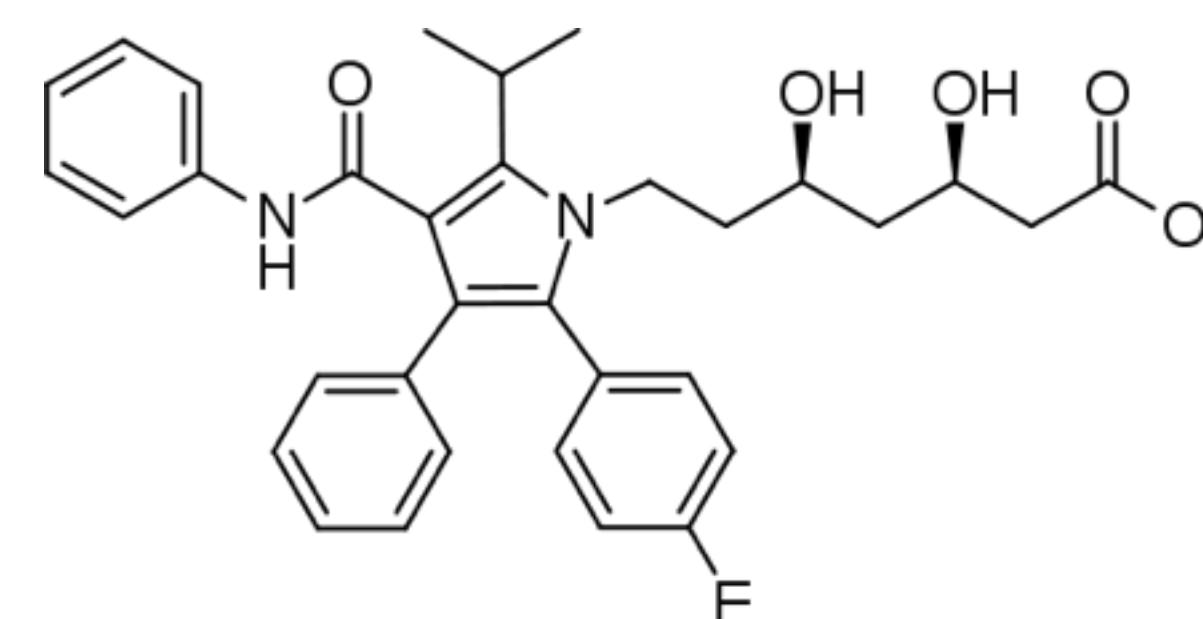
MOORE'S LAW

WE REGULARLY DESIGN PLANES, BRIDGES, AND BUILDINGS ON COMPUTERS



10^3 - 10^6 parts

WHY NOT SMALL MOLECULE DRUGS?



< 10^2 atoms

HOW CAN WE BRING DRUG DESIGN INTO THE 21ST CENTURY?



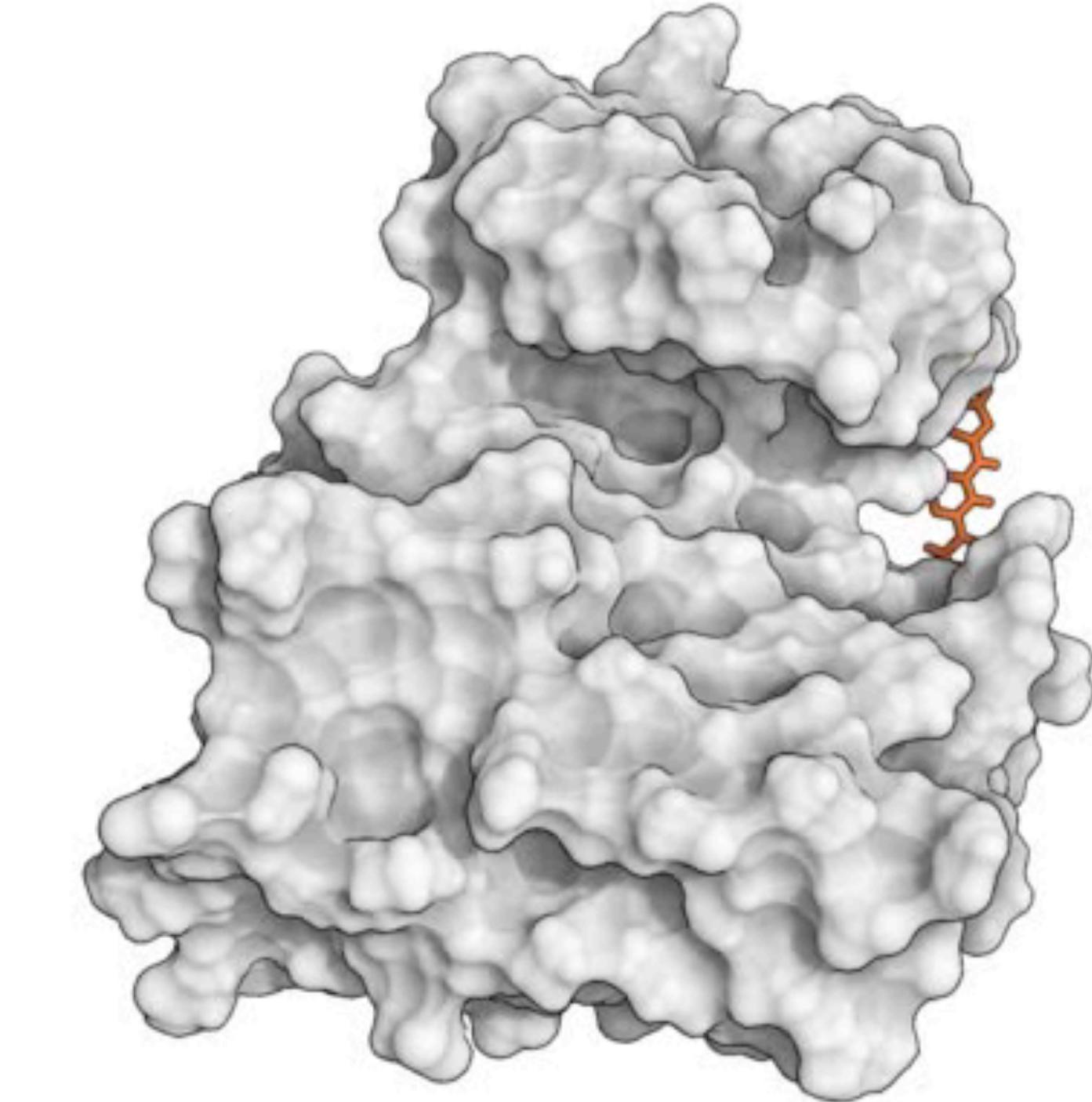
HOW CAN WE COMPUTE A BINDING AFFINITY INCLUDING RELEVANT STATISTICAL MECHANICS?

ANTON

\$50M special-purpose supercomputer from D.E. Shaw Research



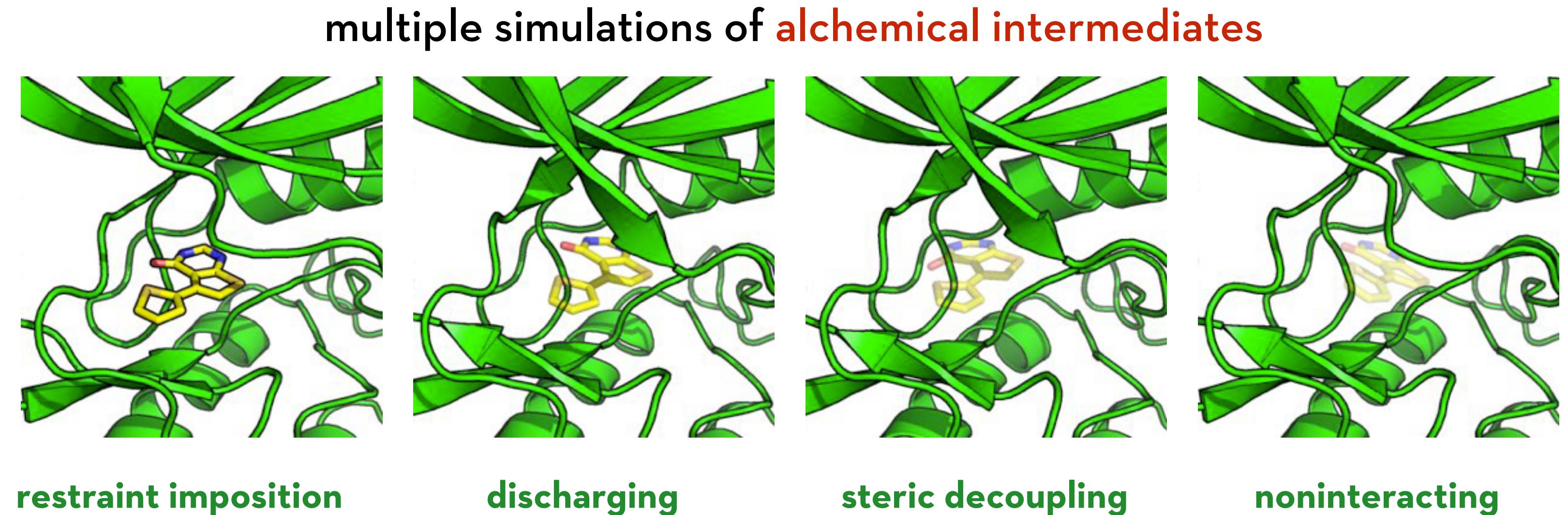
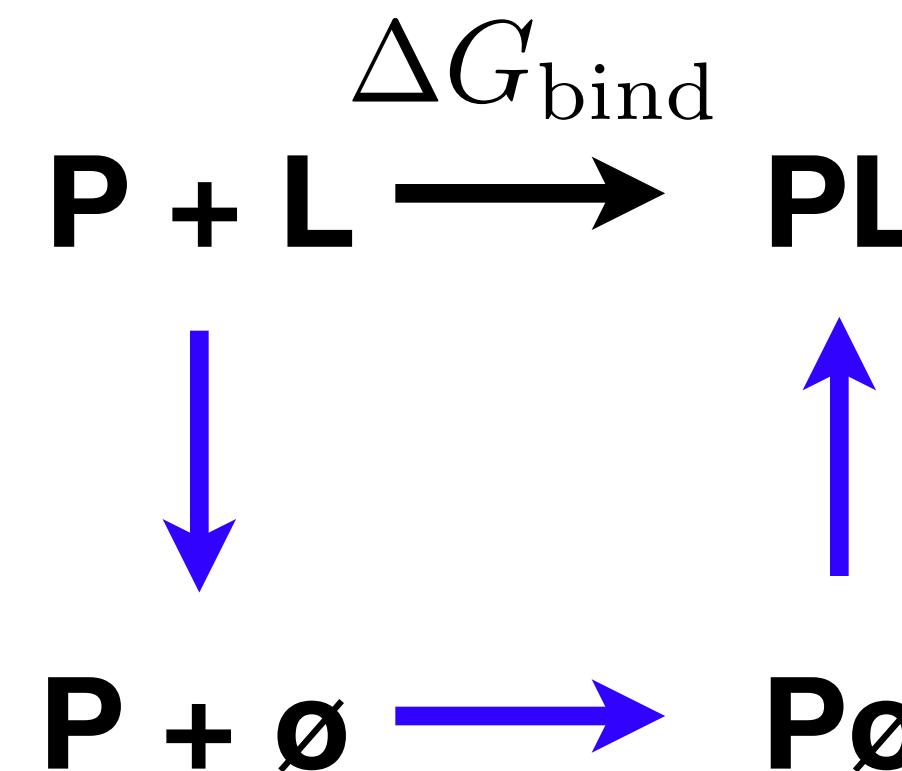
David E. Shaw



Src:dasatanib
(4 us simulation)

For typical drug off-rates (10^{-4} s^{-1}),
reliable calculation of binding affinities would require hour trajectories,
requiring $\sim 10^6$ years to simulate.

ALCHEMICAL FREE ENERGY CALCULATIONS PROVIDE A RIGOROUS WAY TO EFFICIENTLY COMPUTE BINDING AFFINITIES

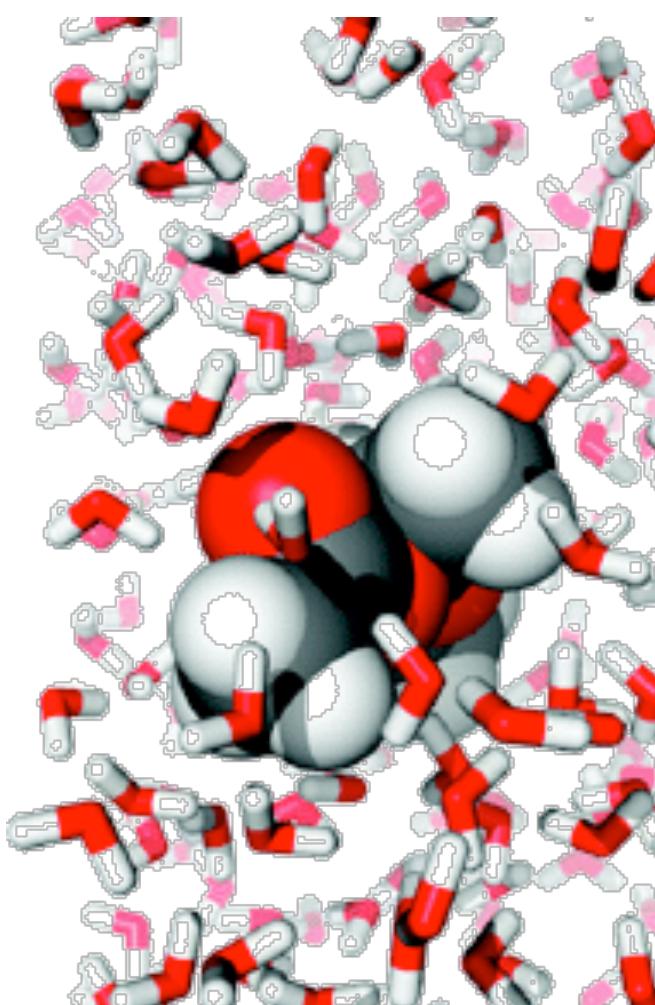


Requires **orders of magnitude** less effort than simulating direct association process, but still includes all enthalpic/entropic contributions to binding free energy.

$$\Delta F_{1 \rightarrow N} = -\beta^{-1} \ln \frac{Z_N}{Z_1} = -\beta^{-1} \ln \frac{Z_2}{Z_1} \cdot \frac{Z_3}{Z_2} \cdots \frac{Z_N}{Z_{N-1}} = \sum_{n=1}^{N-1} \Delta F_{n \rightarrow n+1} \qquad Z_n = \int d\mathbf{x} e^{-\beta U(\mathbf{x})}$$

ALCHEMICAL FREE ENERGY METHODS CAN WORK RELIABLY IN SIMPLE SYSTEMS, BUT COMPLEX SYSTEMS REMAIN CHALLENGING

model systems



hydration free energies

1.04 ± 0.03 kcal/mol (N=44)

Mobley et al. JPC B, 2007

1.23 ± 0.01 kcal/mol (N=502)

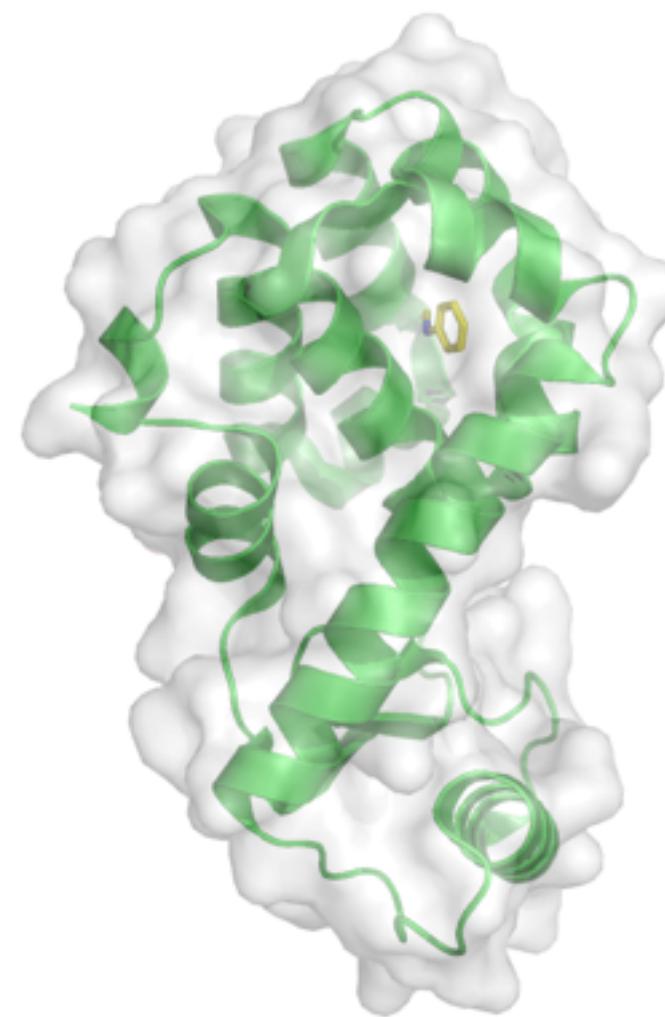
Mobley et al. JPC B 2009.

