Computational biomolecular simulation workflows with

BioExcel building blocks

BioExcel Webinar

10/09/2020

Adam Hospital Gasch

Institute for Research in Biomedicine (IRB) – Barcelona - Spain

Partners

Funding

Horizon 2020
European Union Funding
for Research & Innovation

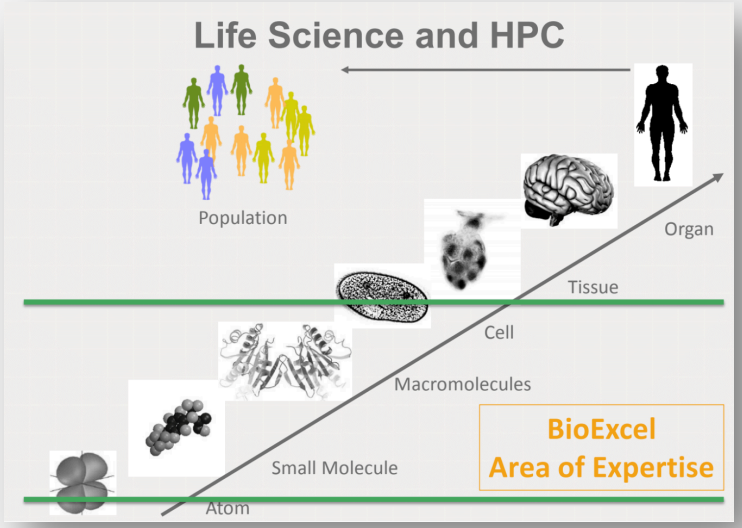
Index

- **BioExcel Center of Excellence (CoE)**
- **Biomolecular workflows**
- **BioExcel Building Blocks** software library **(BioBB)**
- **Biomolecular workflows** using **BioBB**
  - **Demonstration** workflows (Jupyter Notebooks)
  - **Pre-exascale HPC** workflows (PyCOMPSs)
    - ✧ **Example**: **COVID-19** pre-exascale workflows

BioExcel Center of Excellence (CoE)

Centre of Excellence for Computational Biomolecular Research

A central hub for biomolecular modelling and simulations



BioExcel Goals

Centre of Excellence for Computational Biomolecular Research

Enabling **better science** by:

- **Improving** the **performance** and **functionality** of **key applications**
- Providing **support** to **non-experts** and **advanced** users
- Developing **user-friendly computational workflows**

BioExcel Overview: Applications & Support

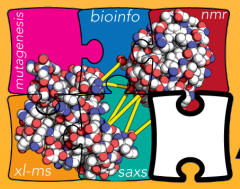
Molecular Dynamics:

GROMACS
FAST. FLEXIBLE. FREE.

Free energy calculations:

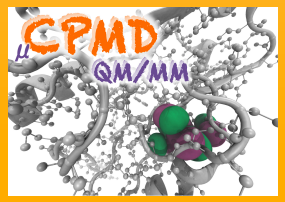
pmx: generate hybrid protein structure and topology
Computational Biomolecular Dynamics Group

Protein-Protein Docking:



ADDOCK
High-Ambiguity Driven Docking

QM / MM:



CP2K

Key Applications

https://bioexcel.eu/

bioexcel

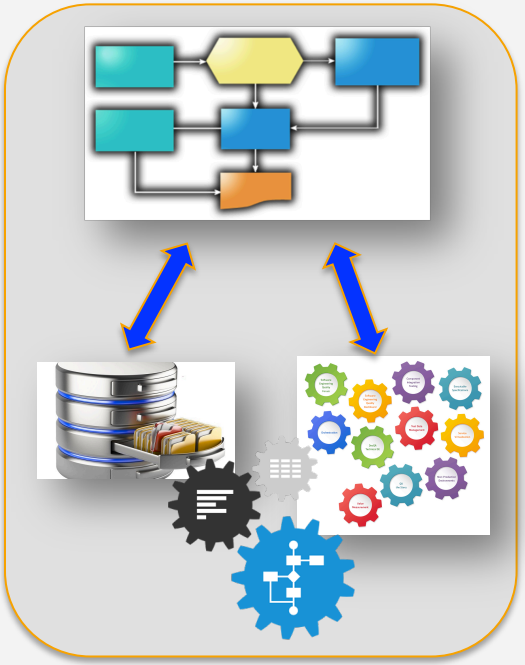
- *Webinars*
- *Training events*
- *Conferences*
- *Workshops*
- *Industry visits*
- *Forum (ask.bioexcel.eu)*

Support

BioExcel Overview: workflows

- **Design, deploy** and **make available** solution-oriented **biomolecular workflows**.

- Excellence in **Usability**.
  - **Ease of use**
  - **Availability**
  - **Reproducibility**
  - **Multi-platform**
  - **Multi-infrastructure**