1.33 ± 0.05 kcal/mol (N=17)

Nicholls et al. J Med Chem 2008.



pharmaceutically relevant

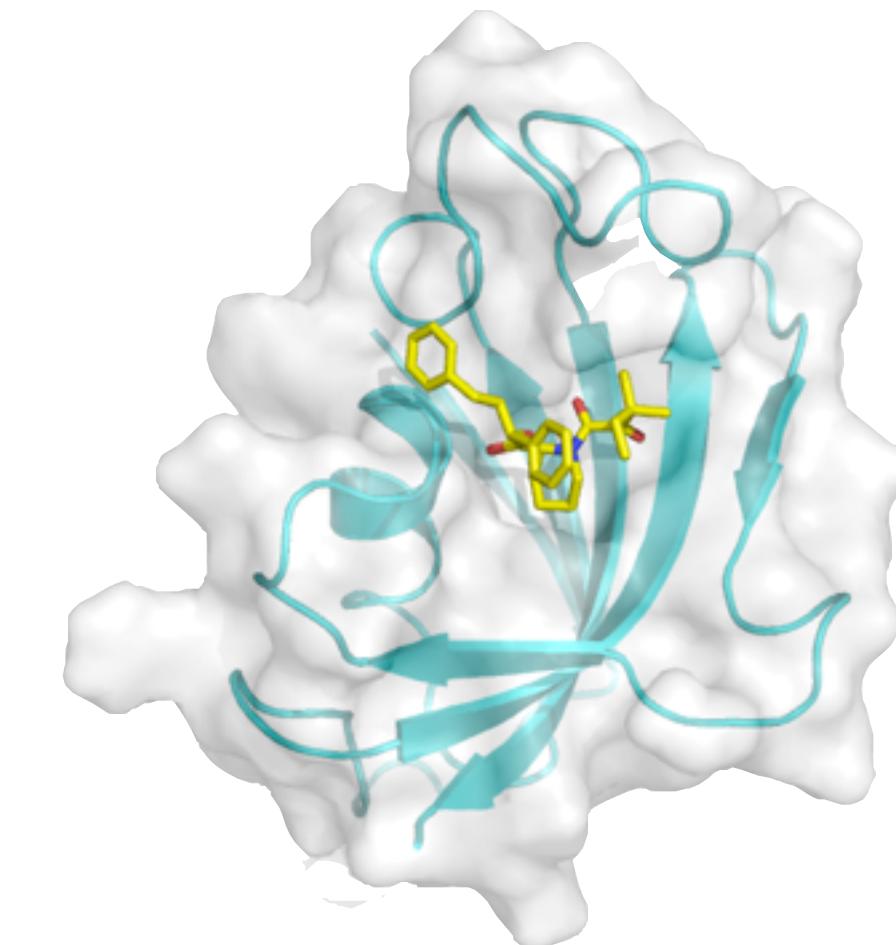


T4 lysozyme L99A

1.89 ± 0.04 kcal/mol (N=13)

Mobley et al. J Mol Biol

371:1118, 2007



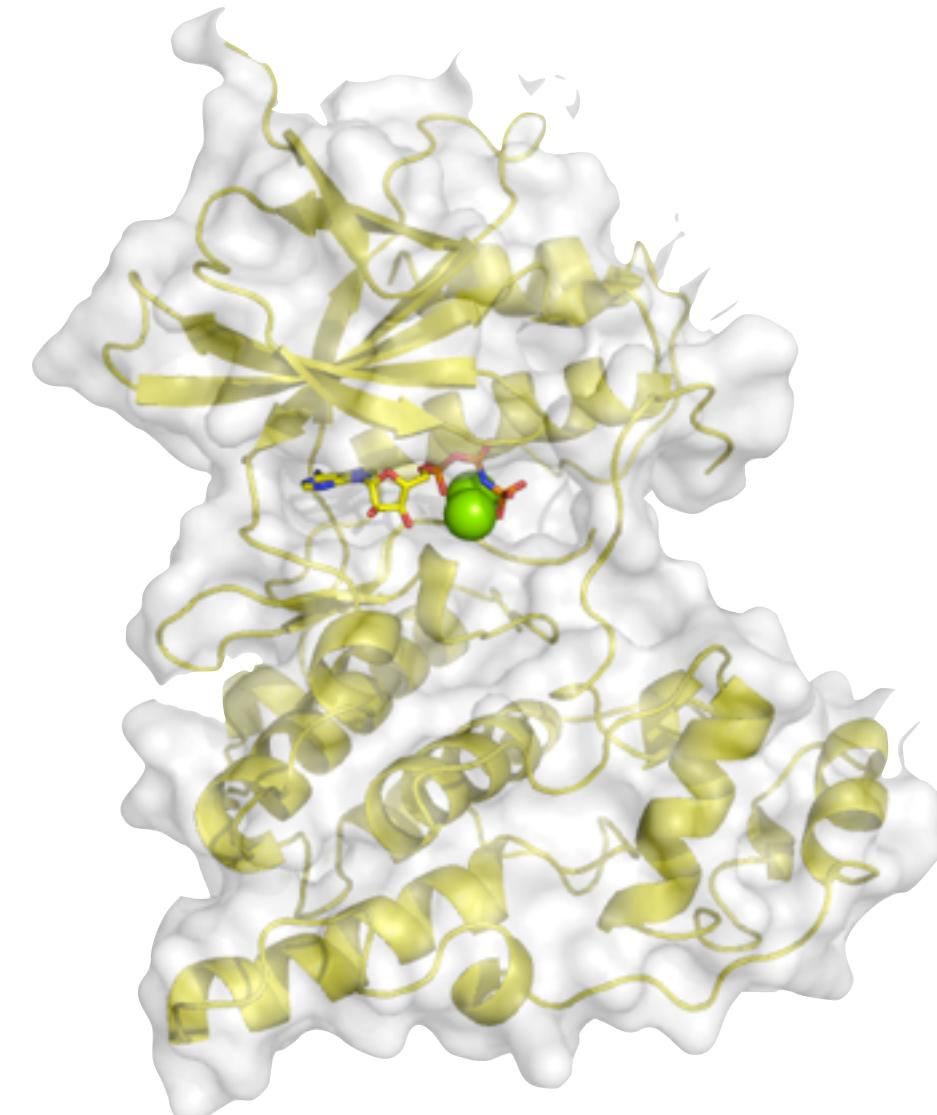
FKBP12

0.4 kcal/mol* (N=8)

Fujitani et al.

JCP 123:084108, 2005

* with 3.2 kcal/mol offset



JNK3 kinase

Anecdotal literature reports of success
(publication bias?)

Calculations are notoriously unreliable.
(e.g. SAMPL predictive challenges)

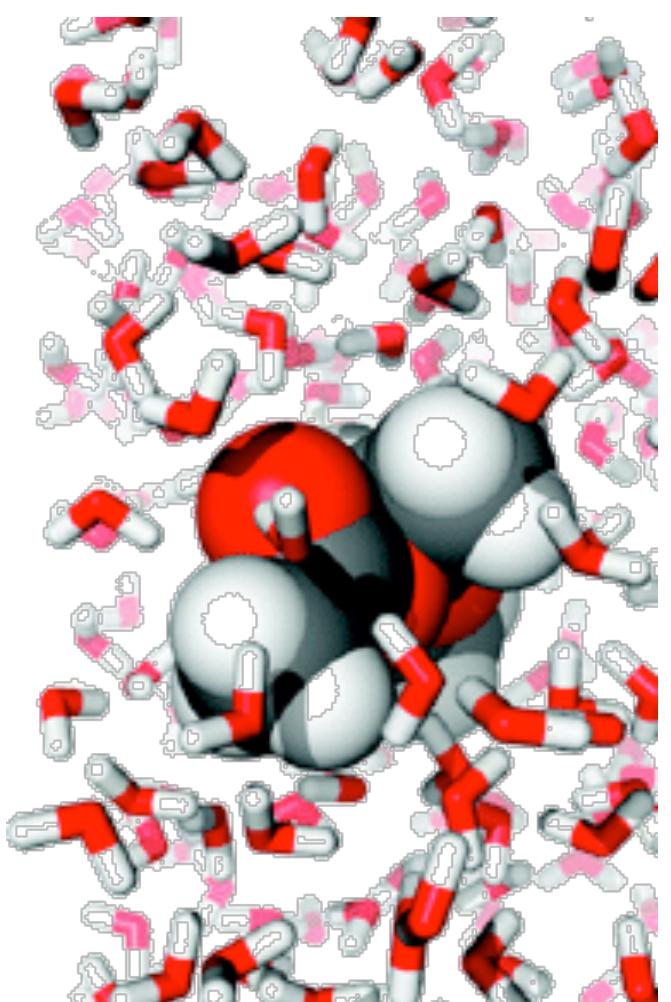
retrospective RMS error [sample size]

prospective RMS error [sample size]

(not to scale)

ALCHEMICAL FREE ENERGY METHODS CAN WORK RELIABLY IN SIMPLE SYSTEMS, BUT COMPLEX SYSTEMS REMAIN CHALLENGING

model systems

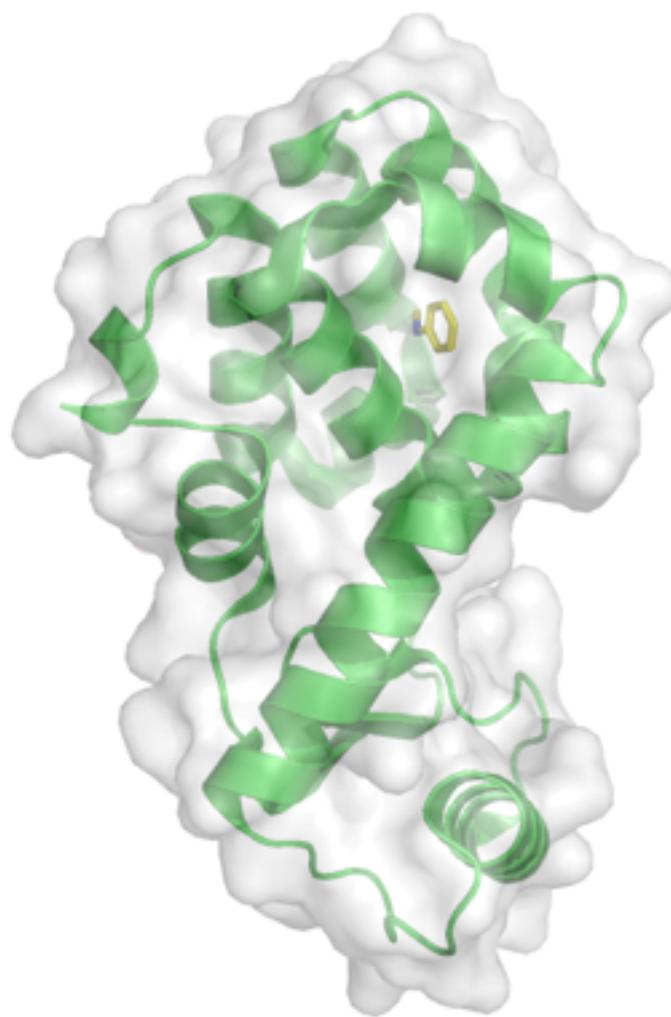


hydration free energies

solvent only
small, neutral molecules
fixed protonation states

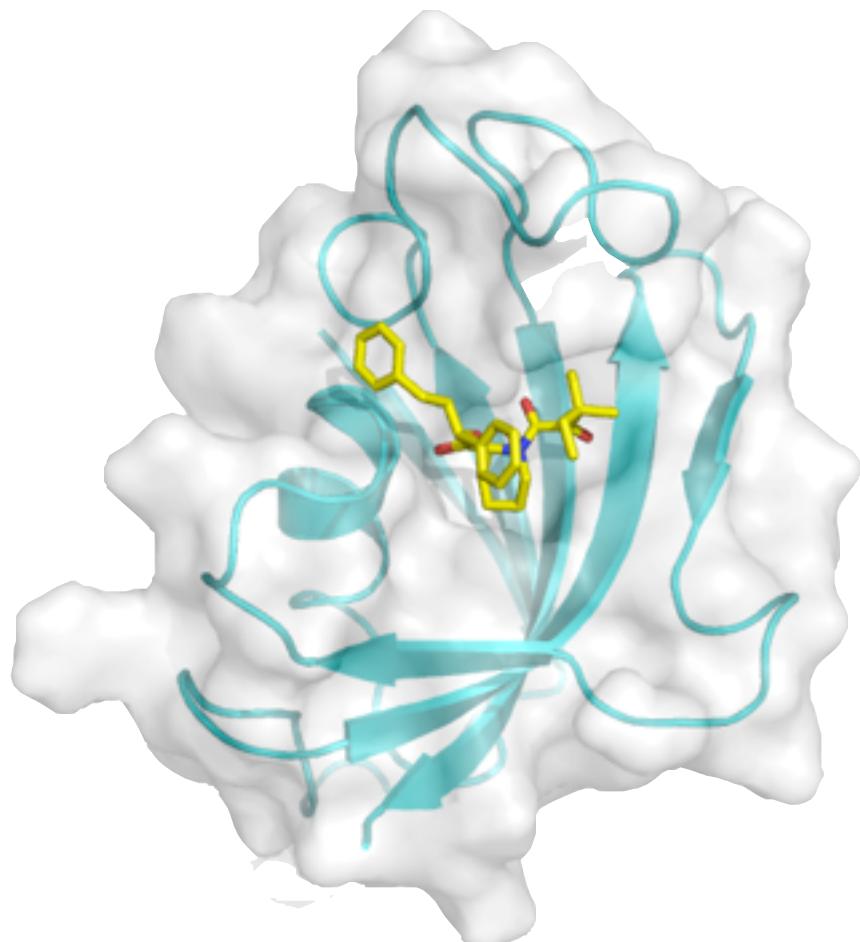


pharmaceutically relevant



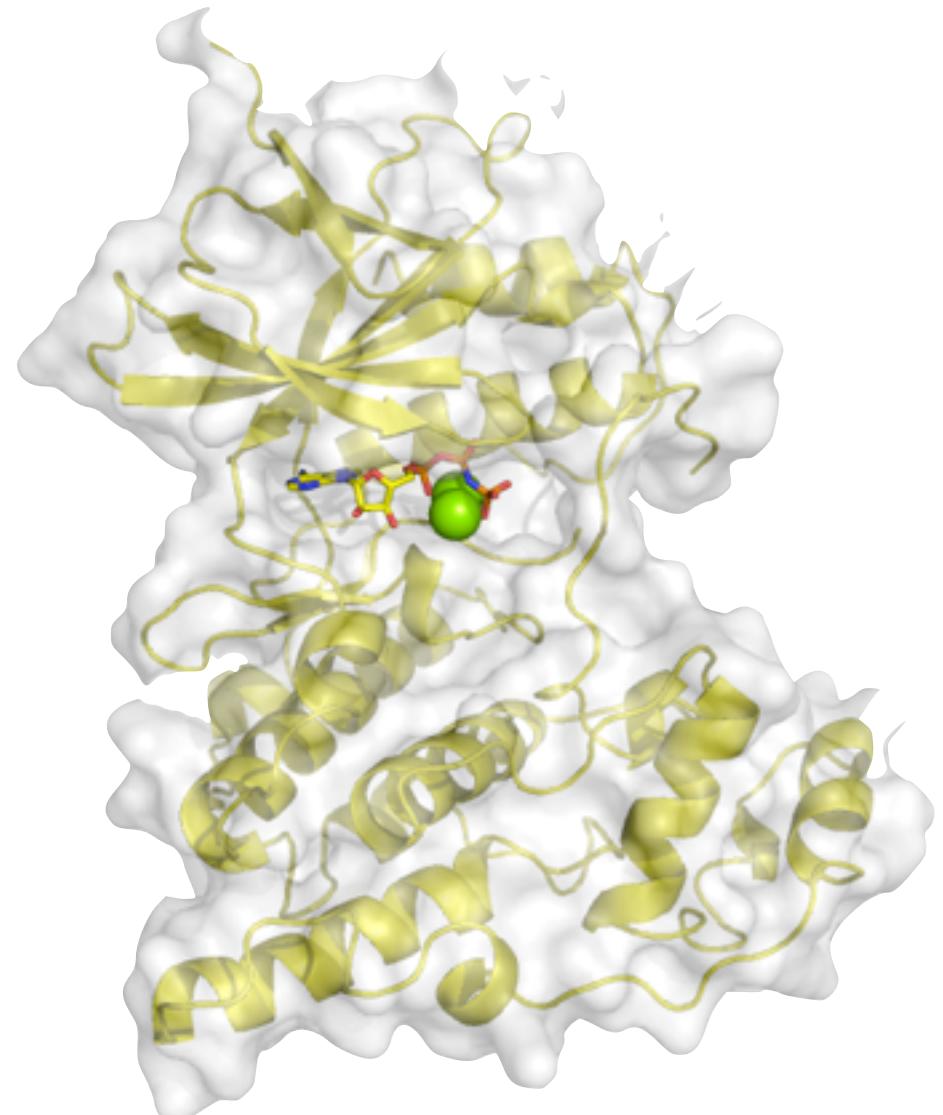
T4 lysozyme L99A

small, rigid protein
small, neutral ligands
fixed protonation states
multiple sidechain orientations
multiple ligand binding modes