Use case-driven

Final deployment

IRB BARCELONA INSTITUTE FOR RESEARCH IN BIOMEDICINE

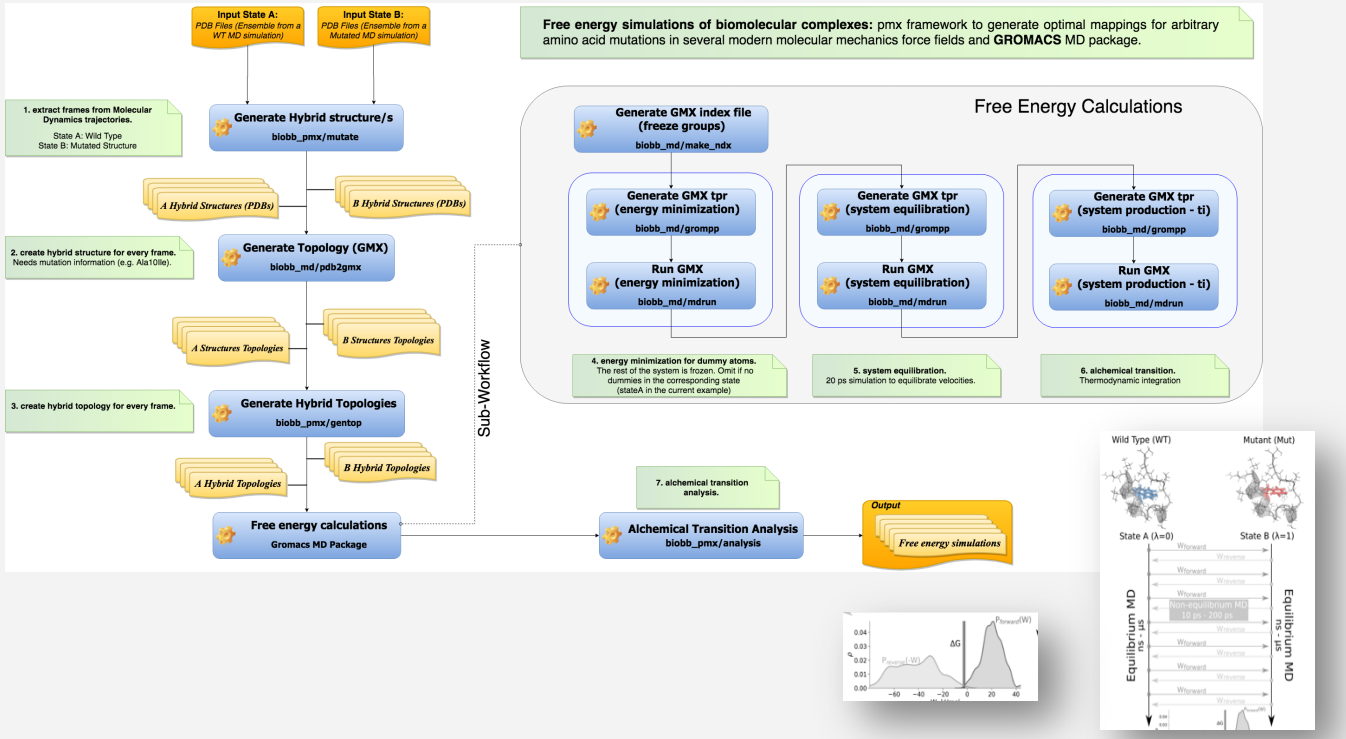
MMB

BSC Barcelona Supercomputing Center
Centro Nacional de Supercomputación

MANCHESTER 1824 The University of Manchester

EMBL-EBI

Biomolecular workflows



Biomolecular workflows: challenges

- **Biomolecular simulation workflows** are usually built from a number of **tools** performing different **tasks**.
  - ***Molecular Structure File format conversions***
  - ***Structure Modelling***
  - ***Molecular Dynamics***
  - ***Quantum Mechanics***
  - ***QM / MM***
  - ***Trajectory analyses***
  - ***Docking***
  - ***Free energy***
  - ***Ligand parameterization***
  - ***Cheminformatics***
  - ***Data analytics***

Biomolecular workflows: challenges

GROMACS
FAST. FLEXIBLE. FREE.

Amber18

AmberTools19

HADDOCK
High-Ambiguity Driven Docking

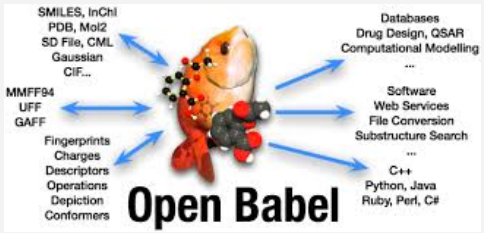
NAMD
Scalable Molecular Dynamics

VMD
Visual Molecular Dynamics

CPMD
QM/MM

CP2K

ACPYPE



RDKit

pmx: generate hybrid protein structure and topology
Computational Biomolecular Dynamics Group

Bio3D

MD ANALYSIS

AutoDock 4

Modeller
Program for Comparative Protein Structure Modelling by Satisfaction of Spatial Restraints

TensorFlow

scikit learn

And many many more...

Biomolecular workflows: challenges

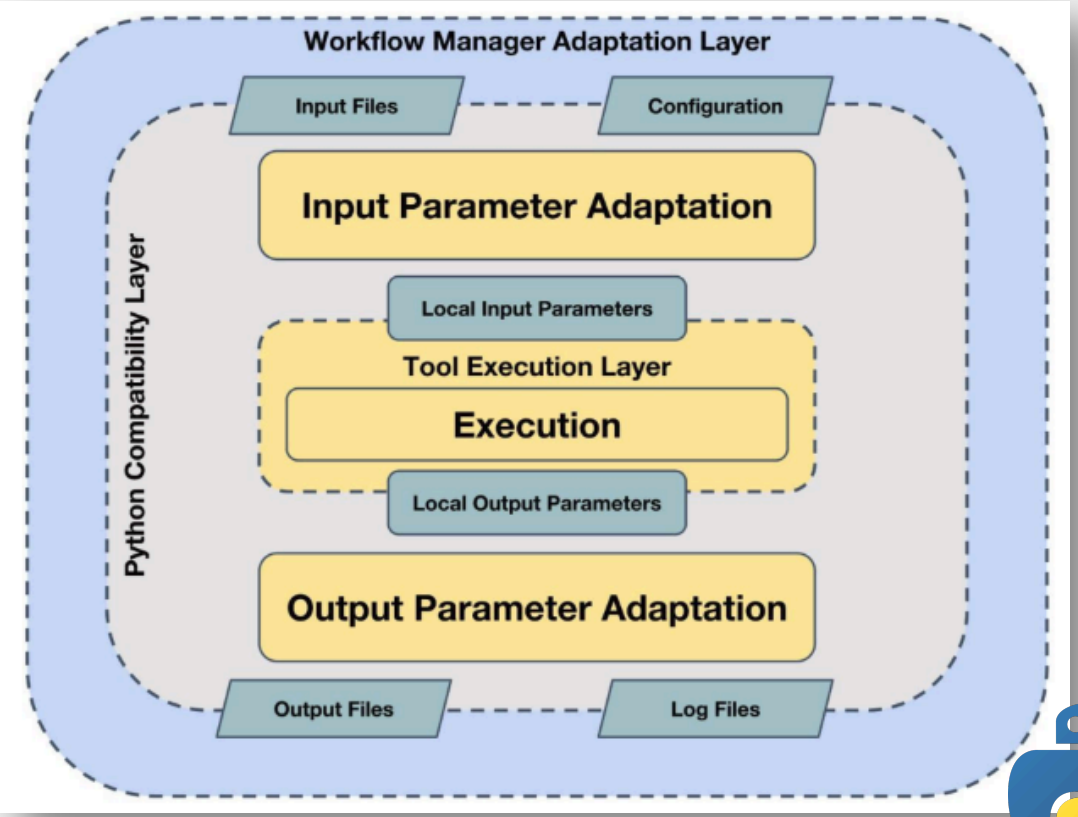
Shell script:

- ***Step 1***
- ***Step 2***
- ***Step n***

Usability
Interoperability
Portability
Reproducibility
Scalability

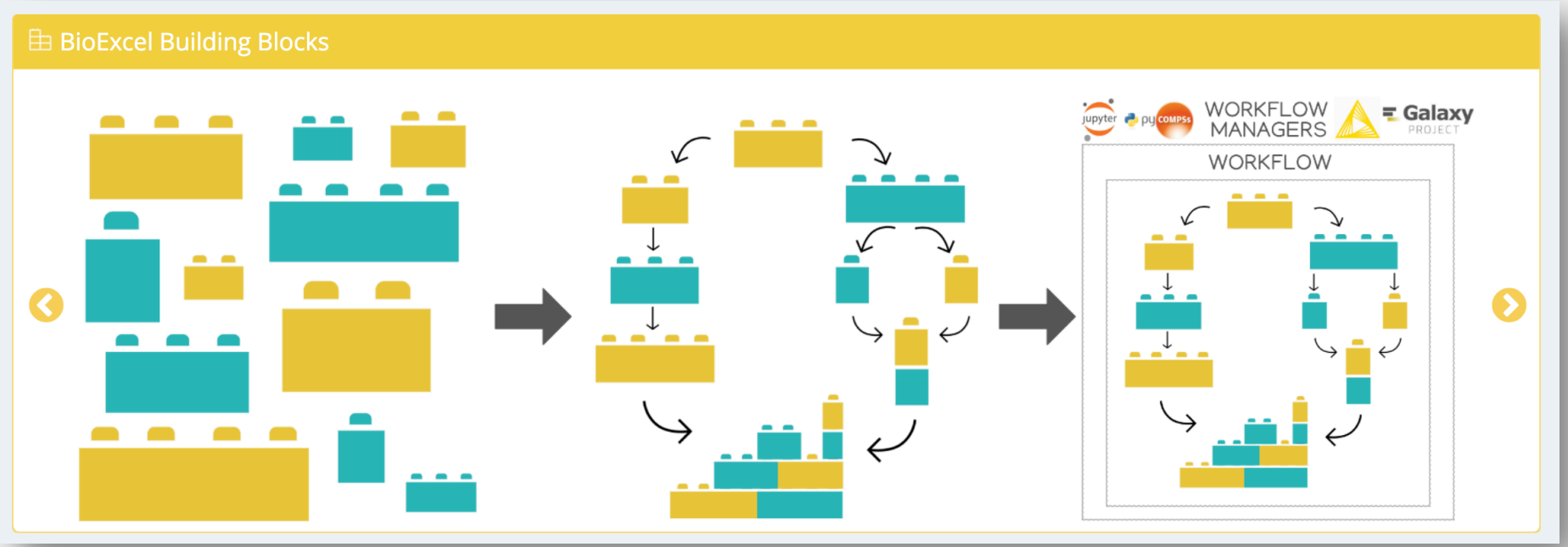
http://mmb.irbbarcelona.org/biobb/

BioExcel Building Blocks



http://mmb.irbbarcelona.org/biobb/

BioExcel Building Blocks software library



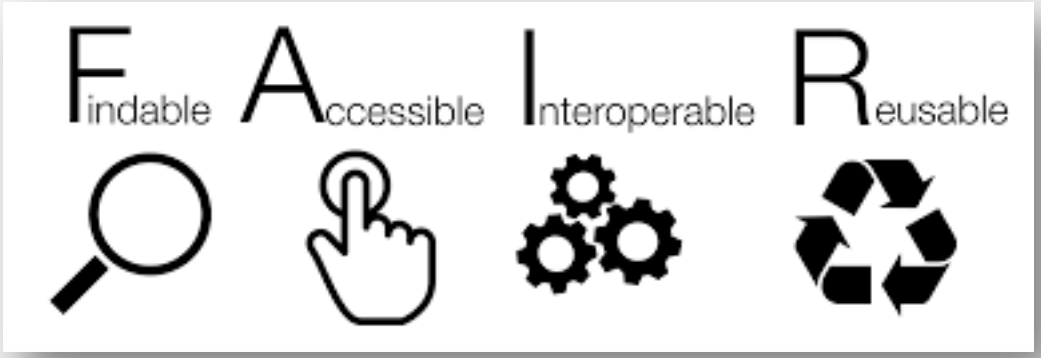
1) Building Blocks

2) Workflows

3) Workflow Managers

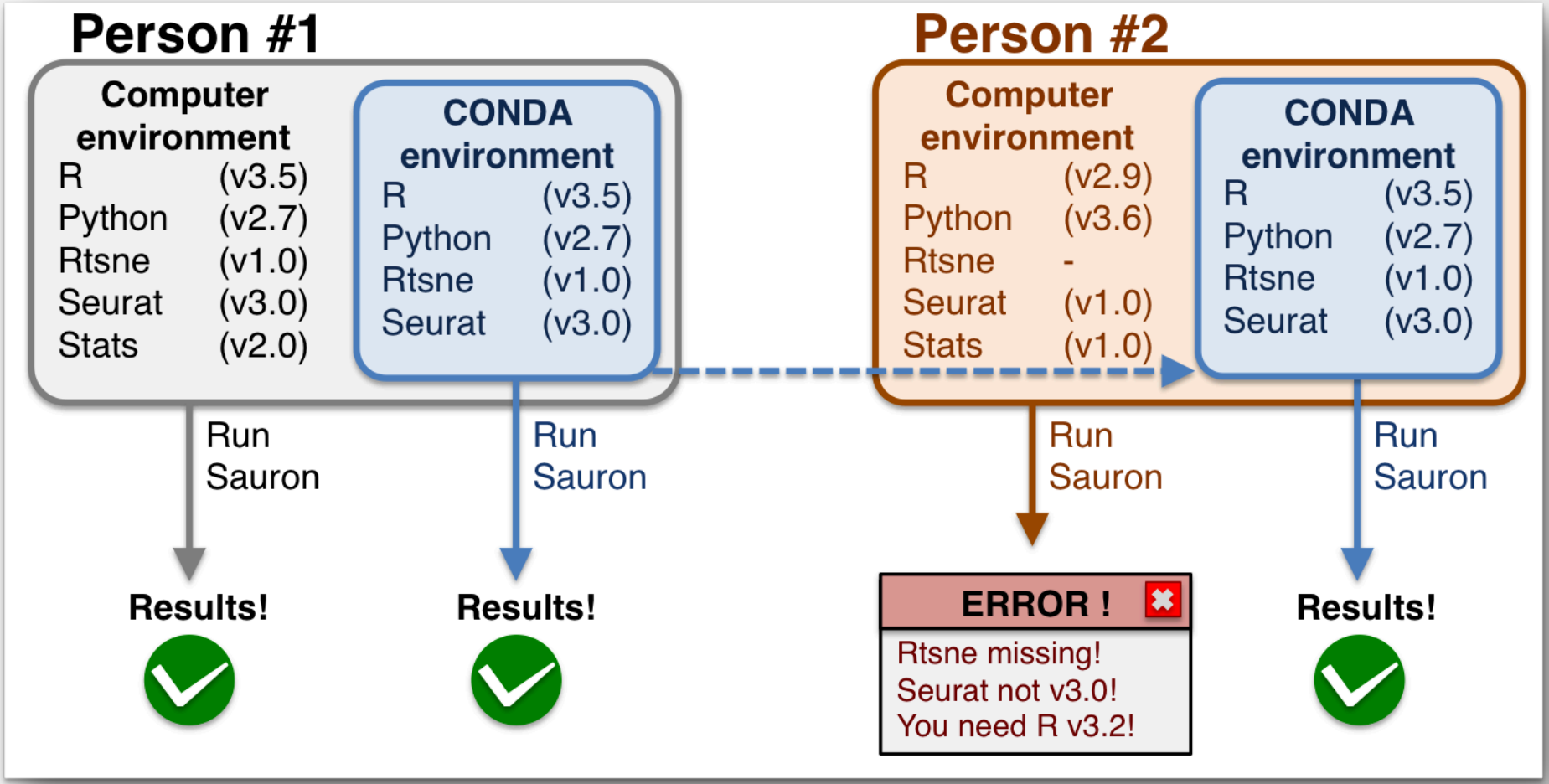
biobb

BioExcel Building Blocks: FAIR



Conda Environments

https://nbisweden.github.io/

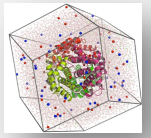


CONDA®

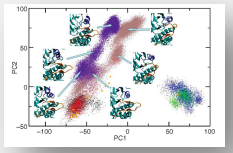
Reproducibility

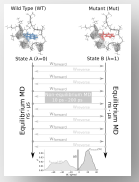
BioExcel Building Blocks modules

- **biobb_common:** base package required to use the biobb library
- **biobb_io**: collection to fetch data from biological databases