FKBP12

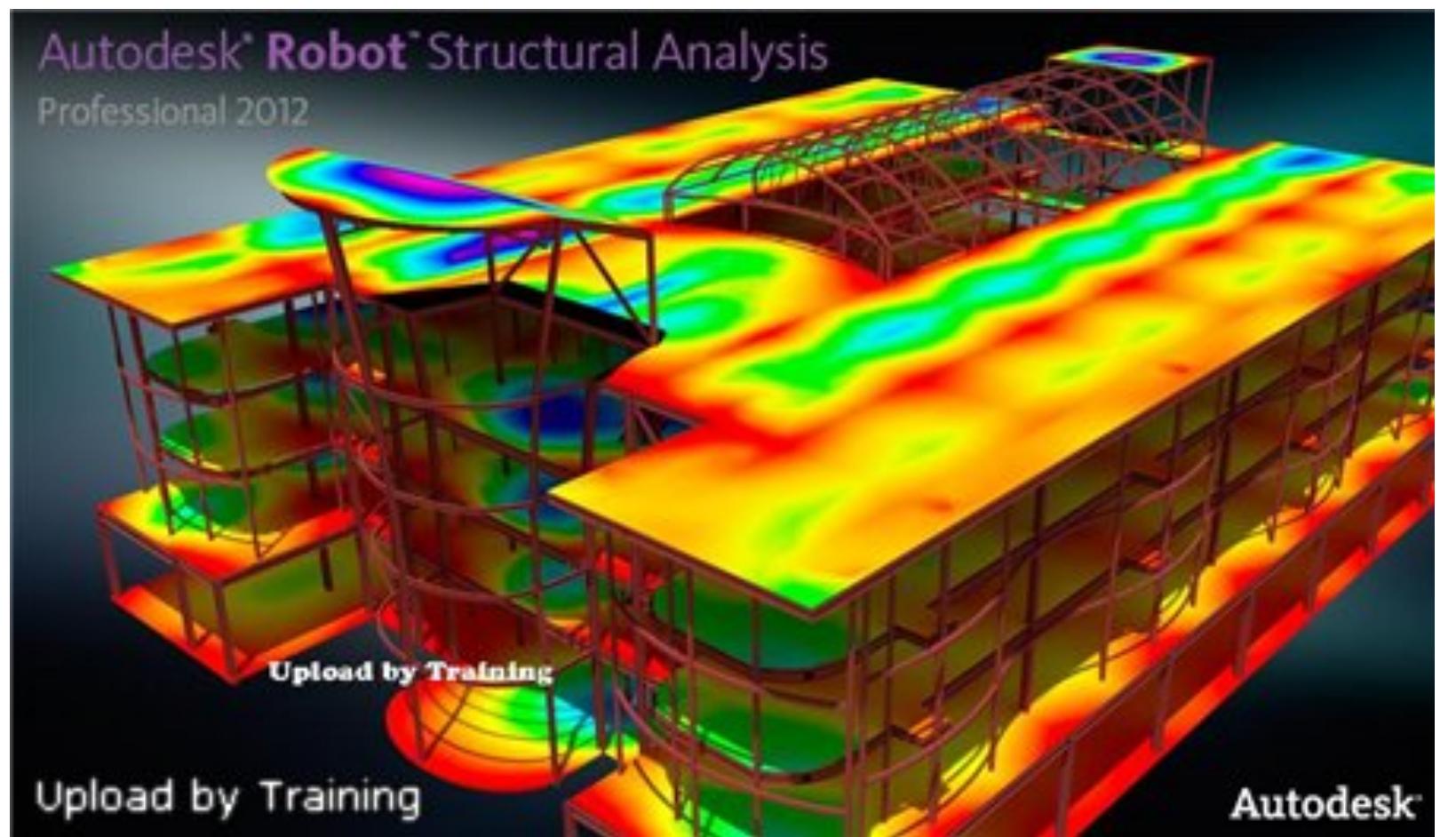
small, rigid protein
fixed protonation states
larger drug-like ligands
with few rotatable bonds



JNK3 kinase

large protein, multiple conformations
large drug-like ligands, rotatable bonds
multiple protonation states? tautomers?
phosphorylation and activation
peptide substrate?
 $MgCl_2$ salt effects?

(not to scale)



STRUCTURAL ENGINEERING WASN'T ALWAYS SO SUCCESSFUL

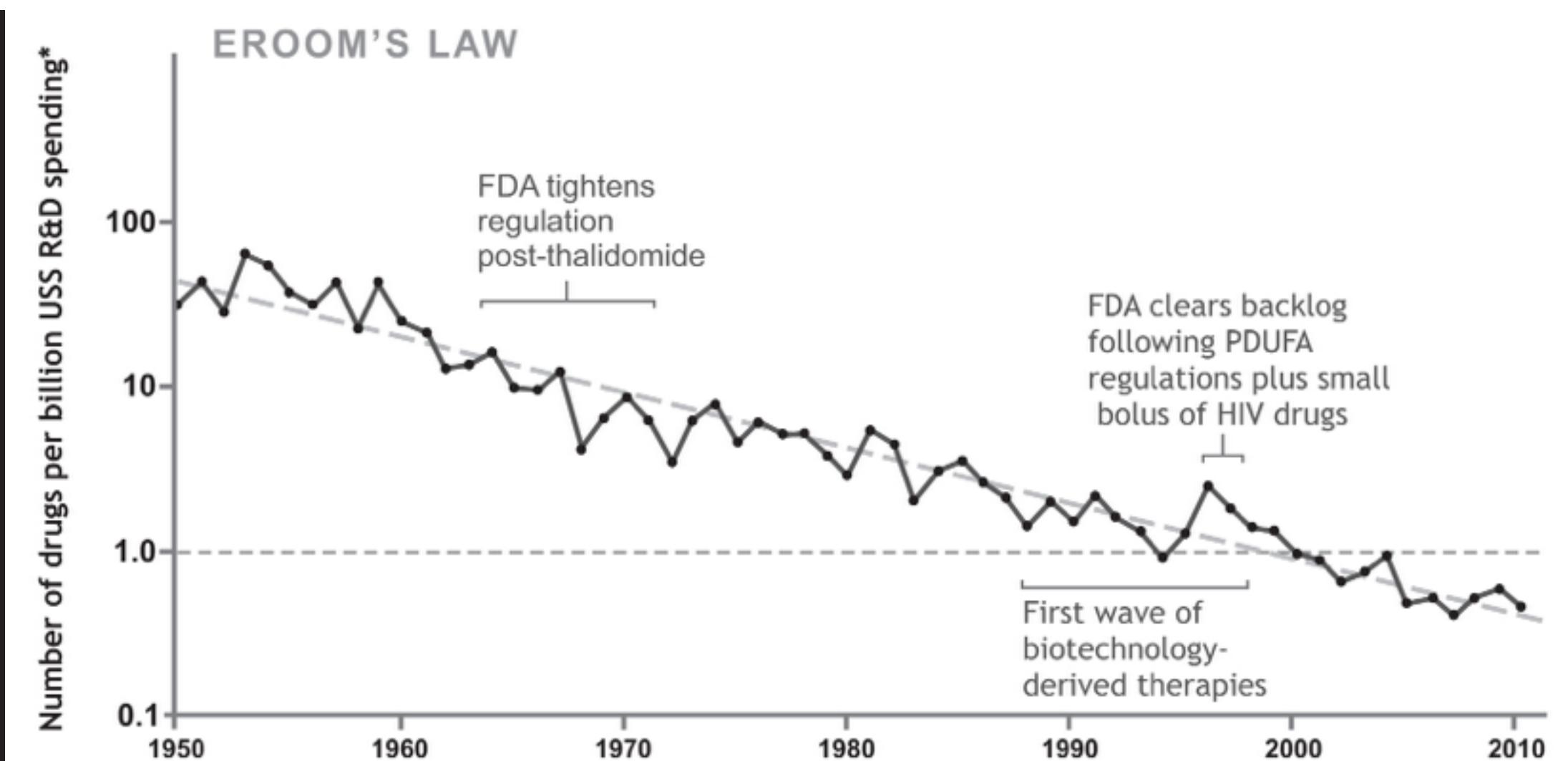
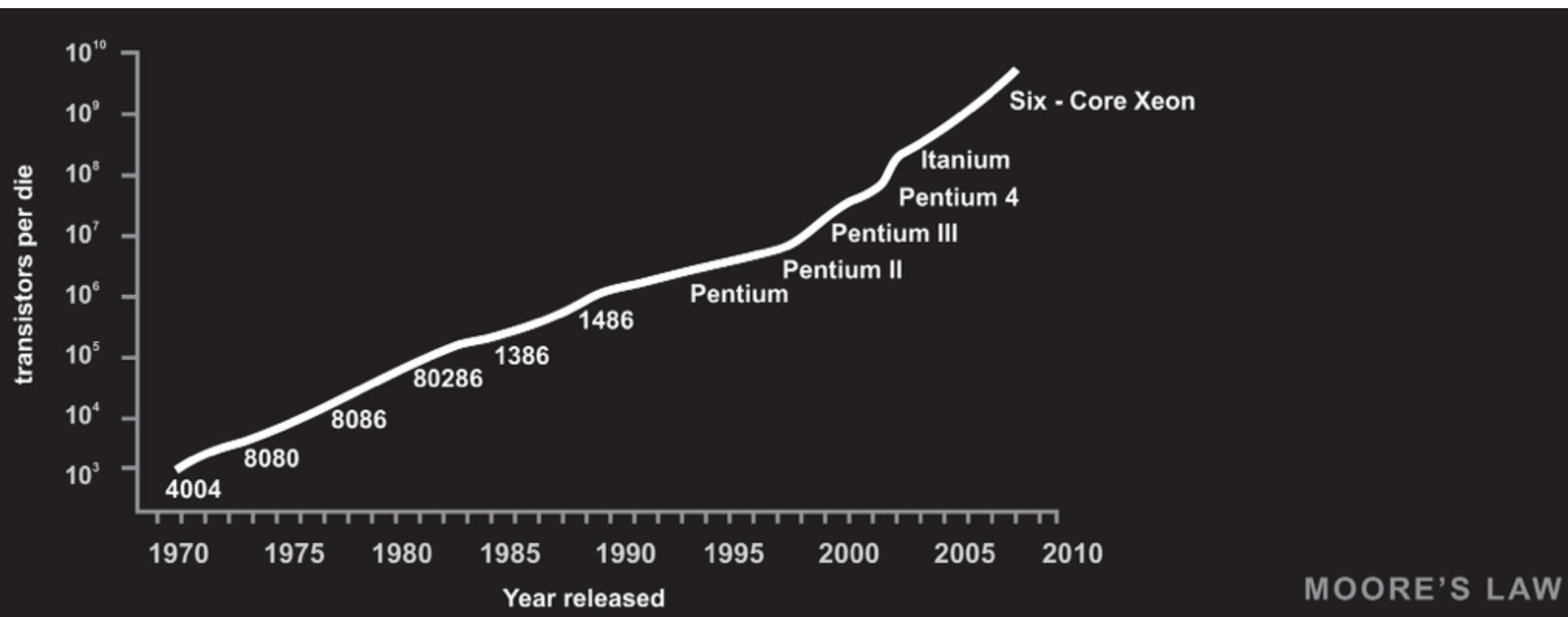


There were **250 bridge failures** in the US and Canada between 1878-1888.

*"The subject of **mechanical pathology** is relatively as legitimate and important a study to the engineer as medical pathology is to the physician. While we expect the physician to be familiar with physiology, without pathology he would be of little use to his fellow-men, and it [is] as much within the province of the engineer to investigate causes, study symptoms, and find remedies for mechanical failures as it is to direct the sources of power in nature for the use and convenience of man."*

- George Thomson, 1888

LEARNING FROM FAILURE DRIVES PROGRESS



"When I started Intel we couldn't make a device twice in a row in the same way. I earned my reputation by being part of a team that figured out why a thing was not reproducible... The attitude [in high-tech] is, something went wrong for a reason, let's find the gold nugget... But in pharma, if a clinical trial doesn't work...they just throw [the drug] away..."

- Andy Grove (former CEO of Intel)

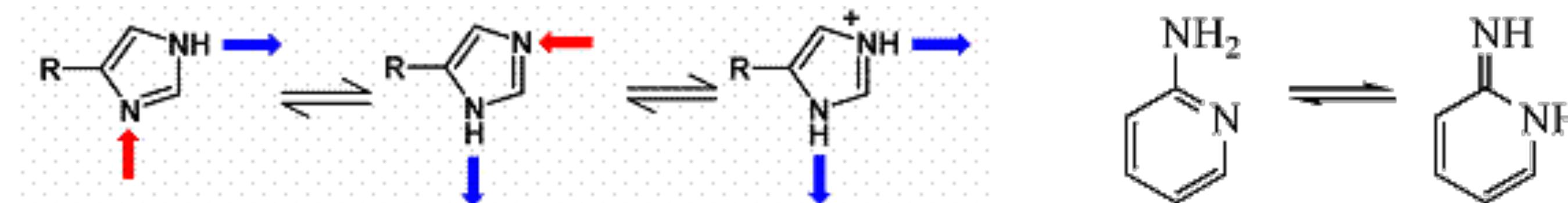


PREDICTIONS FAIL FOR THREE REASONS

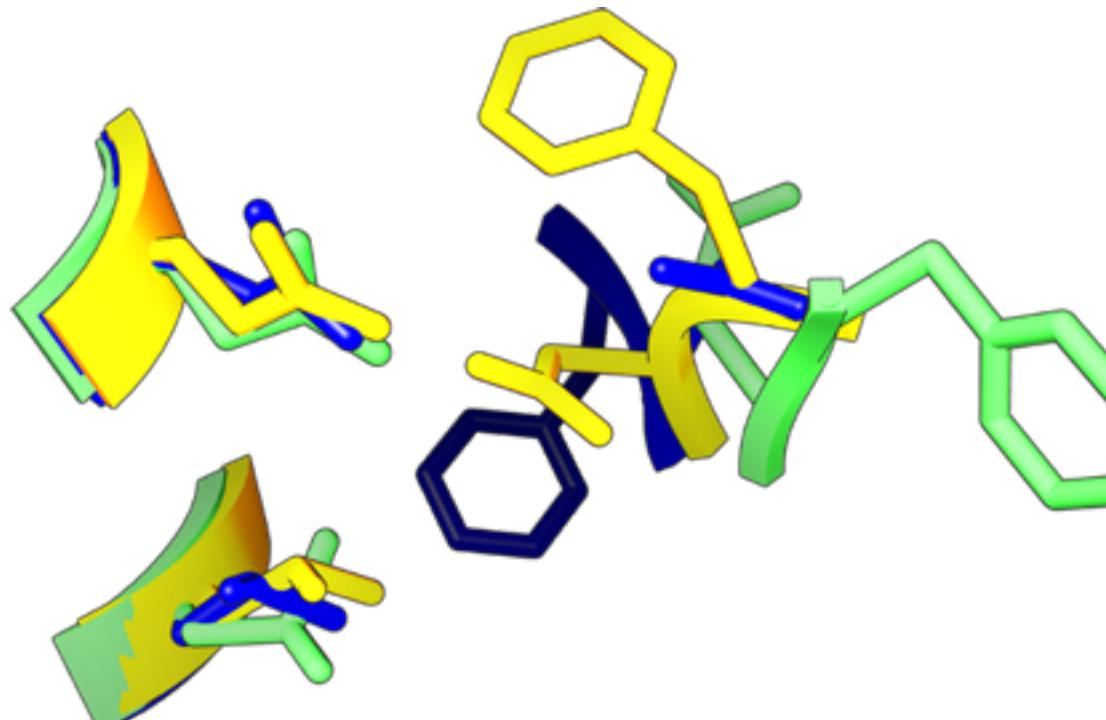
1. The **forcefield** does a poor job of modeling the physics of our system

$$V(\mathbf{q}) = \sum_{\text{bonds}} K_r(r - r_{eq})^2 + \sum_{\text{angles}} K_\theta(\theta - \theta_{eq})^2 + \sum_{\text{dihedrals}} \frac{V_n}{2}[1 + \cos(n\phi - \gamma)] + \sum_{i < j} \left[\frac{A_{ij}}{R_{ij}^{12}} - \frac{B_{ij}}{R_{ij}^6} + \frac{q_i q_j}{\epsilon R_{ij}} \right]$$

2. We're missing some **essential chemical** in our simulations
(e.g. protonation states, tautomers, covalent association)



3. We haven't **sampled** all of the relevant conformations



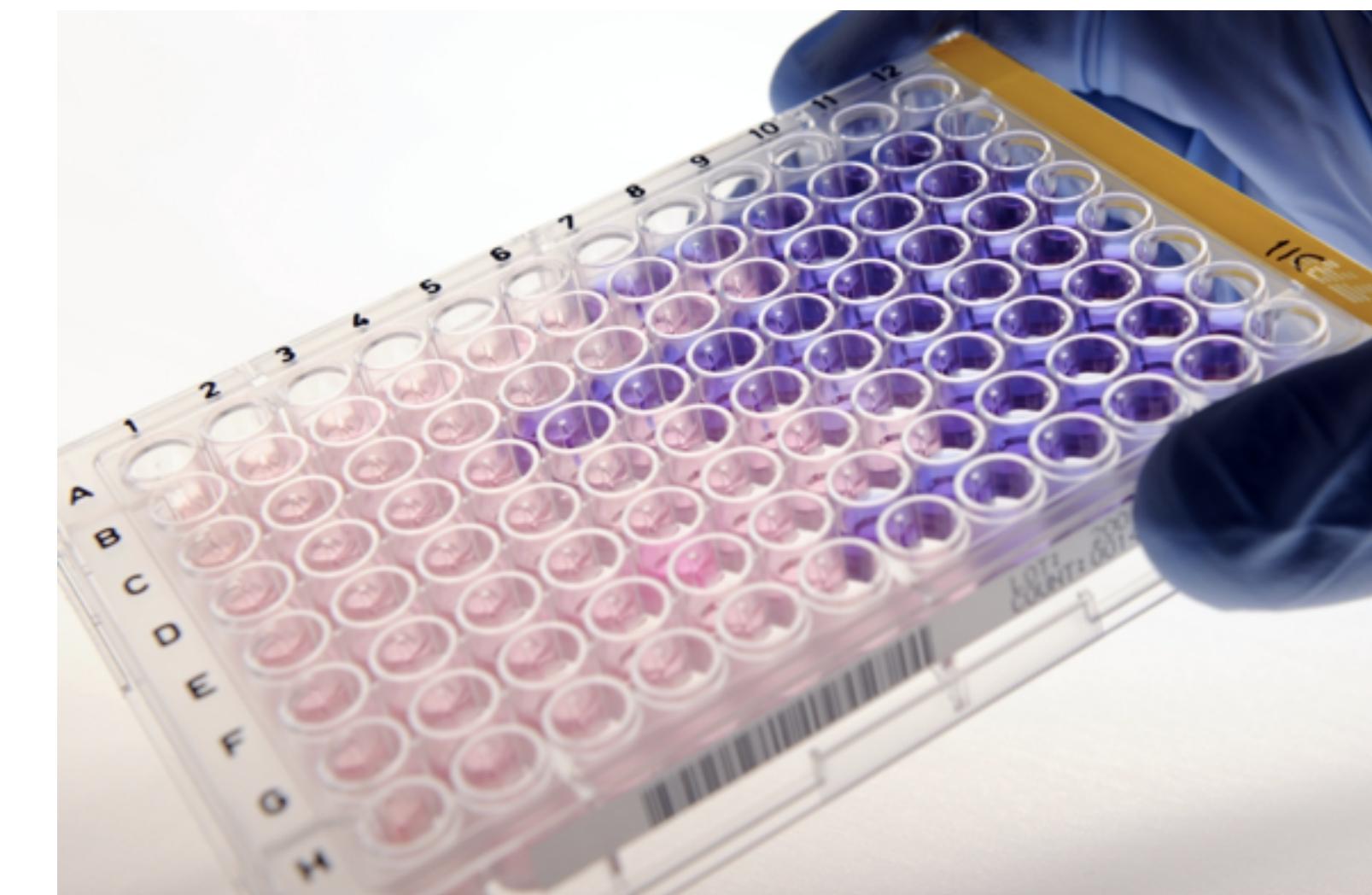
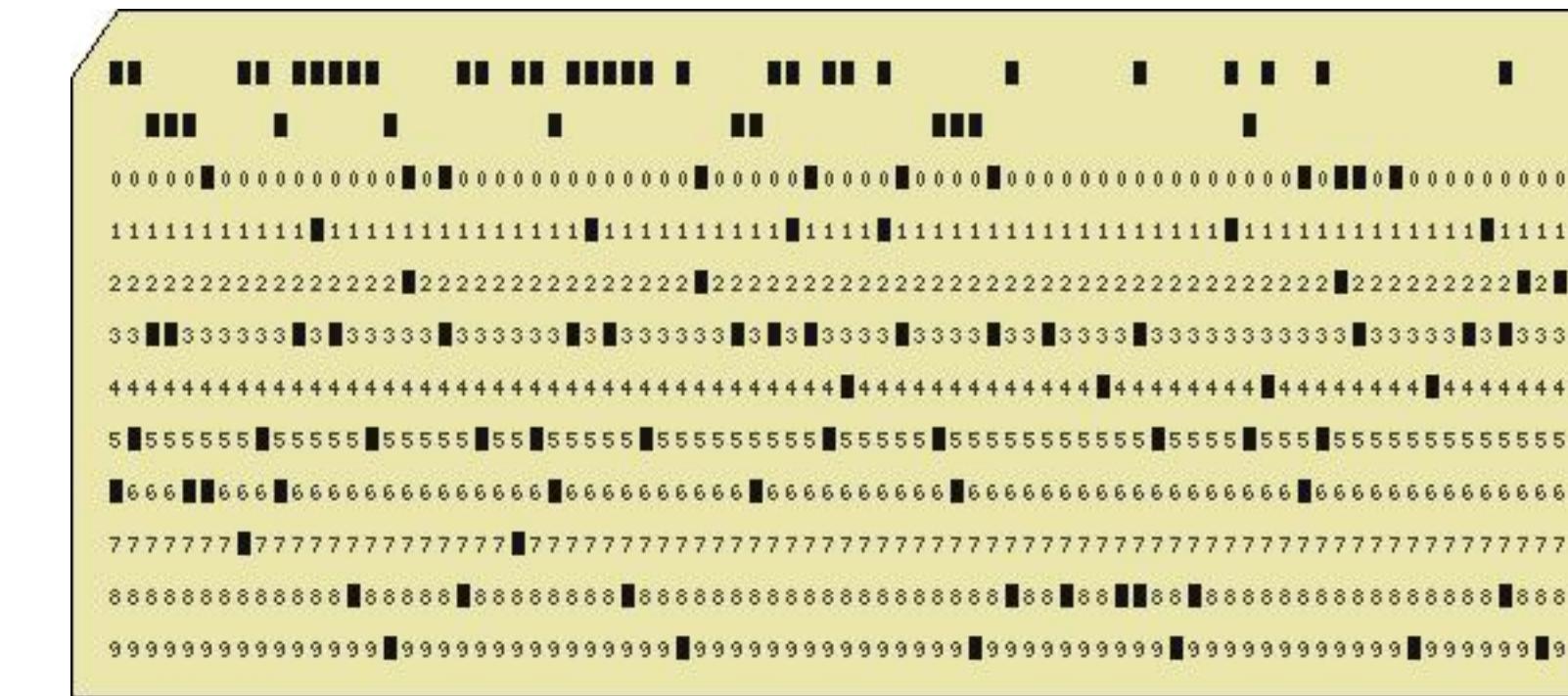
WE NEED TO UNDERSTAND WHY FAILURES OCCUR TO IMPROVE THE ROBUSTNESS OF OUR PREDICTIVE MODELS

FAIL FAST, FAIL CHEAP

computational
predictions



experimental
confirmation



WE NEED TO

- * **RUN SIMULATIONS QUICKLY ON FAST HARDWARE**
- * **EASILY BE ABLE TO TEST OUT NEW ALGORITHMS
OR TINKER WITH EXISTING ONES**
- * **EASILY MANAGE MANY SIMULATIONS AT ONCE**
- * **RUN FAULT-TOLERANT SIMULATIONS**
- * **SHUTTLE LOTS OF DATA AROUND WITHOUT MUCH HASSLE**

HOW CAN WE SPEED UP FREE ENERGY CALCULATIONS?

**\$50K
5 TFLOP/S**



**many CPU-weeks/
calculation**

**DOESN'T FIT NEATLY IN A SYNTHETIC CHEMIST'S
TIMEFRAME TO WAIT WEEKS FOR AN ANSWER.**

HOW CAN WE SPEED UP FREE ENERGY CALCULATIONS?

**\$50K
5 TFLOP/S**



**many CPU-weeks/
calculation**



**\$500
5 TFLOP/S**

**overnight on a
workstation?**

**WE CAN EXPLOIT NEW GPU TECHNOLOGIES TO REACH
PRACTICAL COMPUTATION TIMES**

YANK: AN OPEN-SOURCE, COMMUNITY-ORIENTED PLATFORM FOR GPU-ACCELERATED FREE ENERGY CALCULATIONS



NVIDIA GTX-TITAN-X (\$1000 FOR 7 TFLOP/S)

Simulation Type	CUDA (GTX Titan X)	OpenCL (GTX Titan X)	OpenCL (Radeon R9 Fury)	CPU (i7-2700K)	
Implicit, 2 fs	464	349	188	6.1	Modular Python framework for easily exploring new algorithms
Implicit, 5 fs HMR	670	580	230	14.9	GPU-accelerated via the OpenMM toolkit
Explicit-RF, 2 fs	266	244	127	14.6	Alchemical free energy calculations in both explicit and implicit solvent
Explicit-RF, 5 fs HMR	463	451	192	34.0	Hamiltonian exchange among alchemical intermediates with Gibbs sampling framework
Explicit-PME, 2 fs	174	161	71	10.5	General Markov chain Monte Carlo framework for exploring enhanced sampling methods
Explicit-PME, 5 fs HMR	353	340	131	25.1	Built-in equilibration detection and convergence diagnostics
					Support for AMBER prmtop/inpcrd files
					Support for absolute binding free energy calculations
					Support for transfer free energies (such as hydration free energies)

A free, open-source, extensible platform
for free energy calculations and ligand design

Docs » YANK

YANK

A GPU-accelerated Python framework for exploring algorithms for alchemical free energy calculations

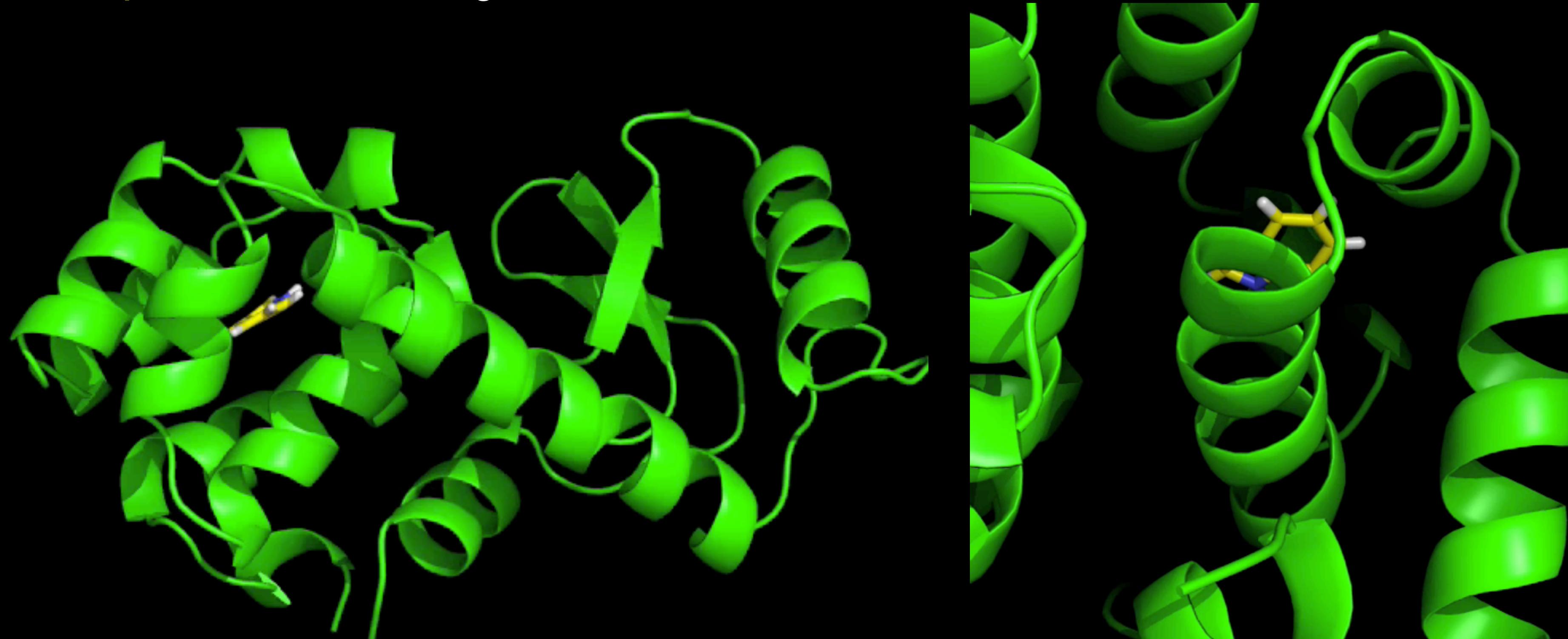
Features



<http://www.getyank.org>

HAMILTONIAN EXCHANGE PROTOCOL ALLOWS FOR REPEATED BINDING/UNBINDING EVENTS AND REORIENTATION IN SITE

solid fully interacting
transparent noninteracting

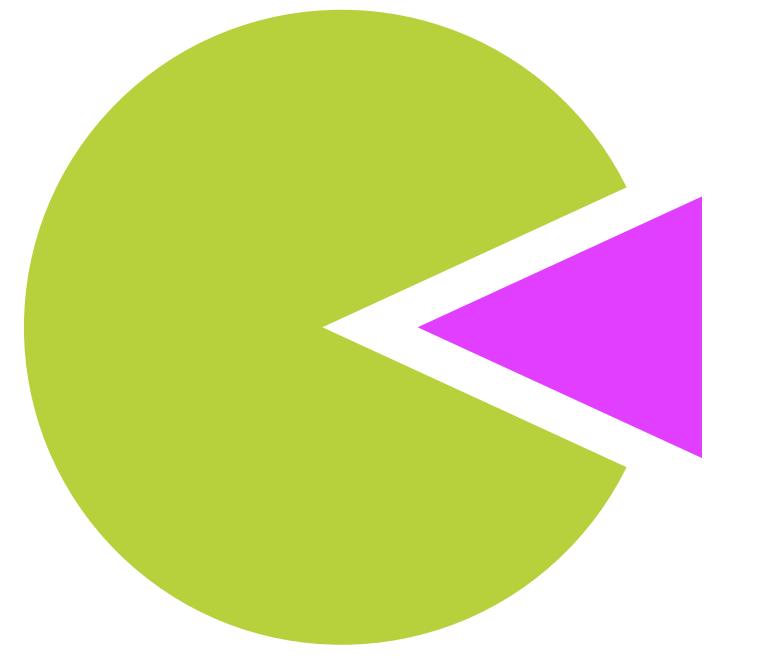


indole binding to T4 lysozyme L99A
12 h on 2 NVIDIA Tesla M2090 GPUs
Hamiltonian exchange with Gibbs sampling

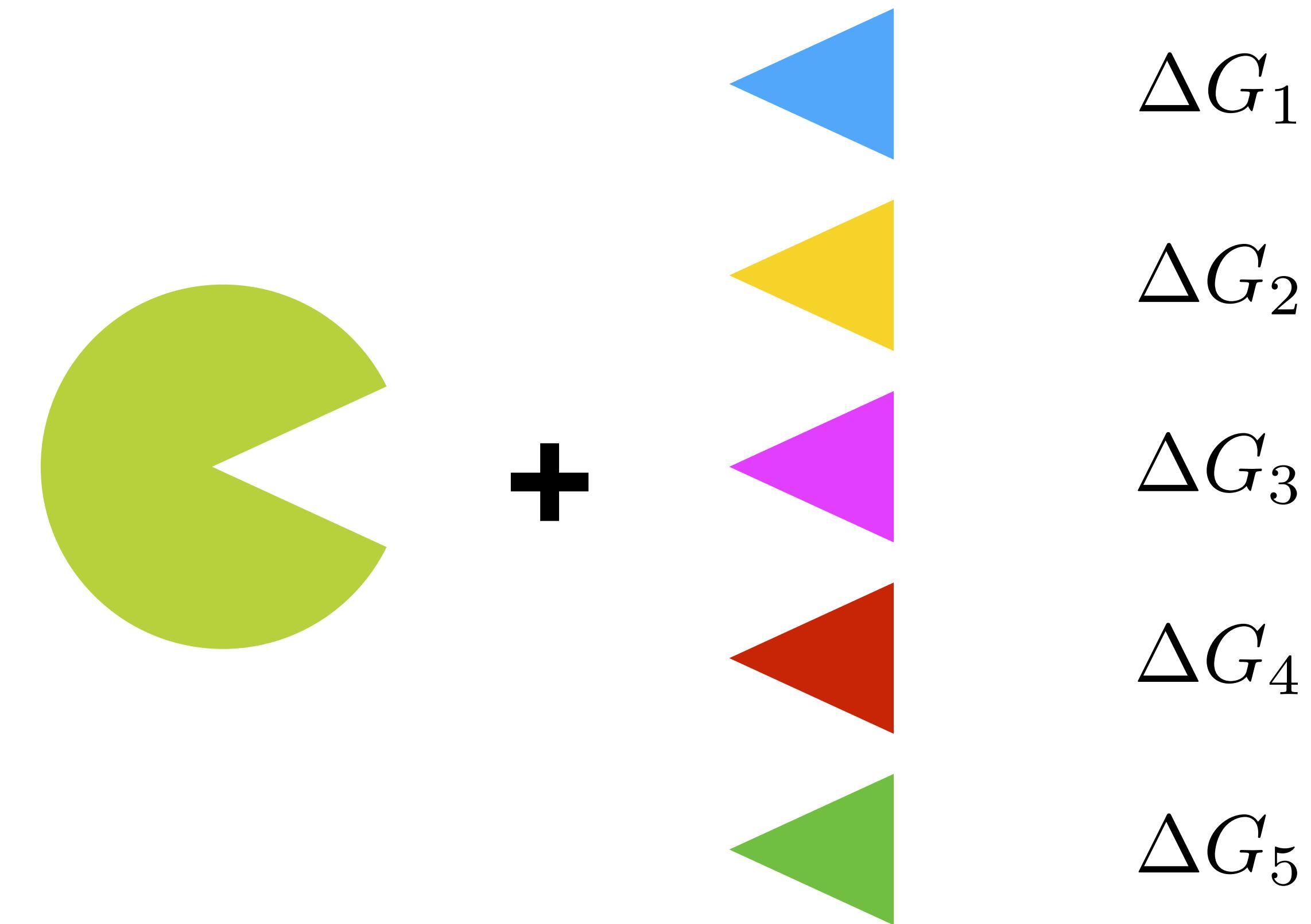
Chodera and Shirts. JCP 135:194110, 2011
Wang, Chodera, Yang, and Shirts. JCAMD 27:989, 2013.
<http://github.org/choderlab/yank>