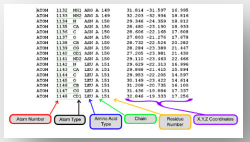


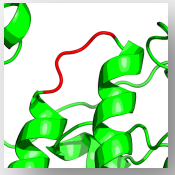
- **biobb_md:** collection to perform Molecular Dynamics simulations
- **biobb_analysis:** collection to perform analysis over MD simulations



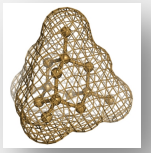


- **biobb_pmx:** collection to perform free energy calculations
- **biobb_structure_utils:** collection to modify or extract information from a PDB





- **biobb_model:** collection to check and model 3D structures
- **biobb_chemistry:** cheminformatics analyses and format conversions



- **biobb_ml:** machine learning algorithms (scikit learn, tensorflow)

And many others: biobb_vs, biobb_cmip, …

BioExcel Building Blocks modules

SOURCE AND DOCS FOR BIOEXCEL BUILDING BLOCKS

http://mmb.irbbarcelona.org/biobb/availability/source

Search by text

Write something

Search by keywords

Select or write keyword(s)

Package	Description	Python	ReadTheDocs	Bioconda	Docker	Singularity	Version
biobb_analysis	Biobb_analysis is the Biobb module collection to perform analysis of molecular dynamics simulations.						3.0.1
biobb_chemistry	Biobb_chemistry is the Biobb module collection to perform chemistry over molecular dynamics simulations.						3.0.1
biobb_common	Biobb_common is the base package required to use the biobb packages						3.0.0
biobb_io	Biobb_io is the Biobb module collection to fetch data to be consumed by the rest of the Biobb building blocks.						3.0.0
biobb_md	Biobb_md is the Biobb module collection to perform molecular dynamics simulations.						3.0.0
biobb_model	Biobb_model is the Biobb module collection to check and model 3d structures, create mutations or reconstruct missing atoms.						3.0.0
biobb_pmx	Biobb_pmx is the Biobb module collection to perform PMX executions.						2.0.2
biobb_structure_utils	Biobb_structure_utils is the Biobb module collection to modify or extract information from a PDB structure file.						3.0.0

BioExcel Building Blocks modules

SOURCE AND DOCS FOR BIOEXCEL BUILDING BLOCKS

http://mmb.irbbarcelona.org/biobb/availability/source

Search by text

Write something

Search by keywords

Select or write keyword(s)

Package	Description	Python	ReadTheDocs	Bioconda	Docker	Singularity	Version
biobb_analysis	Biobb_analysis is the Biobb module collection to perform analysis of molecular dynamics simulations.						3.0.1

conda install biobb_analysis

AmberTools19

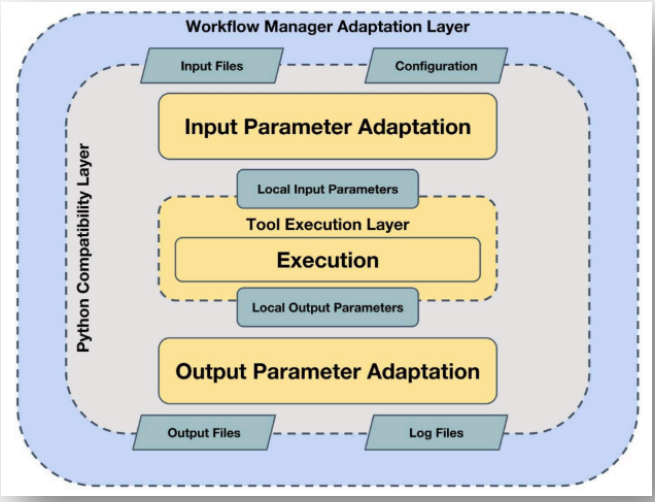
GROMACS
FAST. FLEXIBLE. FREE.

Building block	Wrapped tool	Description
GMXCluster	gmx cluster	Creates cluster structures from a given GROMACS compatible trajectory.
GMXRms	gmx rms	Performs a Root Mean Square deviation (RMSd) analysis from a given GROMACS compatible trajectory.
GMXRgyr	gmx gyrate	Computes the radius of gyration (Rgyr) of a molecule about the x-, y- and z-axes, as a function of time, from a given GROMACS compatible trajectory.
GMXEnergy	gmx energy	Extracts energy components from a given GROMACS energy file.
GMXImage	gmx trjconv	Corrects periodicity (image) from a given GROMACS compatible trajectory file.
GMXTrjconvStr	gmx trjconv	Converts between GROMACS compatible structure file formats and/or extracts a selection of atoms.
GMXTrjconvStrEns	gmx trjconv	Extracts an ensemble of frames containing a selection of atoms from GROMACS compatible trajectory files.
GMXTrjconvTrj	gmx trjconv	Converts between GROMACS compatible trajectory file formats and/or extracts a selection of atoms.
CpptrajAverage	Ambertools cpptraj	Calculates a structure average of a given cpptraj compatible trajectory.
CpptrajBfactor	Ambertools cpptraj	Calculates the Bfactor fluctuations of a given cpptraj compatible trajectory.
CpptrajRms	Ambertools cpptraj	Calculates the Root Mean Square deviation (RMSd) of a given cpptraj compatible trajectory.

BioExcel Building Blocks: Syntax

- **Import Module**
- **Define**:
  - **inputs** & **output** paths
  - **properties** dictionary
- **Launch** building block

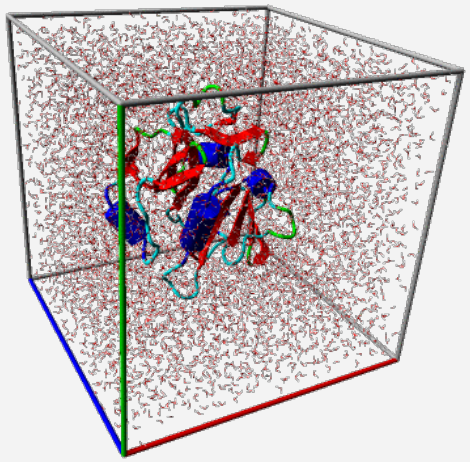
Inputs Paths
Outputs Paths
Properties




```python
# Editconf: Create solvent box
# Import module
from biobb_md.gromacs.editconf import Editconf
```