WITH OPENMM WE CAN

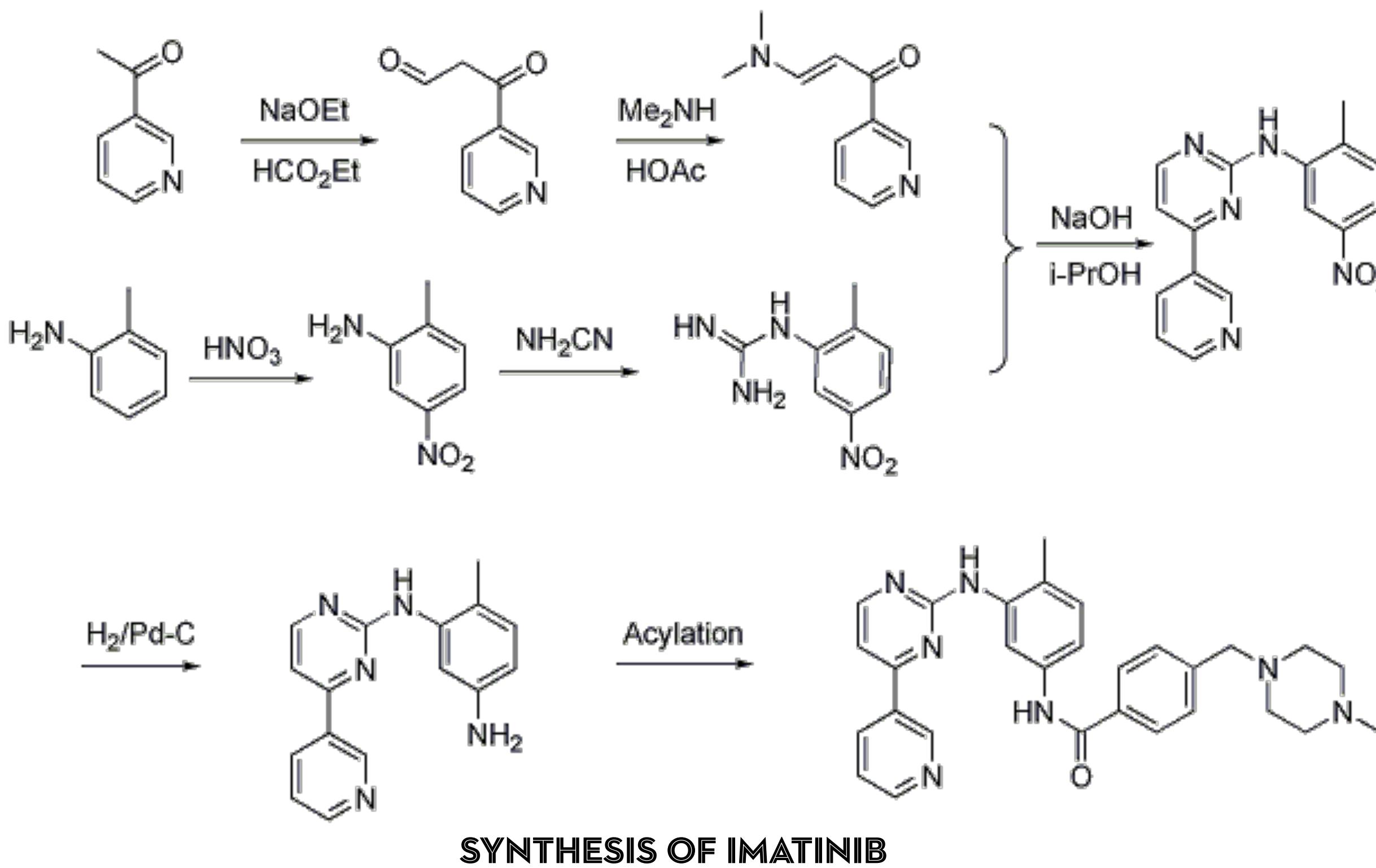
- * **RUN SIMULATIONS QUICKLY ON FAST HARDWARE USING INEXPENSIVE CONSUMER GPUS (<\$1K/GPU)**
- * **EASILY BE ABLE TO TEST OUT NEW ALGORITHMS BY BUILDING THEM IN PYTHON AGAINST A SIMPLE API**
 - * **CUSTOM FORCES (NO CODING, JUST MATH!)**
 - * **CUSTOM INTEGRATORS (NO CODING!)**
- * **EASILY MANAGE MANY SIMULATIONS AT ONCE USING PYTHON TOOLS**
- * **RUN FAULT-TOLERANT SIMULATIONS AND SHUTTLE DATA AROUND WITHOUT MUCH HASSLE USING NETCDF4 STORAGE**



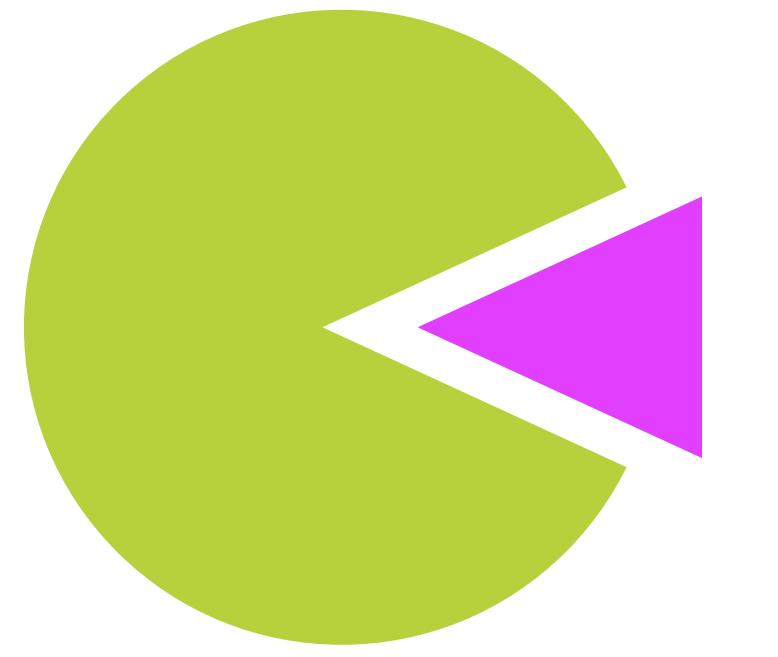
1. PREDICT HOW MODIFICATIONS TO **LIGAND** WILL CHANGE AFFINITY
2. TEST EXPERIMENTALLY



SYNTHESIS OF NEW COMPOUNDS TO TEST HYPOTHESES IS EXPENSIVE AND TIME-CONSUMING

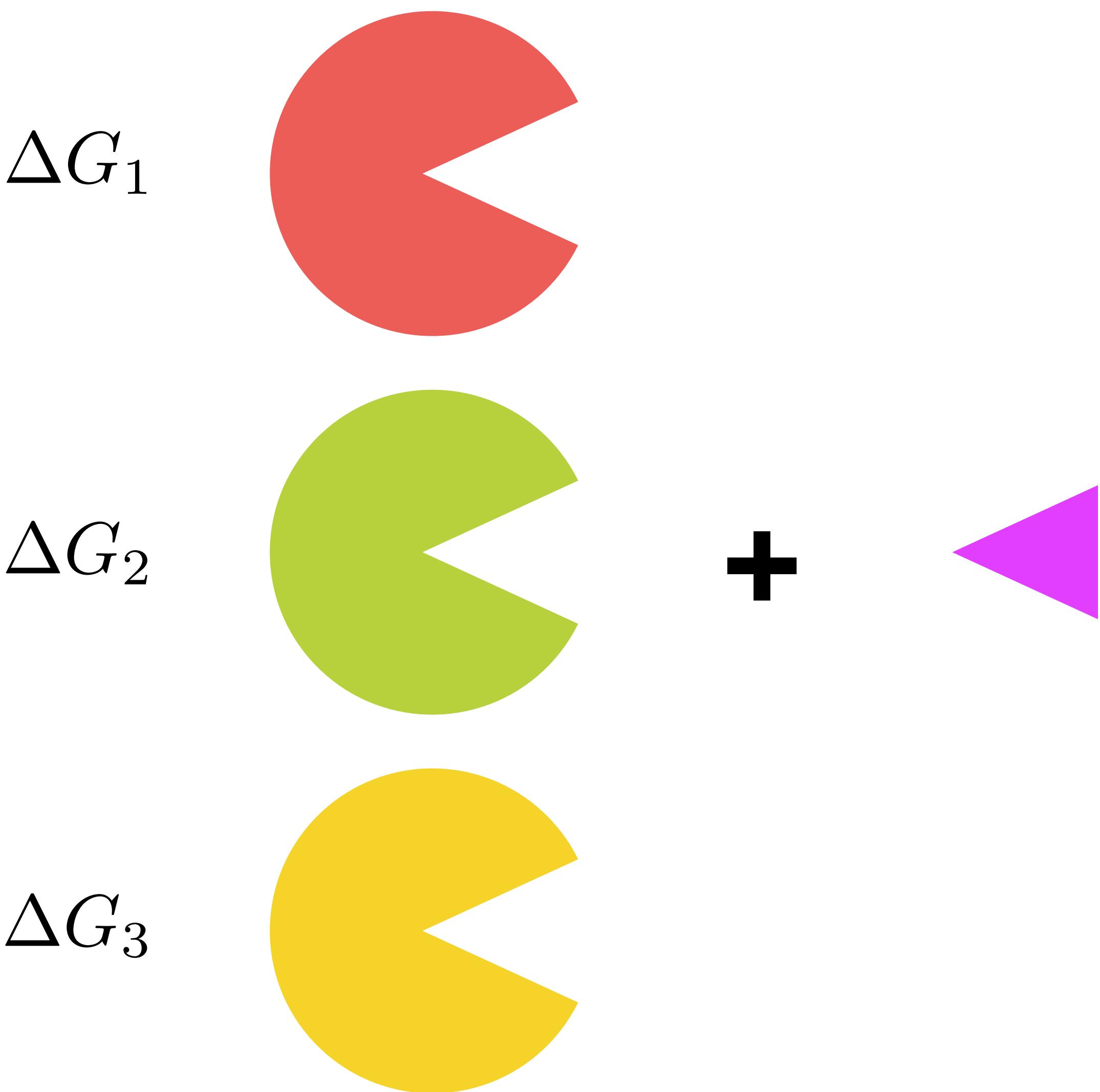


wiseGE

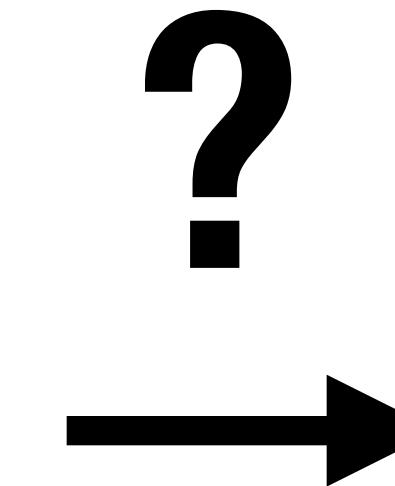


1. PREDICT HOW MODIFICATIONS TO PROTEIN WILL CHANGE AFFINITY

2. TEST EXPERIMENTALLY



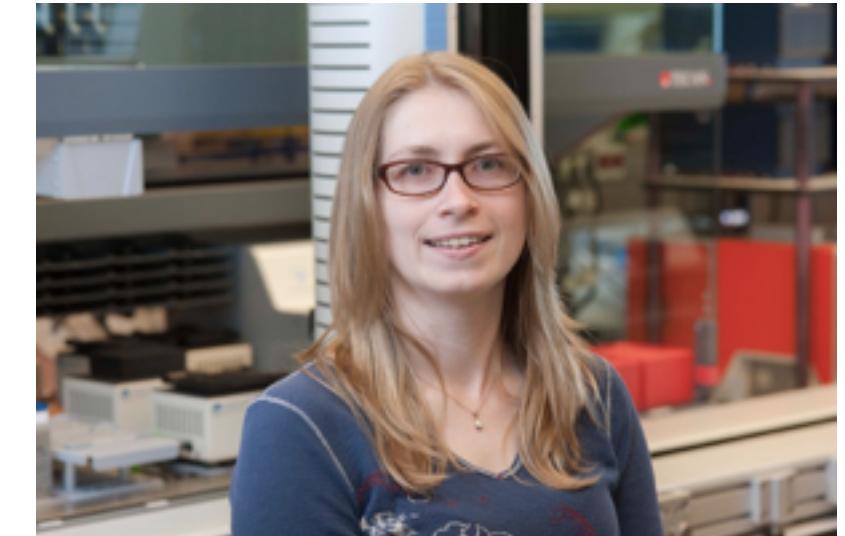
HOW CAN WE MAKE WETLAB EXPERIMENTS LOOK MORE LIKE PROBLEMS WE KNOW HOW TO SOLVE EFFICIENTLY?



messy
laborious
inconsistent
skill-dependent
9 am - 5 pm

precise
structured
consistent
reproducible
round-the-clock

ASSAY AUTOMATION CAN CONTROL ERROR



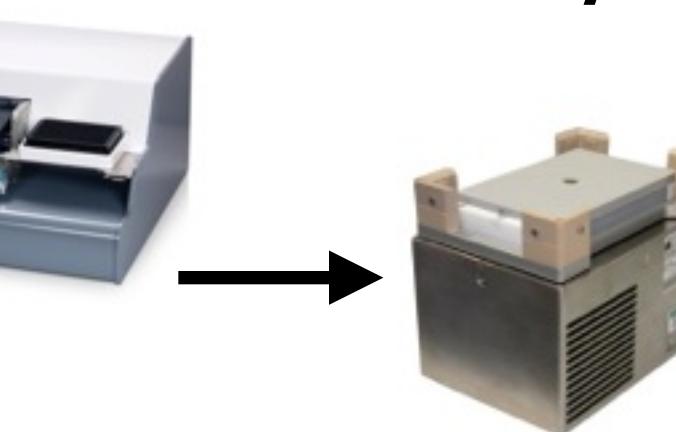
Sonya Hanson

Dispense

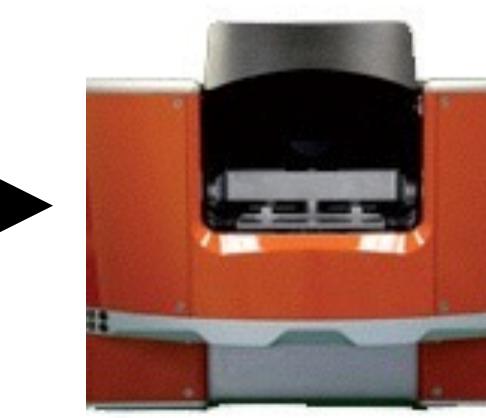
Bosutinib concentration: 20 μ M to 8 nM

	1	2	3	4	5	6	7	8	9	10	11	12
A	0	0	0	0	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0	0	0	0	0
C	0	0	0	0	0	0	0	0	0	0	0	0
D	0	0	0	0	0	0	0	0	0	0	0	0
E	0	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0	0	0	0	0	0
H	0	0	0	0	0	0	0	0	0	0	0	0

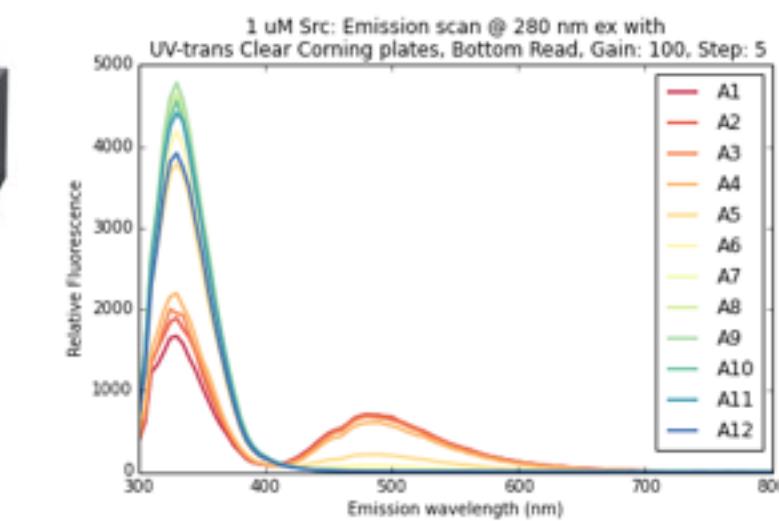
Shake/Mix



Spin



Read Fluorescence



TECAN.

MOVING TOWARD OPEN APIs FOR EXPERIMENTS



What is Autoprotocol?

Autoprotocol is a language for specifying experimental protocols for biological research in way that is precise, unambiguous, and understandable by both humans and computers. Autoprotocol was originally developed at [Transcriptic](#) as a way to define experiments that could be run over the internet on remote robotic automation, with the aim of moving research into the "cloud".

Since then, Autoprotocol has grown into an [open standard](#) with a rapidly growing [community of users](#), both big and small.

```
{  
  "refs": { ... },  
  "instructions": [  
    { "op": "pipette",  
      "groups": [  
        { "distribute": {  
          "from": "water/0",  
          "to": [  
            { "well": "test/A1",  
              "volume": "40:microliter" },  
            { "well": "test/A2",  
              "volume": "40:microliter" },  
            { "well": "test/A3",  
              "volume": "40:microliter" }  
          ]  
        }, { "distribute": {  
          "from": "dye/0",  
          "to": [  
            { "well": "test/A1",  
              "volume": "5:microliter" },  
            { "well": "test/A2",  
              "volume": "5:microliter" },  
            { "well": "test/A3",  
              "volume": "5:microliter" }  
          ]  
        }  
      ]  
    }, {  
      "op": "spin",  
      "object": "test",  
      "duration": "30:second",  
      "acceleration": "2000:g"  
    }, {  
      "op": "absorbance",  
      "object": "test",  
      "wells": ["A1", "A2", "A3"],  
      "wavelength": "600:nanometer"  
    }  
  ]  
}
```

Autoprotocol for Python:

<https://github.com/autoprotocol/autoprotocol-python>



Omnia
empowering molecular simulation

**OPEN SOURCE, HIGH PERFORMANCE, HIGH USABILITY
TOOLKITS FOR PREDICTIVE BIOMOLECULAR SIMULATION.**



OpenMM



<http://omnia.md>



**PREPARE X-RAY STRUCTURES
FOR SIMULATION**



**AUTOMATE MODELING OF
PROTEIN SEQUENCES
FOR SIMULATION**



**BUILD MARKOV STATE MODELS
TO ANALYZE STATISTICAL
BIOMOLECULAR DYNAMICS**



**RUN SIMULATIONS
BUILD MM CODES
TEST ALGORITHMS**



**RUN FREE ENERGY CALCULATIONS
EXPERIMENT WITH ALGORITHMS**



OpenMM



**READ/WRITE/ANALYZE
TRAJECTORIES**

AND MORE TO COME!

Vijay S. Pande, Stanford University



Vijay Pande is professor of Chemistry, Structural Biology, Biophysics, and Computer Science at Stanford University. Vijay is the founder and director of [Folding@Home](#), the world's largest distributed computing project.

[Pande lab webpage](#)

Peter Eastman, Stanford University



Peter Eastman is the lead architect and principal developer of the [OpenMM](#) molecular dynamics suite, as well as the lead developer of [PDBFixer](#).

Frank Noé, FU Berlin



Frank Noé is professor in the department for mathematics and computer science, with affiliations to physics and chemistry at Freie Universität Berlin (Germany), and one of the lead designers of the [PyEMMA](#) package for Estimation, Validation and Analysis of Markov models.

[Noé lab webpage](#)

John D. Chodera, MSKCC



John Chodera is an assistant professor in the Computational Biology Program at the Memorial Sloan-Kettering Cancer Center, and the lead developer of the [YANK](#) package for alchemical binding free energy calculations.

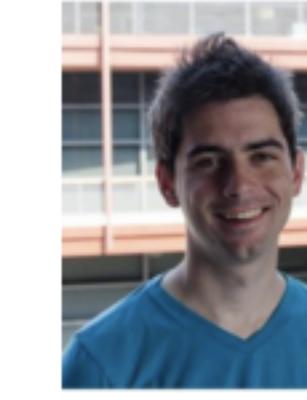
[Chodera lab webpage](#)

Jason M. Swails, Rutgers University



Jason Swails is a postdoctoral researcher in the Case lab at Rutgers University. He is the principal developer of the [ParmEd](#) program to rapidly prototype force field modifications and development. He is also a contributor to the [OpenMM](#) and [MDTraj](#) projects.

Robert T. McGibbon, Stanford University



Robert McGibbon is a graduate student in the Pande lab at Stanford. Robert is the lead developer [MDTraj](#), a co-principle developer of [MSMBuilder](#), and a contributor to [OpenMM](#).

Justin L. MacCallum, University of Calgary



Justin MacCallum is an assistant professor in the Department of Chemistry at the University of Calgary. He is the lead developer of the [MELD](#) package for inferring protein structure from sparse and unreliable data.

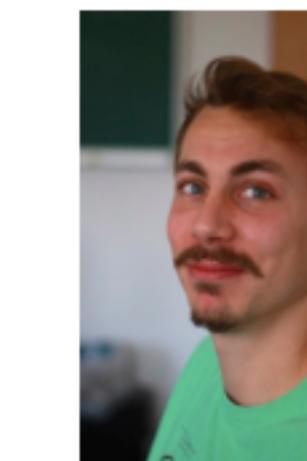
[MacCallum lab webpage](#)

Jan-Hendrik Prinz, FU Berlin



Jan-Hendrik is currently a postdoctoral researcher in the Noé group at Freie Universität Berlin (Germany), and has also worked in John Chodera's and Peter Bolhuis' groups. He is the co-principal developer of the [OpenPathSampling](#) package for path-based algorithms to efficiently sample rare events in molecular dynamics.

Martin K. Scherer, FU Berlin



Martin is a PhD student in the Noé group at Freie Universität Berlin (Germany). Martin is the co-principle developer of [PyEMMA](#) and a contributor to [msmtools](#).

Benjamin Trendelkamp-Schroer, FU Berlin



Benjamin is a PhD student in the Noé group at Freie Universität Berlin (Germany). He is a co-principal developer of [PyEMMA](#) and the lead developer of [msmtools](#).

GOALS OF OMNIA

- * **BUILD A ROBUST COMMUNITY OF INTEROPERABLE TOOLS
WITHOUT DRACONIAN LEADERSHIP**
- * **MINIMIZE NEEDS FOR IMPOSING HEAVY-HANDED STANDARDS SPECIFICATIONS**
- * **ALLOW RAPID DEVELOPMENT OF TOOLS THAT USE OTHER TOOLS**
- * **ALLOW STANDARDS TO EVOLVE NATURALLY**

STILL NEED TO AGREE ON SOME BASICS

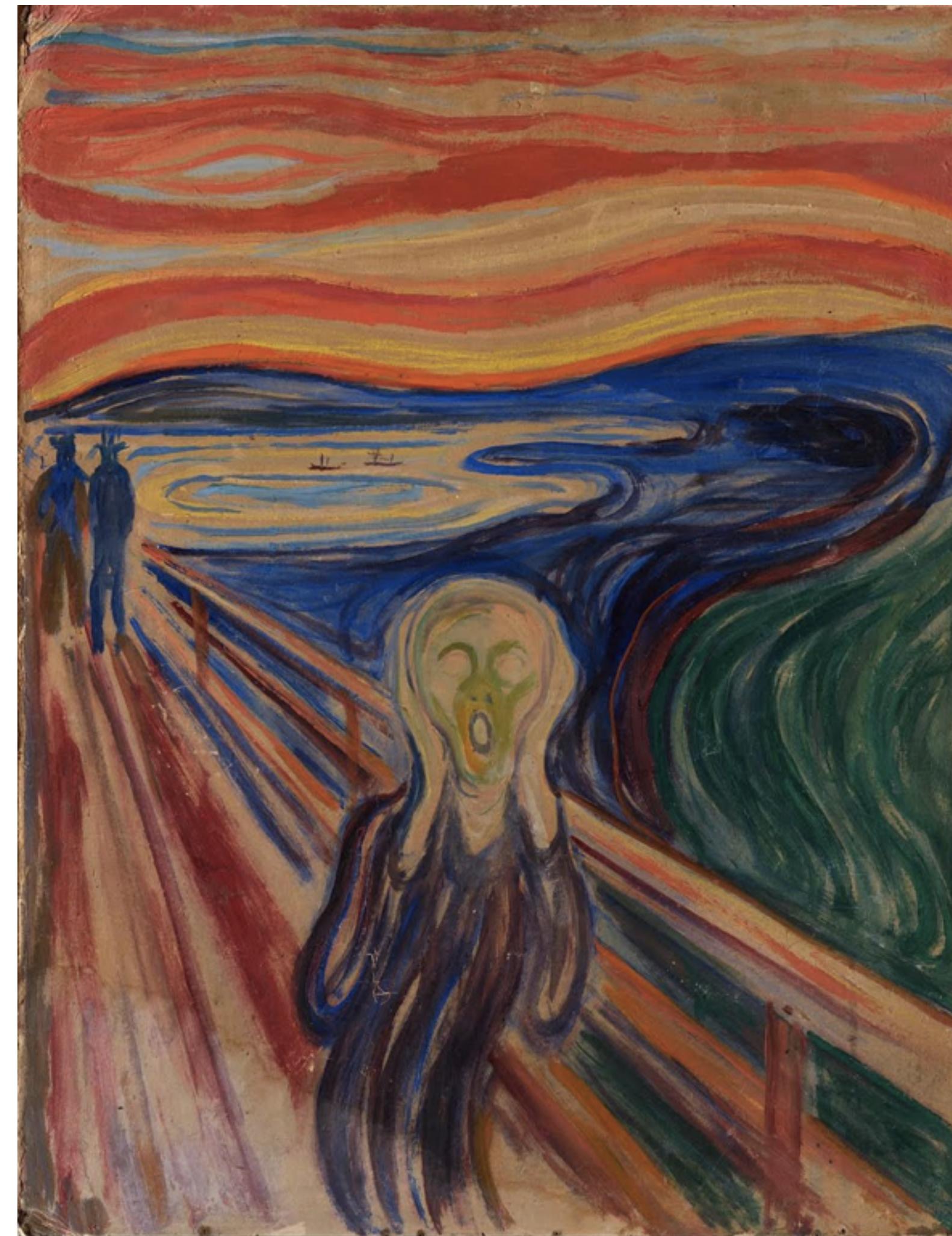
- * **SIMULATION PACKAGES HAVE TOO MANY SCRIPTING LANGUAGES:**
AGREE ON PYTHON AS A LINGUA FRANCA
- * **DEPENDENCIES WERE HARD TO INSTALL:**
AGREE ON A COMMON PACKAGING AND DISTRIBUTION METHOD
- * **DEVELOPERS ARE HARD TO COORDINATE:**
AGREE ON A COLLABORATIVE DEVELOPMENT ENVIRONMENT
- * **CODE IS HARD TO TEST AND VERIFY:**
AGREE ON A TESTING PHILOSOPHY
- * **TOO MANY FORMATS, STANDARDS, AND ENGINES:**
MAKE HEAVY USE OF SOME COMMON CORE COMPONENTS
- * **RELEASES ARE MANUALLY TIME-CONSUMING:**
USE A SINGLE AUTOMATED FRAMEWORK FOR RELEASES

PYTHON

- * **FREE**
- * **EASY TO READ**
- * **EASY TO LEARN**
- * **BATTERIES INCLUDED**
- * **GOOD NUMERIC LIBRARIES**
- * **LOTS OF SCIENTIFIC CODES NOW WRAPPED IN PYTHON**
- * **ADOPTED BY MANY ORGANIZATIONS**
- * **CAN EASILY TALK TO THE INTERNETS**

INSTALLING DEPENDENCIES

HAS ANYONE HERE EVER TRIED TO COMPILE SCIPY?



INSTALLING OMNIA

```
conda install -c omnia omnia
```

That's really it. Seriously.



ANACONDA®

LIKE PYPI, BUT WAY BETTER

Anaconda Cloud  About Pricing Help Login

 Omnia Molecular Dynamics
omnia

<http://www.omnia.md/>
Joined on Mar 12, 2014
88 Packages

Packages 

Quick Nav [all](#) [personal](#) [private](#) [original](#) [copy](#)

	Packages	conda
	ensembl-dev dev Pipeline for automating omics-scale protein modeling and simulation setup.	conda
	bhmm 0.5.1 Bayesian hidden Markov models for analysis of single-molecule trajectory data	conda
	mdtraj-dev 1.5.0.dev0 A modern, open library for the analysis of molecular dynamics trajectories (development snapshot).	conda
	openbabel 2015.09 A chemical toolbox designed to speak the many languages of chemical data	conda
	pymol 1.7.4.0 No Summary	conda
	openmm-dev 7.0.0.dev0 A toolkit for molecular simulation using high performance GPU code (development snapshot).	conda

LIKE PYPI, BUT WAY BETTER

separate channels

Anaconda Cloud Search Packages ... About Pricing Help Login

omnia / openmm
A toolkit for molecular simulation using high performance GPU code.

Description Files Channels Metrics Badges Builds

older versions for reproducibility

binary packages for each architecture and Python version

download statistics!

<input type="checkbox"/> type	size	name	uploaded	uploaded/built by	downloads	channels
<input type="checkbox"/> conda	5.7 MB	i win-64/openmm-6.3-py33_0.tar.bz2	2 months and 18 days ago	omnia	6	main
<input type="checkbox"/> conda	5.7 MB	i win-64/openmm-6.3-py34_0.tar.bz2	2 months and 18 days ago	omnia	62	main
<input type="checkbox"/> conda	9.9 MB	i linux-64/openmm-6.3-py34_0.tar.bz2	2 months and 21 days ago	omnia	396	main
<input type="checkbox"/> conda	9.9 MB	i linux-64/openmm-6.3-py33_0.tar.bz2	2 months and 21 days ago	omnia	140	main
<input type="checkbox"/> conda	10.0 MB	i linux-64/openmm-6.3-py27_0.tar.bz2	2 months and 21 days ago	omnia	1199	main
<input type="checkbox"/> conda	7.4 MB	i osx-64/openmm-6.3-py34_0.tar.bz2	2 months and 21 days ago	omnia	77	main
<input type="checkbox"/> conda	7.5 MB	i osx-64/openmm-6.3-py33_0.tar.bz2	2 months and 21 days ago	omnia	22	main
<input type="checkbox"/> conda	7.4 MB	i osx-64/openmm-6.3-py27_0.tar.bz2	2 months and 21 days ago	omnia	131	main
<input type="checkbox"/> conda	3.4 MB	i win-32/openmm-6.2-py34_3.tar.bz2	5 months and 5 days ago	omnia	117	main
<input type="checkbox"/> conda	7.0 MB	i osx-64/openmm-6.2-py34_3.tar.bz2	5 months and 25 days ago	omnia	26	main
<input type="checkbox"/> conda	7.0 MB	i osx-64/openmm-6.2-py33_3.tar.bz2	5 months and 25 days ago	omnia	18	main

You may want to add the **omnia** channel
and install packages individually:

```
conda config --add channels http://conda.anaconda.org/omnia  
conda install openmm
```

Dependencies are automatically resolved!

Install specific versions:

```
conda install openmm==6.2
```

Install latest development snapshots:

```
conda install openmm-dev
```

CONDA PACKAGE MANAGER

- * **ANACONDA PYTHON: FULL SCIENTIFIC PYTHON SUITE
GREAT FOR SCIENTISTS!**
- * **MINICONDA: MINIMAL “BOOTSTRAP” VERSION
GREAT FOR AUTOMATING INSTALLS VIA THE WEB**
- * **ANACONDA SERVER: CENTRALLY-MANAGED PACKAGES**



ANACONDA®

<http://continuum.io>
<http://conda.pydata.com>

ANACONDA

Anaconda for OS X

PYTHON 2.7	PYTHON 3.4
Mac OS X 64-bit Graphical Installer <small>283M (OSX 10.7 or higher)</small>	Mac OS X 64-bit Graphical Installer <small>292M (OSX 10.7 or higher)</small>
Mac OS X 64-bit Command-Line installer <small>249M (OSX 10.7 or higher)</small>	Mac OS X 64-bit Command-Line installer <small>257M (OSX 10.7 or higher)</small>

Anaconda for Linux

PYTHON 2.7	PYTHON 3.4
Linux 64-bit <small>323M</small>	Linux 64-bit <small>336M</small>
Linux 32-bit <small>309M</small>	Linux 32-bit <small>322M</small>

Anaconda for Windows

PYTHON 2.7	PYTHON 3.4
Windows 64-bit Graphical Installer <small>334M</small>	Windows 64-bit Graphical Installer <small>336M</small>
Windows 32-bit Graphical Installer <small>281M</small>	Windows 32-bit Graphical Installer <small>277M</small>

Behind a firewall? Use these zipped Windows installers.



ANACONDA®

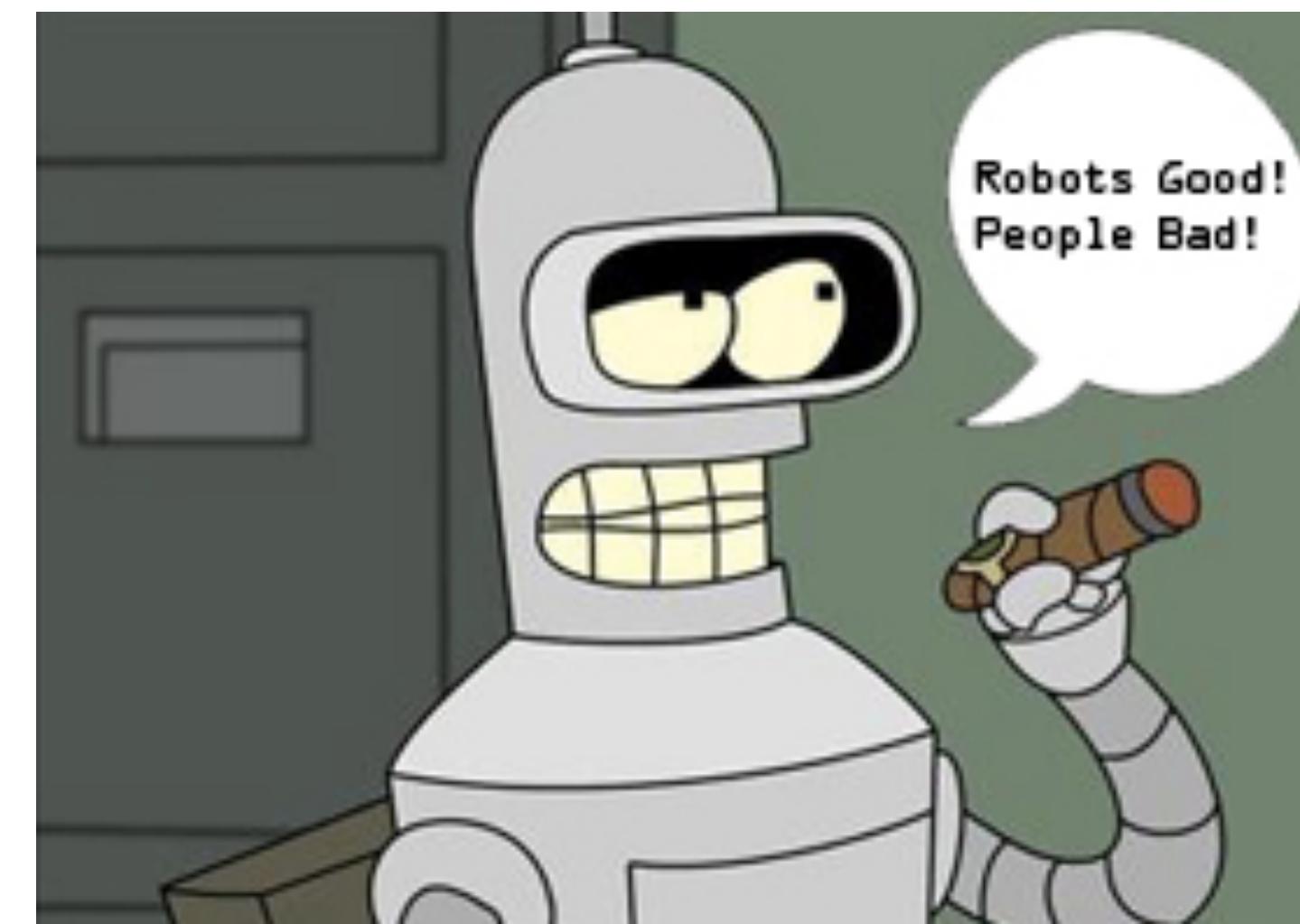
- * **OVER 100 PACKAGES INCLUDED IN INSTALL**
- * **OVER 200 PACKAGES ONLINE**

<http://continuum.io>
<http://conda.pydata.com>

MINICONDA

```
wget https://repo.continuum.io/miniconda/Miniconda-latest-Linux-x86_64.sh  
bash -b ./Miniconda-latest-Linux-x86_64.sh -p $HOME/miniconda  
export PATH=$HOME/miniconda/bin:$PATH
```

GREAT FOR BOTS!