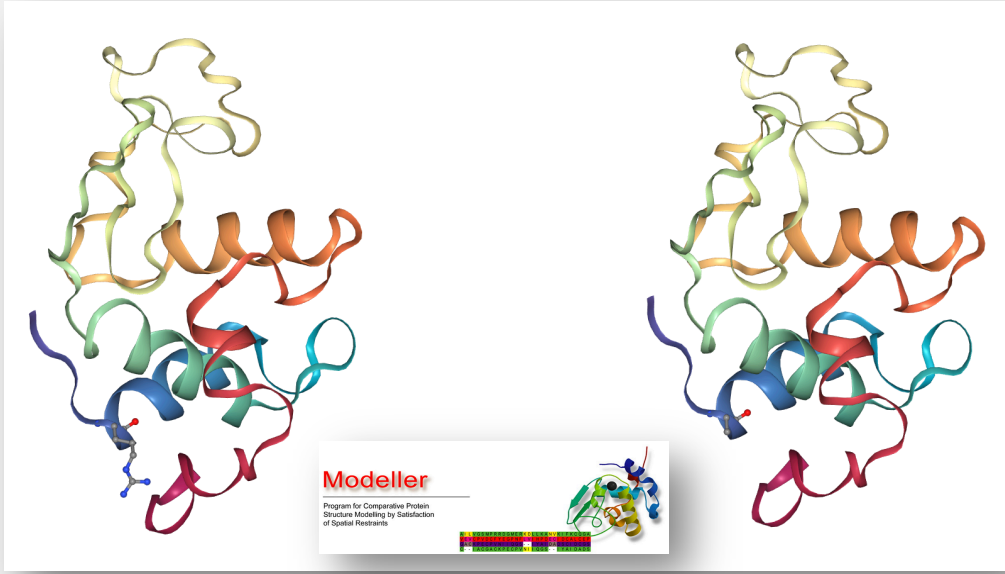
```python
# Create prop dict and inputs/outputs
output_editconf_gro = pdbCode+'_editconf.gro'

prop = {
    'box_type': 'cubic',
    'distance_to_molecule': 1.0
}
```

```python
#Create and launch bb
Editconf(input_gro_path=output_pdb2gmx_gro,
         output_gro_path=output_editconf_gro,
         properties=prop).launch()
```



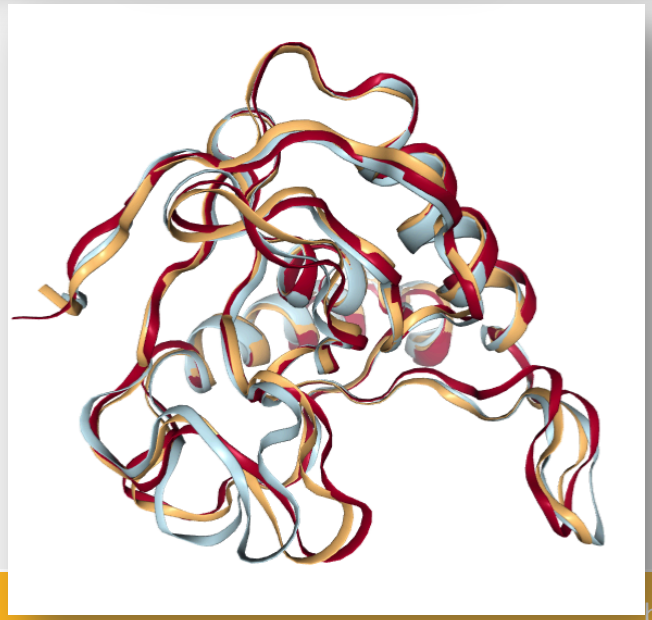
BioExcel Building Blocks: Syntax

```python
# Mutate Residue from PDB
# Import module
from biobb_model.model.mutate import Mutate

# Create prop dict and inputs/outputs
mutated_pdb = pdbCode + '_mutated.pdb'

prop = {
    'mutation_list': 'A:Arg5Ala'
}

# Create and launch bb
Mutate(input_pdb_path=fixed_pdb,
               output_pdb_path=mutated_pdb,
              properties=prop).launch()
```




```python
from biobb_analysis.gromacs.gmx_cluster import GMXCluster

#traj = 'stateA_1ns.xtc'
#top = 'stateA.tpr'

#traj = '1nql_bound.0-200.imaged.rot.10.xtc' # 700MB !!
#top = '1nql_bound.dry.pdb'

traj = '1nql_G39R.0-50.imaged.rot.10.xtc'
top = '1nql_G39R.dry.pdb'

out_ensemble = 'ensemble.pdb'

prop = {
    'fit_selection': 'Protein',
    'output_selection': 'Protein'
}

GMXCluster(
    input_structure_path=top,
    input_traj_path=traj,
    output_pdb_path=out_ensemble,
    properties = prop
).launch()
```

GROMACS
FAST. FLEXIBLE. FREE.

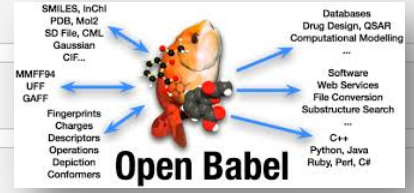


In [ ]:
```
# Ligand: Download ligand structure from MMB PDB mirror REST API (http://mmb.irbbarcelona.org/api/)
# Import module
from biobb_io.api.ligand import Ligand
from biobb_io.api.pdb import Pdb

# Create prop dict and inputs/outputs
input_structure = ligandCode + '.pdb'

prop = {
    'ligand_code' : 'IBP'
}

#Create and launch bb
Ligand(output_pdb_path=input_structure,
       properties=prop).launch()
```

{ REST-API }

In [ ]:
```
from biobb_chemistry.babelm.babel_add_hydrogens import BabelAddHydrogens

ligandFile_wH = 'IBP.H.pdb'

BabelAddHydrogens(input_path=input_structure,
                  output_path=ligandFile_wH,
                 ).launch()
```

In [ ]:
```
view = nglview.show_file(ligandFile_wH)
view
```

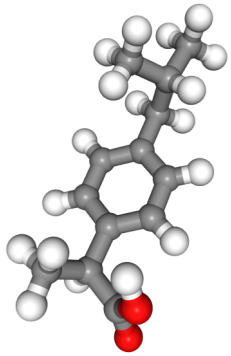


In [ ]:
```
from biobb_chemistry.babelm.babel_minimize import BabelMinimize

ligandFile_wH_min = 'IBP.H.min.pdb'

BabelMinimize(input_path=ligandFile_wH,
              output_path=ligandFile_wH_min,
              properties=prop).launch()
```

In [ ]:
```
view = nglview.show_file(ligandFile_wH_min)
view
```

In [ ]:
```
from biobb_chemistry.acpype.acpype_params_gmx import AcpypeParamsGMX

output_acpype_gro = 'IBP_params.gro'
output_acpype_itp = 'IBP_params.itp'
output_acpype_top = 'IBP_params.top'
output_acpype = 'IBP_params'

prop = {
    'basename' : 'IBP'
}

AcpypeParamsGMX(input_path=ligandFile_wH_min,
                output_path_gro=output_acpype_gro,
                output_path_itp=output_acpype_itp,
                output_path_top=output_acpype_top,
                properties=prop).launch()
```

ACPYPE



BioExcel Building Blocks: Documentation

acpype.acpype_params_gmx module

Module containing the Acpype class and the command line interface.

class **acpype.acpype_params_gmx.AcpypeParamsGMX**(*input_path*, *output_path_gro*, *output_path_itp*, *output_path_top*, *properties=None*, ***kwargs*) [source]

Bases: `object`

Small molecule parameterization for GROMACS MD package. Wrapper for the Acpype module. Generation of topologies for GROMACS. Acpype is a tool based in Python to use Antechamber to generate topologies for chemical compounds and to interface with others python applications like CCPN or ARIA. Visit the official page.