CONDA RECIPES

`meta.yaml:`

```
package:
  name: openmm
  version: 6.3

source:
  url: https://github.com/pandegroup/openmm/archive/6.3.tar.gz
  fn: 6.3.tar.gz

build:
  number: 0
  preserve_egg_dir: yes

requirements:
  build:
    # on windows, need to install cmake manually
    - cmake      [not win]
    - python
    - fftw3f
    - sphinx
    - sphinxcontrib-bibtex
    - swig
    - doxygen   [win]
    - jom        [win]
  run:
    - python
    - fftw3f

test:
  requires:
    - python
  imports:
    - simtk
    - simtk.openmm
  commands:
    - python -m simtk.testInstallation

about:
  home: http://openmm.org
  license: LGPL and MIT
  summary: A high performance toolkit for molecular simulation.
```

package and version info

where to get the source

dependencies

test the package

package metadata

`build.sh:`

```
$PYTHON setup.py install
```

CONDA RECIPES

build.sh:

```
#!/bin/bash

CMAKE_FLAGS="-DCMAKE_INSTALL_PREFIX=$PREFIX -DBUILD_TESTING=OFF"

if [[ "$OSTYPE" == "linux-gnu" ]]; then
    # setting the rpath so that libOpenMMPME.so finds the right libfftw3
    CMAKE_FLAGS+=" -DCMAKE_C_COMPILER=clang -DCMAKE_CXX_COMPILER=clang++"
    CMAKE_FLAGS+=" -DCUDA_CUDART_LIBRARY=/usr/local/cuda-7.0/lib64/libcudart.so"
    CMAKE_FLAGS+=" -DCUDA_NVCC_EXECUTABLE=/usr/local/cuda-7.0/bin/nvcc"
    CMAKE_FLAGS+=" -DCUDA_SDK_ROOT_DIR=/usr/local/cuda-7.0/"
    CMAKE_FLAGS+=" -DCUDA_TOOLKIT_INCLUDE=/usr/local/cuda-7.0/include"
    CMAKE_FLAGS+=" -DCUDA_TOOLKIT_ROOT_DIR=/usr/local/cuda-7.0/"
    CMAKE_FLAGS+=" -DOPENCL_INCLUDE_DIR=/opt/AMDAPPSDK-2.9-1/include/"
    CMAKE_FLAGS+=" -DOPENCL_LIBRARY=/opt/AMDAPPSDK-2.9-1/lib/x86_64/libOpenCL.so"
elif [[ "$OSTYPE" == "darwin"* ]]; then
    CMAKE_FLAGS+=" -DCMAKE_C_COMPILER=clang -DCMAKE_CXX_COMPILER=clang++"
    CMAKE_FLAGS+=" -DCMAKE_OSX_DEPLOYMENT_TARGET=10.9"
    CMAKE_FLAGS+=" -DCMAKE_OSX_SYSROOT=/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.9.sdk"
fi

# Set location for FFTW3 on both linux and mac
CMAKE_FLAGS+=" -DFFT3_INCLUDES=$PREFIX/include"
if [[ "$OSTYPE" == "linux-gnu" ]]; then
    CMAKE_FLAGS+=" -DFFT3_LIBRARY=$PREFIX/lib/libfftw3f.so"
    CMAKE_FLAGS+=" -DFFT3_THREADS_LIBRARY=$PREFIX/lib/libfftw3f_threads.so"
elif [[ "$OSTYPE" == "darwin"* ]]; then
    CMAKE_FLAGS+=" -DFFT3_LIBRARY=$PREFIX/lib/libfftw3f.dylib"
    CMAKE_FLAGS+=" -DFFT3_THREADS_LIBRARY=$PREFIX/lib/libfftw3f_threads.dylib"
fi

# TODO: What do we do about other dependencies, such as pdflatex and doxygen?
if [[ "$OSTYPE" == "linux-gnu" ]]; then
    export PATH=${PATH}
elif [[ "$OSTYPE" == "darwin"* ]]; then
    # LaTeX
    export PATH=${PATH}:/usr/texbin/
fi

# Build in subdirectory.
mkdir build
cd build
cmake .. $CMAKE_FLAGS
make -j$CPU_COUNT all DoxygenApiDocs sphinxpdf # build docs as well
make install

# Install Python wrappers.
export OPENMM_INCLUDE_PATH=$PREFIX/include
export OPENMM_LIB_PATH=$PREFIX/lib
cd python
$PYTHON setup.py install
cd ..

# Remove one random file
#rm $PREFIX/bin/TestReferenceHarmonicBondForce

# Copy all tests to bin directory so they will be distributed with install package.
#cp `find . -name "Test*" -type f -maxdepth 1` $PREFIX/bin

# Put docs into a subdirectory.
cd $PREFIX/docs
mkdir openmm
mv *.pdf *.html api-* openmm/

# Put examples into an appropriate subdirectory.
mkdir $PREFIX/share/openmm/
```

COORDINATING DEVELOPERS

HOW CAN WE COORDINATE A DOZEN
DEVELOPERS ACROSS MULTIPLE CONTINENTS?



<http://github.com>

This image is a screenshot of a GitHub repository page for 'pandegroup/openmm'. The page includes various annotations with arrows pointing to specific features:

- An arrow points from the top left to the search bar with the text "search for anything!".
- An arrow points from the top right to the 'Unwatch' button with the text "get emails when stuff changes".
- An arrow points from the top left to the organization name 'pandegroup / openmm' with the text "organizations can have multiple codebases".
- An arrow points from the top right to the 'Issues' section with the text "see who did stuff".
- An arrow points from the left side to the sidebar with the text "browse the code from the web".
- An arrow points from the center to the commit history with the text "easily manager releases".
- An arrow points from the right side to the right sidebar with the text "talk about stuff".
- An arrow points from the bottom left to the commit history with the text "wikis are so 2002 but you can have those too".
- An arrow points from the bottom right to the right sidebar with the text "propose changes from your fork and discuss them!".

VERSION KONTROL

ZOMG METRICS

organizations can have multiple codebases

search for anything!

get emails when stuff changes

pandegroup / openmm

see who did stuff

talk about stuff

browse the code from the web

easily manager releases

wikis are so 2002 but you can have those too

propose changes from your fork and discuss them!

OpenMM is a toolkit for molecular simulation using high performance GPU code.

4,418 commits 2 branches 10 releases 20 contributors

Branch: master / +

peastman Merge pull request #1178 from peastman/nancheck

Latest commit 56d3b8 8 days ago

cmake_modules Update FindOpenCL.cmake 4 months ago

devtools Add a test for AmoebaTorsionTorsionForce that fails because stripUnit... 19 days ago

docs-source Created OPENMM_DEFAULT_PLATFORM environment variable a month ago

examples Use MTSIntegrator for AMOEBA benchmarks 16 days ago

libraries Fixed compilation errors on Windows a month ago

olla One file that got missed in the last commit 16 days ago

openmmpapi More optimizations to CPU platform a month ago

platforms Added checks for nan coordinates in CUDA and OpenCL 9 days ago

plugins Removed some unnecessary double precision 10 days ago

serialization Added openmmVersion attribute to serialized XML files a month ago

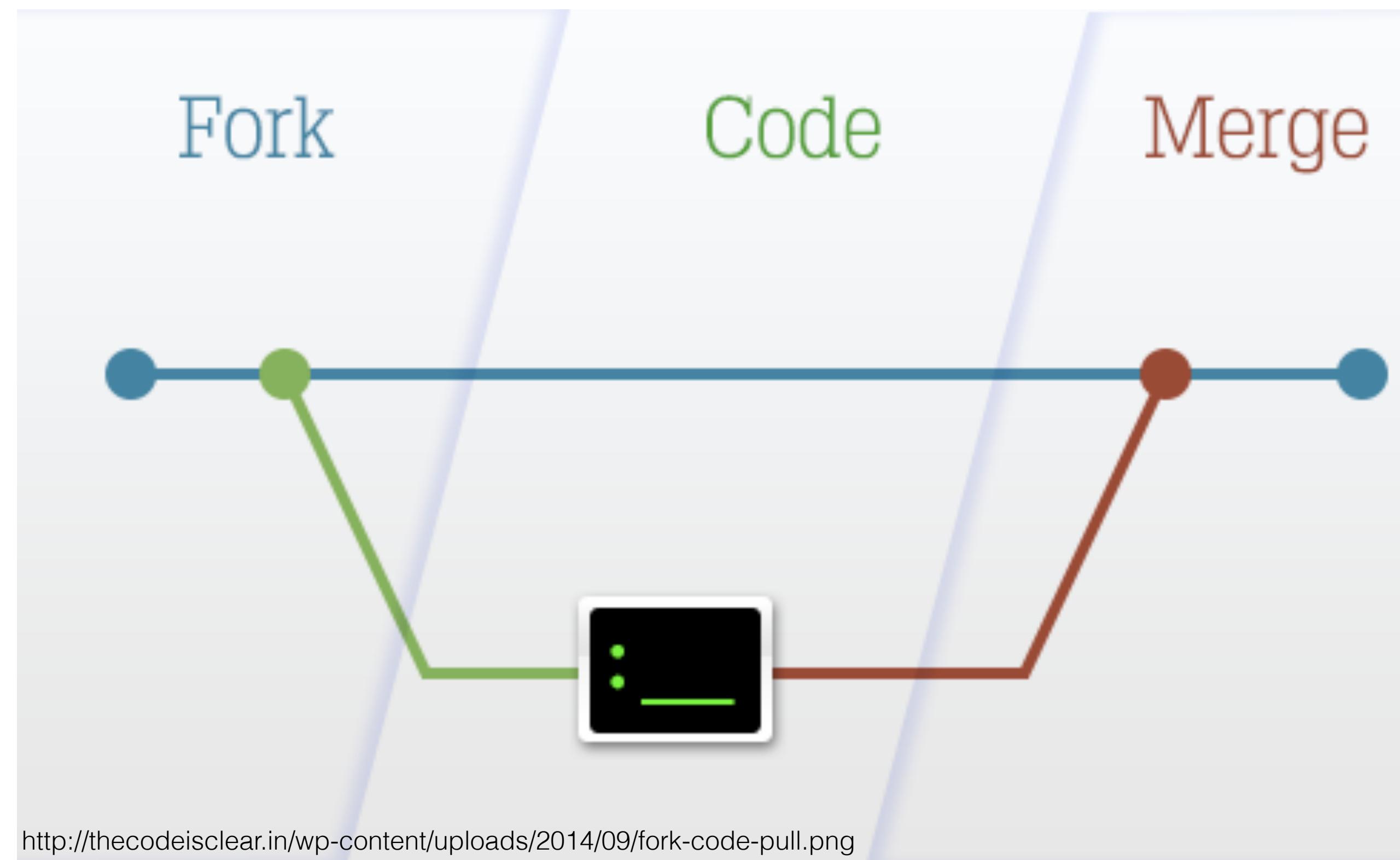
HTTPS clone URL
https://github.com

You can clone with HTTPS, SSH, or Subversion.

Clone in Desktop

Download ZIP

FORK-CODE-MERGE MODEL



MERGING HANDLED BY PULL REQUESTS

hubot / **Spoon-Knife**
forked from octocat/Spoon-Knife

Watch 0 Star 0 Fork 34,641

octocat:master ... hubot:master Edit

Create pull request Discuss and review the changes in this comparison with others.

1 commit 1 file changed 0 commit comments 1 contributor

Commits on Sep 10, 2014

hubot Change description of the repository 8b7afff

Showing 1 changed file with 1 addition and 1 deletion.

Unified Split

2 README.md

@@ -1,6 +1,6 @@
1 1 ### Well hello there!
2 2
3 -This repository is meant to provide an example for *forking* a repository on GitHub.
3 +This is my fork of the octocat/Spoon-Knife repository.
4 4
5 5 Creating a *fork* is producing a personal copy of someone else's project. Forks act as a sort of bridge
 between the original repository and your personal copy. You can submit *Pull Requests* to help make other
 people's projects better by offering your changes up to the original project. Forking is at the core of
 social coding at GitHub.
6 6

DISCUSSION CAN HELP REFINE CODE

[WIP] Per-atom parameters can be taken from residue templates #1182

[Open](#) peastman wants to merge 1 commit into `pandegroup:master` from `peastman:residueparams`

Conversation 6 Commits 1 Files changed 3 +220 -119

peastman commented 9 days ago

This fixes #1168.

I still need to write the documentation for this. Also, there are a couple of force types that aren't yet supported. AmoebaMultipoleForce is really complicated, and I'm not sure how this should work for it. For example, it has two different sets of tags for per-atom properties, each with its own attributes: `<Multipole>` and `<Polarize>`. And then there's AmoebaGeneralizedKirkwoodForce. I have no idea what the heck is going on with that code! Its `addParticle()` method takes three arguments: `charge`, `radius`, and `scalingFactor`. The XML format, on the other hand, only has two attributes: `charge` and `shct`. But it doesn't actually use them! The generator completely ignores the attributes, instead getting the charge from the AmoebaMultipoleForce, selecting the radius based on the element, and hardcoding the scaling factor as 0.69. So until I can figure out what's going on with that, I'm not going to touch it.

Anyway, if you want to start trying this out, it should be functional.

Per-atom parameters can be taken from residue templates 65a581b

jchodera commented 8 days ago

Do you want to tackle the changes to `<Bond>` to permit `fromName` and `toName` in a separate PR? I also realize that the same extension should apply to `<ExternalBond>` as well.

Labels None yet

Milestone No milestone

Assignee No one assigned

Notifications [Unsubscribe](#) You're receiving notifications because you commented.

2 participants

CAN COMMENT ON INDIVIDUAL LINES OF CODE



juliangiaca started a discussion in the diff 4 hours ago

script/background/cron_hourly_tasks.rb [View full changes](#)

```
... ... @@ -32,9 +32,6 @@  
32   32 # Run next application migration request  
33   33 Async::Command.new('script/background/migrate_applications.rb -v -q').notify(CORE).enqueue  
34   34  
35 -# Lamest. Hack. Ever.  
36 -Async::Command.new('script/background/clear_orphaned_jobs.rb --log_level info').notify('
```

2

juliangiaca 4 hours ago [Edit](#) [Delete](#)

Has this been intentionally removed?

aughr 4 hours ago [Edit](#) [Delete](#)

It's a BJ-specific hack for when the runner dies while the job is still running, leaving an orphaned record behind. I don't think that can happen in Resque.

[Add a line note](#)

I WAS HAVING FUN



**UNTIL YOU BROKE
THE BUILD**

THOROUGH TESTING IS REALLY IMPORTANT

From: Berk Hess <gmx3@hotmail.com>
Date: January 29, 2009 4:21:41 AM CST
To: Discussion list for GROMACS users <gmx-users@gromacs.org>, <chris.neale@utoronto.ca>, <dmobley@gmail.com>
Subject: free energy with TIP4P bug fixed

Hi,

The Coulomb energy difference that Chris Neale observed recently was caused by a bug in the neighborlist assignment with the combination of free energy and tip4p water optimization.
This bug would cause a few tip4p-tip4p charge interactions to be missing.
I think it has been present in all Gromacs version which have tip4p optimized loops,
for sure it was in 3.3.
I have fixed this for the upcoming Gromacs 4.0.4 release.

I assume this bug also caused the cut-off dependence that David Mobley observed.

I have done a lot of free energy calculation with tip4p and never noticed any problems. This was because I always had the perturbed molecule in a separate energy group, which circumvents the problem.

So for the moment and for checking if you had the problem with older Gromacs versions, you can simply put the perturbed atoms and tip4p in separate energy groups.

Berk

CVS revision history for gromacs:

Erik Lindahl 27 Dec 2004 Added TIP4P loop optimization The nonbonded kernels and neighborsearching has been updated to allow TIP4p-specific optimization as well as SPC. All references to the MNO general solvent loops have been removed.

Erik Lindahl 27 Dec 2004 TIP4p assembly loops Assembly loops optimized for TIP4p have been added both for SSE & SSE2. There are also new assembly loops for standard coulomb or reaction-field combined with tabulated LJ, so now all interactions except for buckingham are accelerated.

Erik Lindahl 27 Dec 2004 New solvent detection The solvent detection code has been rewritten from scratch: It now finds all charge groups that fulfill the conditions for either the SPC/TIP3p nonbonded kernels, or the TIP4P kernels. Parameters are compared between groups, and only the most abundant type is set to the optimized solvent. (The previous code never checked if the parameters matched between groups).

Erik Lindahl 27 Dec 2004 Reverted atom order in TIP4p The TIP4p coordinates and itp files now have the atoms in standard order: OW HW1 HW2 MW Note that this is the only order which will enable the new TIP4p-specific assembly loops.

We had thoroughly tested three-site water free energies against literature values and other programs; hadn't even realized four-site waters used a different codepath until looking back at old CVS logs!

Unit tests/integration tests/validation/QA absolutely essential.

CONTINUOUS INTEGRATION TO THE RESCUE

Travis CI Blog Status Help John Chodera 

Search all repositories 

My Repositories +

✓ choderalab/openpathsampling	# 1266
⌚ Duration: 22 min 58 sec	
📅 Finished: about 11 hours ago	
✗ choderalab/perses	# 40
⌚ Duration: 1 min 14 sec	
📅 Finished: about 19 hours ago	
✓ choderalab/enssembler	# 148
⌚ Duration: 24 min 3 sec	
📅 Finished: 2 days ago	

choderalab / openmmtools  build passing 

Current Branches Build History Pull Requests

Pull Request #98 Changed switch defaults for WaterBox; added FlexibleDischarge
Added PMEWaterBox
John Chodera authored and committed

240 passed Commit c916375 #98: Changed switch defaults fo ran for 39 min 3 days ago

Build Jobs

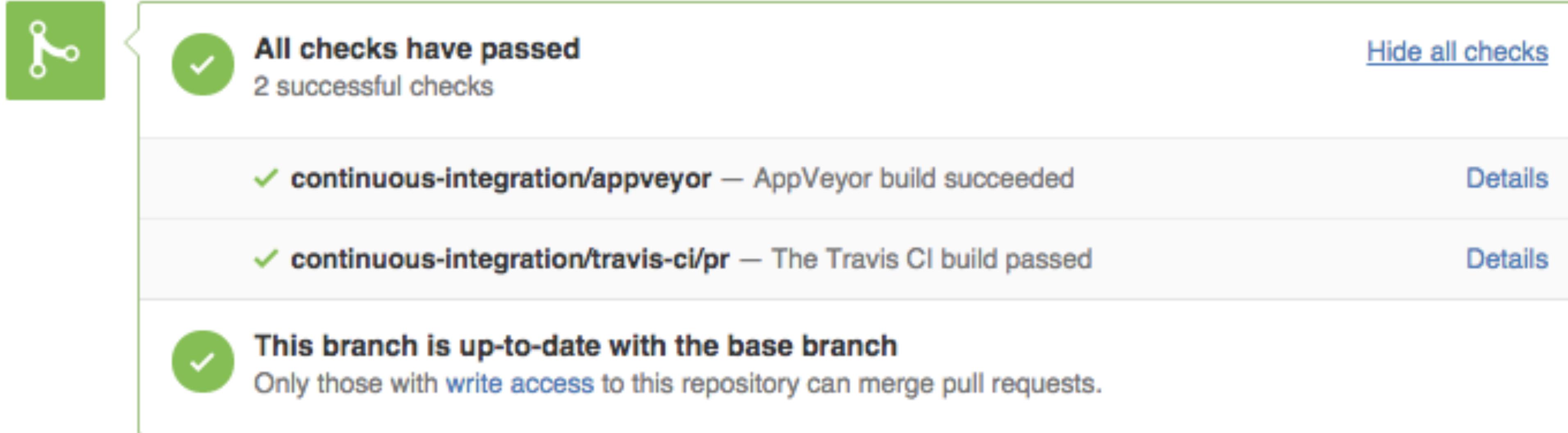
✓ # 240.1	∅ </> no language set	CONDA_PY=27	⌚ 13 min 15 sec
✓ # 240.2	∅ </> no language set	CONDA_PY=33	⌚ 12 min 43 sec
✓ # 240.3	∅ </> no language set	CONDA_PY=34	⌚ 13 min 2 sec

<http://travis-ci.org>

CONTINUOUS INTEGRATION TO THE RESCUE

```
.travis.yml:  
language: c  
  
branches:  
  only:  
    - master  
  
install:  
  - source devtools/ci/install.sh  
  - export PYTHONUNBUFFERED=true  
  
script:  
  # Set package name for downstream devtools.  
  - export PACKAGENAME="openmmtools"  
  # this builds the binary, unpacks it, and runs the tests  
  - conda build devtools/conda-recipe  
  
env:  
  matrix:  
    - CONDA_PY=27  
    - CONDA_PY=33  
    - CONDA_PY=34  
  global:  
    # encrypted BINSTAR_TOKEN for push of dev package to binstar  
    - secure: "YIpTwys0xPT1XLKgddr3lZmBoL8UleB4Gk3eDI+mgt4G0cfI2U/xdx+0053wh1KpkoxRulK/  
1mA7oX3v5zTDMwCHRjZ8VAuCwUPBhkFbJ35IgFQP/P0nDkW5odyGTZ0V7JNFaC07KyzSQWRaTNGsPPjas3lprdQawrpP0PNLzao="  
  
after_success:  
  - echo "after_success"  
  - source devtools/ci/after_success.sh
```

CONTINUOUS INTEGRATION TO THE RESCUE



A screenshot of a continuous integration status card. It features a green icon of a hand holding a gear. The main message is "All checks have passed" with "2 successful checks". Below this, two items are listed with green checkmarks: "continuous-integration/appveyor — AppVeyor build succeeded" and "continuous-integration/travis-ci/pr — The Travis CI build passed". Each item has a "Details" link to its right. At the bottom, there's a note: "This branch is up-to-date with the base branch" followed by the instruction "Only those with write access to this repository can merge pull requests." A "Hide all checks" link is located in the top right corner of the card.

All checks have passed
2 successful checks

continuous-integration/appveyor — AppVeyor build succeeded

continuous-integration/travis-ci/pr — The Travis CI build passed

This branch is up-to-date with the base branch
Only those with write access to this repository can merge pull requests.

[Hide all checks](#)

win

linux [+ osx]

CONTINUOUS INTEGRATION TO THE RESCUE

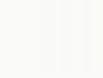
pandegroup / openmm  build passing

Current Branches Build History Pull Requests > **Build #1423** 

 Pull Request #1182 [WIP] Per-atom parameters can be taken from residue temp
 Per-atom parameters can be taken from residue templates
 peastman authored and committed

1423 passed
Commit ba006bf
#1182: [WIP] Per-atom paramet
ran for 37 min 39 sec
9 days ago

Build Jobs

 # 1423.1	 Python: 2.7_with_system_site_  OPENMM_BUILD_STATIC_LIB="OFF"	⌚ 8 min 57 sec
 # 1423.2	 Python: 2.7_with_system_site_  OPENMM_BUILD_STATIC_LIB="ON"	⌚ 9 min 41 sec
 # 1423.3	 Python: 3.4  OPENMM_BUILD_STATIC_LIB="OFF"	⌚ 9 min 3 sec
 # 1423.4	 Python: 3.4  OPENMM_BUILD_STATIC_LIB="ON"	⌚ 9 min 58 sec

<http://travis-ci.org>

DEV PACKAGE BUILDS AUTOMATICALLY PUSHED TO ANACONDA CLOUD

after_success:

```
anaconda -t $ANACONDA_TOKEN upload --force -u omnia -p mdtraj-dev $HOME/miniconda/conda-bld/linux-64/mdtraj-dev-*
```

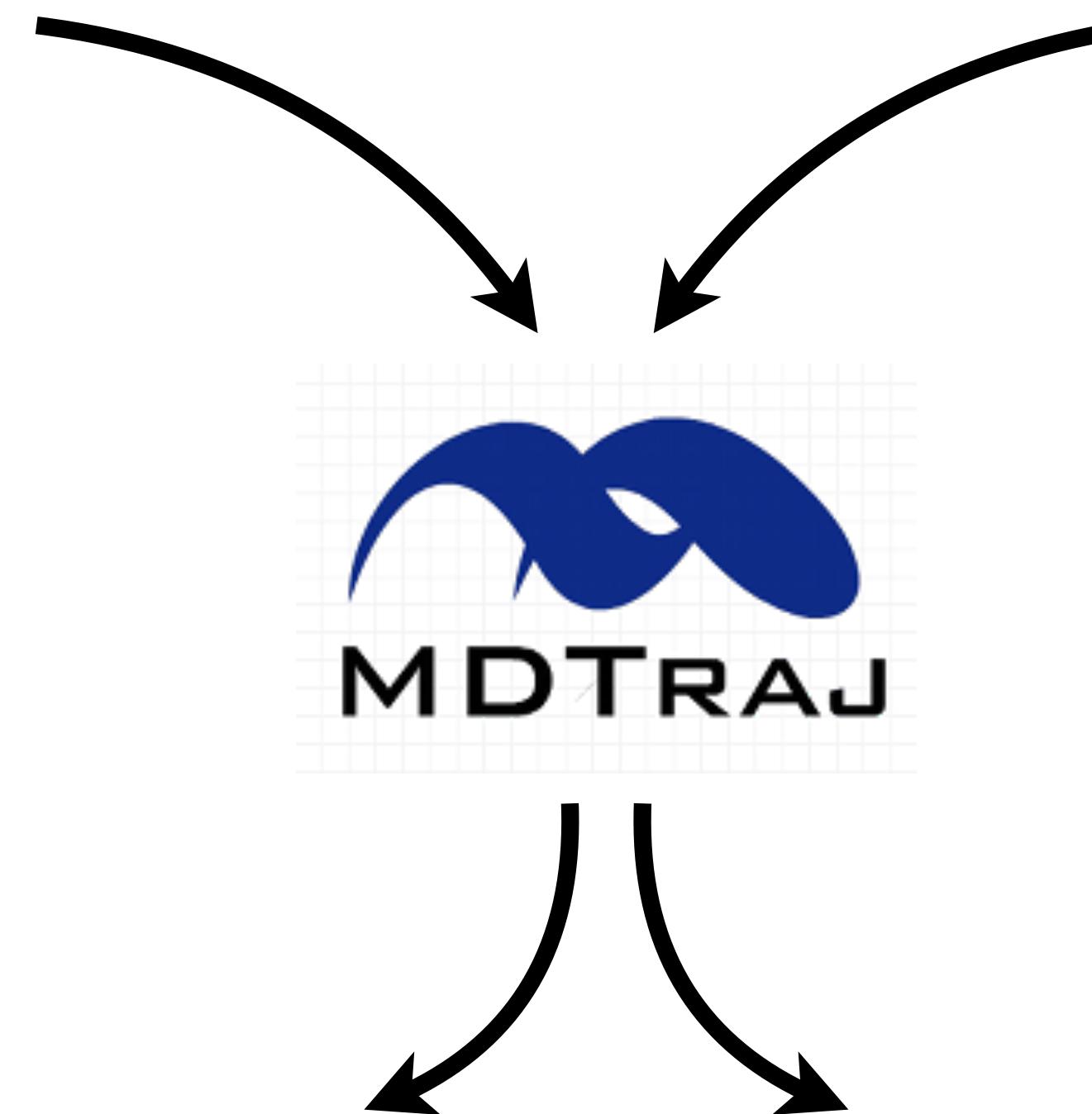
ONLY WORKING BUILDS ARE PUSHED!

**CAVEAT: CAN ONLY RUN LIMITED CPU TESTS;
DON'T YET HAVE GPU COMPUTE-INTENSIVE CLOUD TESTS WORKING
(SOMETHING WE REALLY NEED FOR TESTING MD)**

STANDARDIZE ON A FEW PACKAGES FOR INTEROPERABILITY



OpenMM



GROMACS
CHARMM
AMBER
OTHERS

BUILDING RELEASE PACKAGES

 [omnia-md / conda-recipes](https://github.com/omnia-md/conda-recipes) Unwatch 9 Star 5 Fork 13

conda build recipes for the Omnia project <http://omnia.md> — [Edit](#)

933 commits 1 branch 0 releases 13 contributors

Branch: master [conda-recipes / +](#)

rmcgibbo	Update meta.yaml	Latest commit ff3204 4 days ago
alchemy	Simplify	8 days ago
algopy	Use the new skip mechanism	8 days ago
ambermini	Faster	2 months ago
appdirs	Add pyvisfile test	6 months ago
assaytools	Updated recipe to fix errors.	8 months ago
behave	MAINT: Update recipes to latest pypi versions	6 months ago
bhmm	Pull from URL instead of git_url	6 days ago
boost	Use the new skip mechanism	8 days ago
cclib	cclib from url instead of git	8 days ago
clapack	MAINT: Add missing summaries	6 months ago
clusterutils	Updated version in clusterutils/meta.yaml	9 months ago
covar	Update covar package	6 days ago

Code Issues Pull requests Wiki Pulse Graphs Settings

SSH clone URL
`git@github.com:omr` [Copy](#)

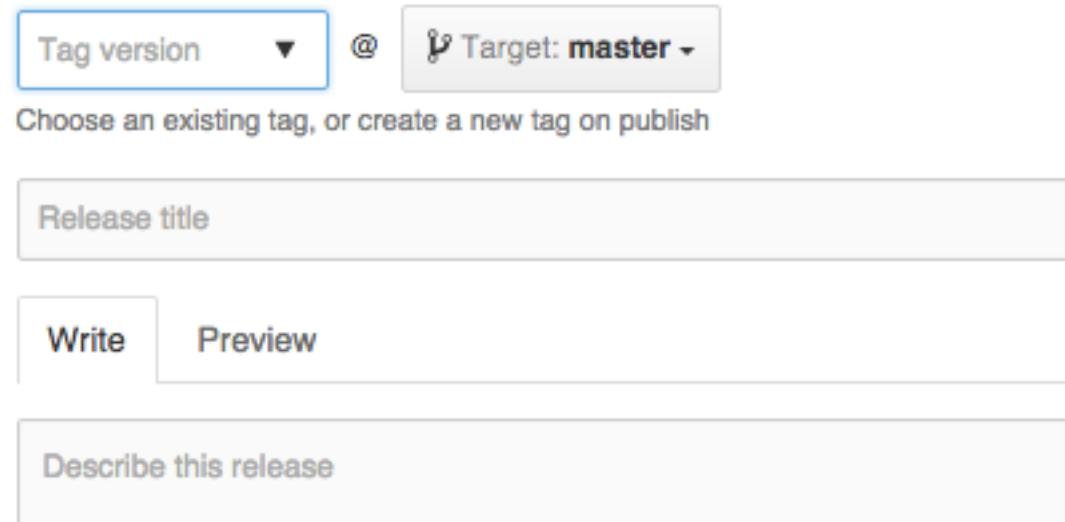
You can clone with [HTTPS](#), [SSH](#), or [Subversion](#). ?

[Clone in Desktop](#) [Download ZIP](#)

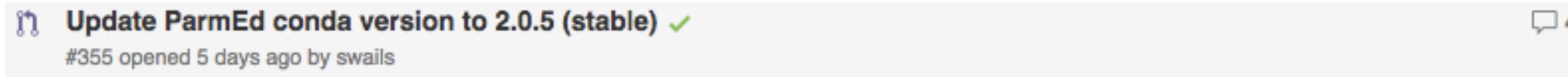
[http://github.com/omnia-md/conda-recipes](https://github.com/omnia-md/conda-recipes)

BUILDING RELEASE PACKAGES

1. CUT A RELEASE ON GITHUB



2. UPDATE CONDA PACKAGE IN CONDA-RECIPES REPO



3. UPDATED RECIPES AUTOMAGICALLY REBUILT AND PUSHED (CURRENTLY JENKINS BOTS, SOON THE CLOUD)

<http://github.com/omnia-md/conda-recipes>

PREDICTIVE CHALLENGES FOR MOLECULAR SIMULATIONS

DRUG DESIGN DATA RESOURCE (D3R)

CONTACT / PRESS

<http://drugdesigndata.org>



D3R

- * protein-ligand cocrystal structures
- * ligand binding affinities

SAMPL

- * cyclohexane-water distribution coefficients
- * host-guest binding affinities

2016 WORKSHOPS

CAMBRIDGE/BOSTON MA, 16-20 MAY 2016

ALCHEMICAL FREE ENERGY METHODS IN DRUG DISCOVERY

Email list signup: <https://goo.gl/9FXPUZ>

orgs: Michael Schnieders, Michael Shirts, David Mobley, John Chodera, Vijay Pande

MARKOV STATE MODELS IN DRUG DISCOVERY

Email list signup: <https://goo.gl/9FXPUZ>

orgs: John Chodera, Rommie Amaro, Benoît Roux, Vijay Pande, Frank Noé, Greg Bowman

THE CHODERA LAB @ MSKCC



Code and data available at <http://www.choderalab.org>



START FOLDING

COLLABORATORS

<http://folding.stanford.edu>

Stanford

Vijay Pande

Peter Eastman

IBM Almaden

Bill Swope

Jed Pitera

Julia Rice

University of Chicago

Nina Singhal Hinrichs

UC Irvine

David Mobley

CCNY

Marilyn Gunner

OpenEye

Christopher Bayly

Anthony Nicholls

Stony Brook

Ken Dill

Markus Seeliger

UCSF

Brian Shoichet

Simon Fraser

David Sivak

University of Virginia

Michael Shirts

Duke

David Minh

Freie Universität Berlin

Frank Noé

Bettina Keller

Jan-Hendrik Prinz

Rutgers

Zhiqiang Tan

University of Edinburgh

Antonia S. J. S. Mey

UC Berkeley

Susan Marqusee

Carlos Bustamante

Christian Kaiser

University of Chicago

Suri Vaikuntanathan

LBNL

Gavin Crooks

Vanderbilt

Joel Tellinghuisen

Hessian Informatics

Kim Branson

AstraZeneca

Michelle Lamb

Melissa Vasbinder

Mihir Date

Merck Darmstadt

Paul Czodrowski

Code and data available at <http://www.choderalab.org>