Parameters:

- **input_path** (*str*) –
  Path to the input file. File type: input. Sample file. Accepted formats: pdb, mdl, mol2.
- **output_path_gro** (*str*) –
  Path to the GRO output file. File type: output. Sample file. Accepted formats: gro.
- **output_path_itp** (*str*) –
  Path to the ITP output file. File type: output. Sample file. Accepted formats: itp.
- **output_path_top** (*str*) –
  Path to the TOP output file. File type: output. Sample file. Accepted formats: top.
- **properties** (*dic*) –
  - **basename** (*str*) - ("BBB") A basename for the project (folder and output files).
  - **charge** (*int*) - (0) Net molecular charge, for gas default is 0.
  - **acpype_path** (*str*) - ("acpype") Path to the acpype executable binary.
  - **remove_tmp** (*bool*) - (True) [WF property] Remove temporal files.
  - **restart** (*bool*) - (False) [WF property] Do not execute if output files exist.
  - **container_path** (*str*) - (None) Container path definition.
  - **container_image** (*str*) - ('mmbirb/acpype:latest') Container image definition.
  - **container_volume_path** (*str*) - ('/tmp') Container volume path definition.
  - **container_working_dir** (*str*) - (None) Container working directory definition.
  - **container_user_id** (*str*) - (None) Container user_id definition.
  - **container_shell_path** (*str*) - ('/bin/bash') Path to default shell inside the container.

BioExcel Building Blocks: Publication

MENU SCIENTIFIC DATA

Article | Open Access | Published: 10 September 2019

BioExcel Building Blocks, a software library for interoperable biomolecular simulation workflows

Pau Andrio, Adam Hospital, Javier Conejero, Luis Jordá, Marc Del Pino, Laia Codo, Stian Soiland-Reyes, Carole Goble, Daniele Lezzi, Rosa M. Badia, Modesto Orozco & Josep Ll. Gelpi

Scientific Data **6**, Article number: 169 (2019) | Cite this article

Innovation Radar

Search by keyword...

Exploring

Innovations actively exploring value creation opportunities.

The Innovation Radar is a European Commission initiative to identify high potential innovations and innovators in EU-funded research and innovation framework programmes (tell me more)

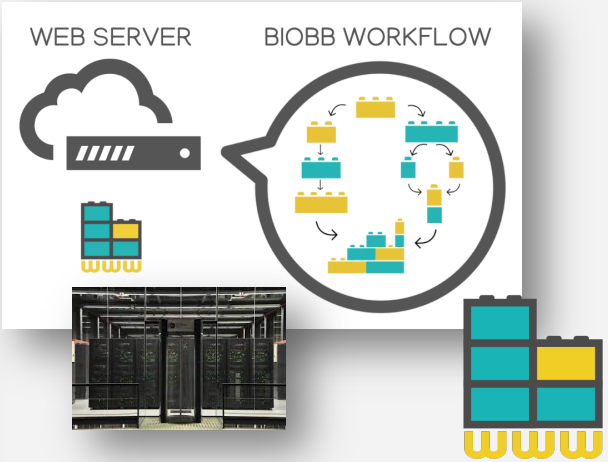
BioExcel Building Blocks: Versatility

LIBRARY VERSATILITY TUTORIALS

Common Workflow Language

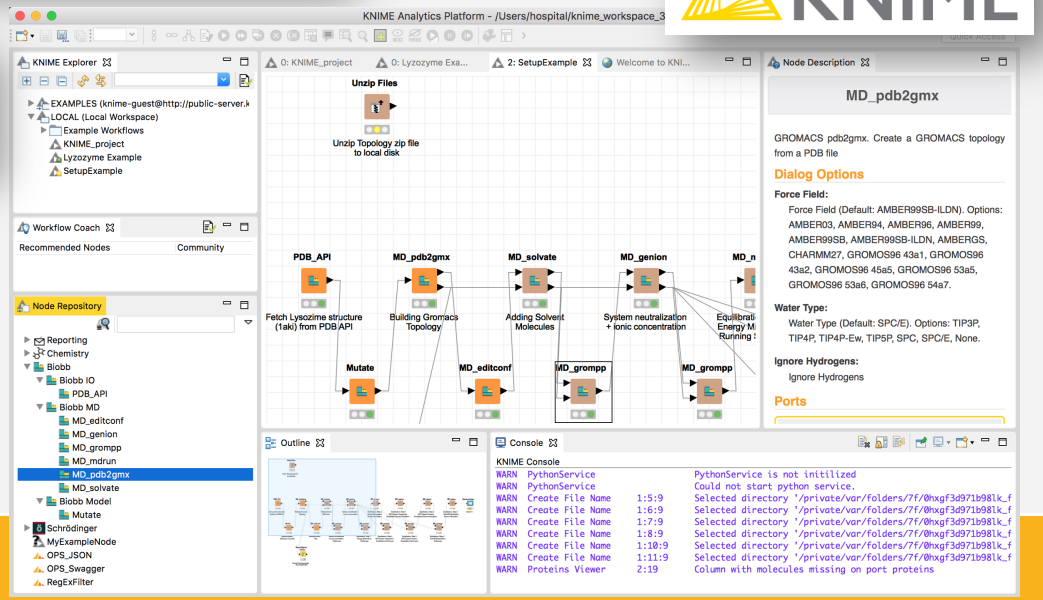
Protein MD Setup REST API

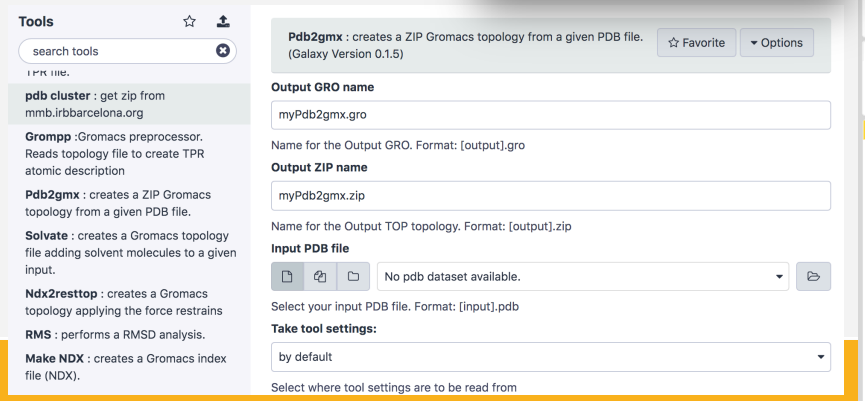
Command-line Workflows



Open for Innovation
KNIME

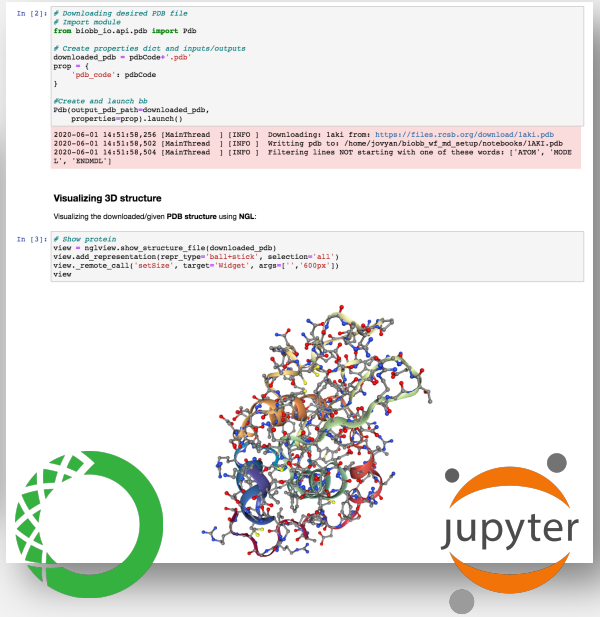
Galaxy
PROJECT





BioExcel Building Blocks: Versatility - Examples

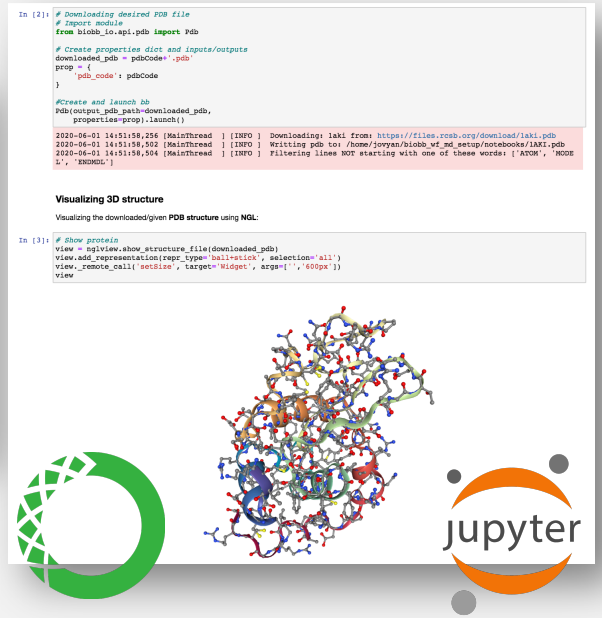
1) Demonstration Workflows (Jupyter Notebooks)



2) Pre-exascale Workflows (PyCOMPSs)

Biomolecular workflows using biobb

Demonstration Workflows



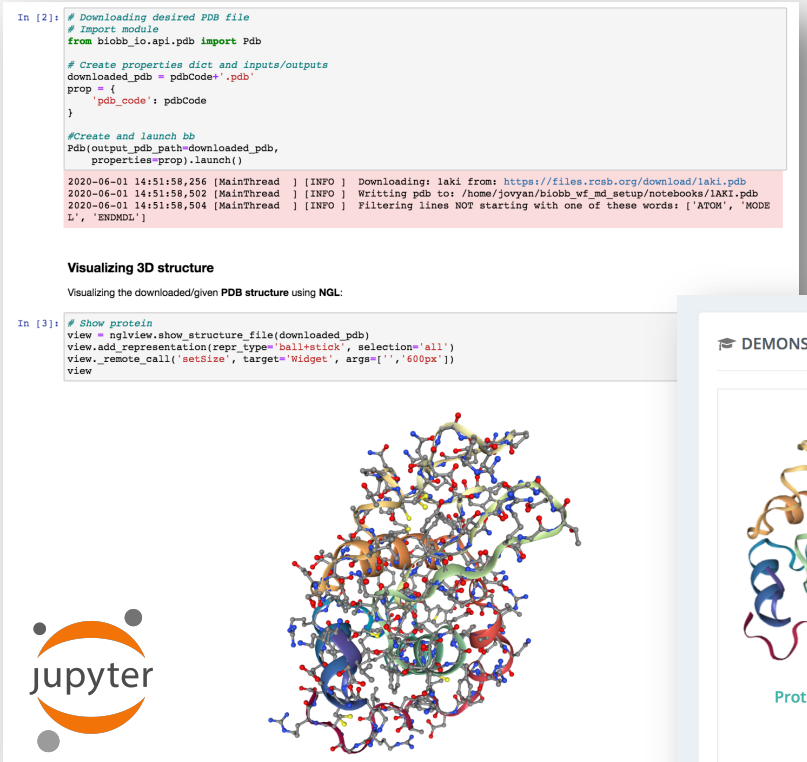
BioBB Workflows: Jupyter Notebooks

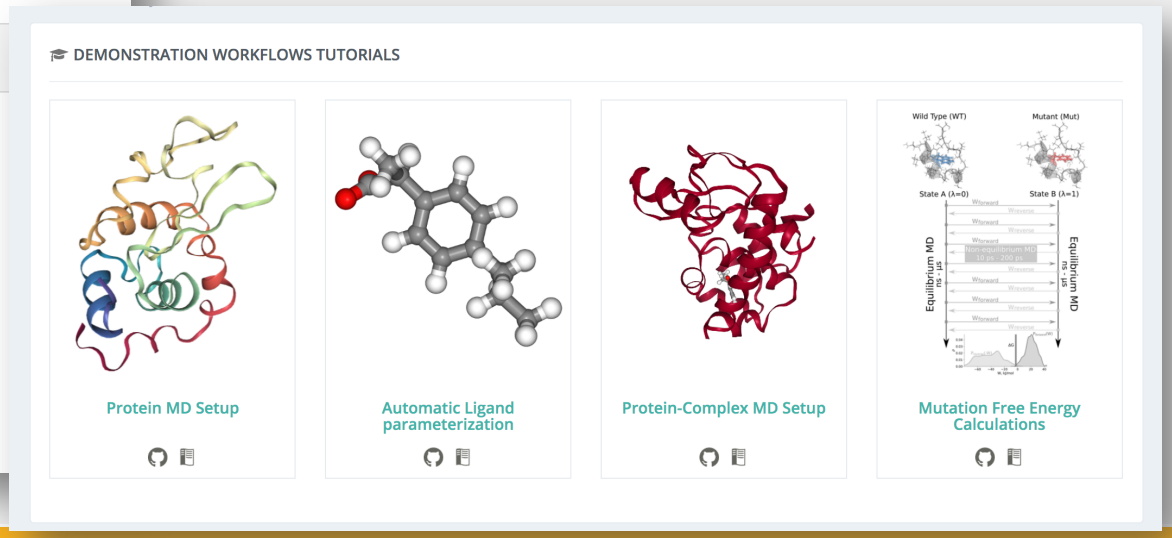
In general:

- Fantastic tool for **training**
- Inspect **Intermediate** results
- **Interactively** modify parameters
- Possibility to run it in **myBinder**

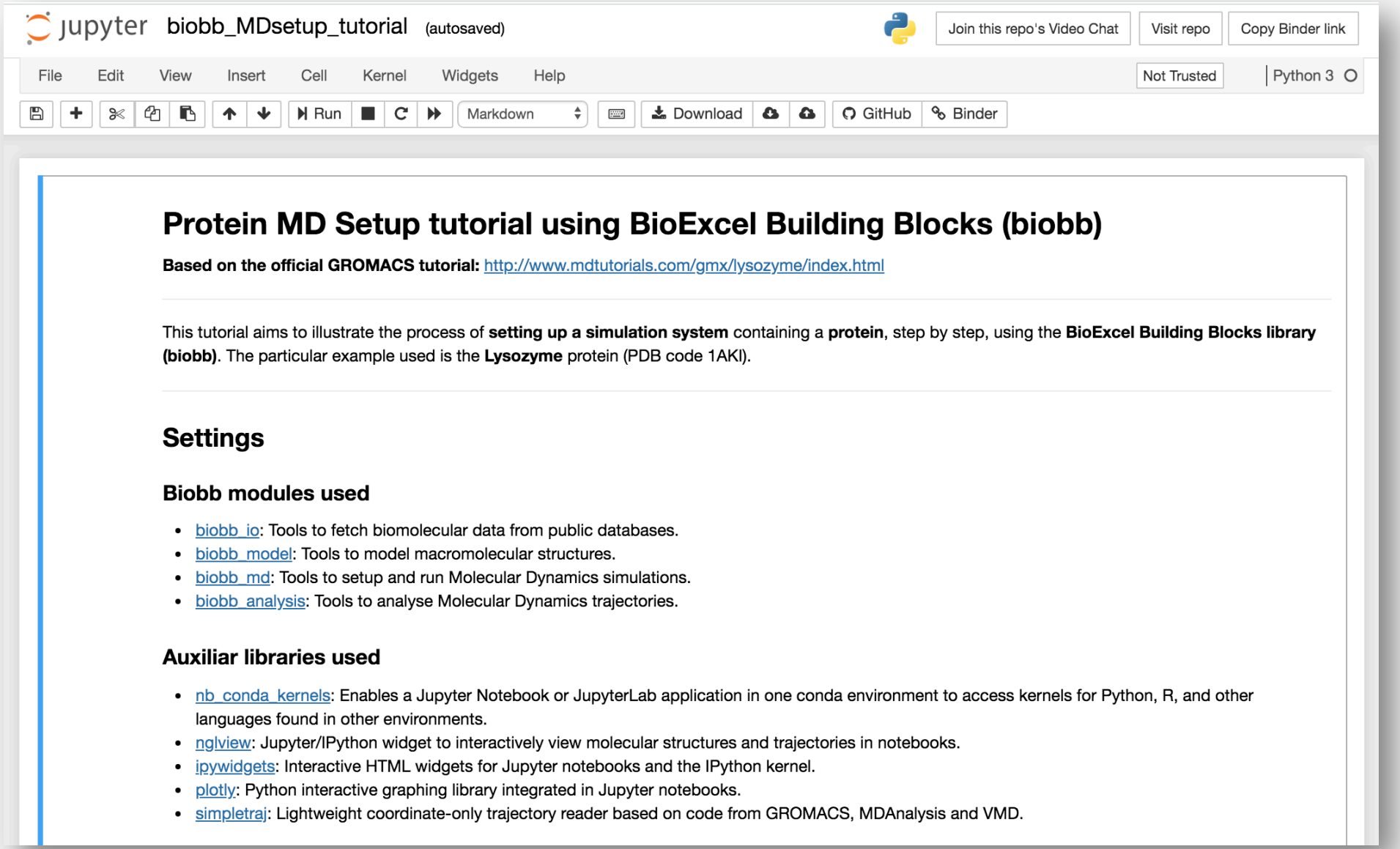
In particular (biobbs):

- Be familiar with **biobb syntax**
- Learn how to build workflows (**tutorials**)
- **Package** workflow (**Conda**)





BioBB Workflows: Tutorials



BioBB Workflows: Tutorials

Create protein system topology

Building GROMACS topology corresponding to the protein structure.
Force field used in this tutorial is amber99sb-ildn: AMBER **parm99** force field with **corrections on backbone** (sb) and **side-chain torsion potentials** (ildn).
Water molecules type used in this tutorial is spc/e.
Adding **hydrogen atoms** if missing. Automatically identifying **disulfide bridges**.

Generating two output files:

- **GROMACS structure** (gro file)
- **GROMACS topology** ZIP compressed file containing:
  - *GROMACS topology top file* (top file)
  - *GROMACS position restraint file/s* (itp file/s)

Building Blocks used:

- Pdb2gmx from **biobb_md.gromacs.pdb2gmx**

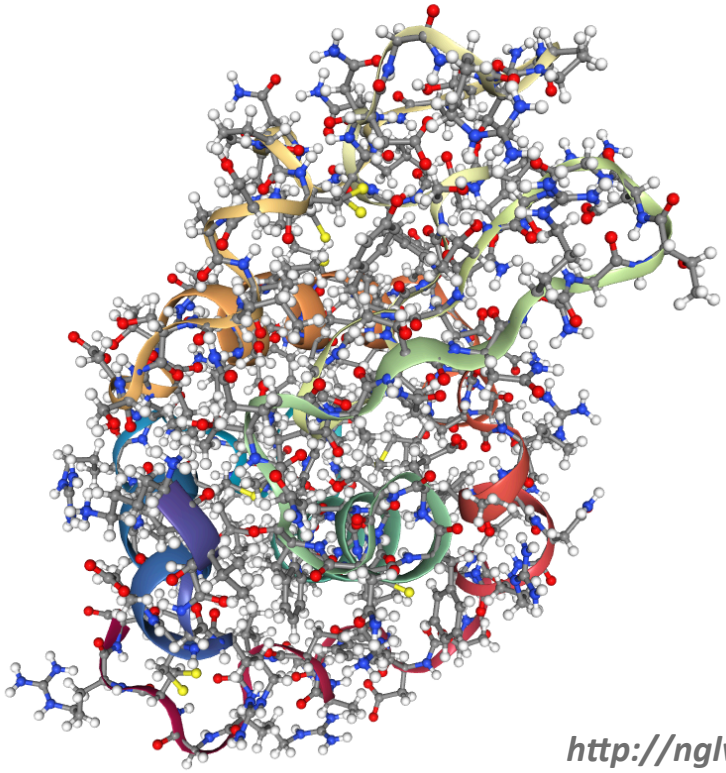
In [14]:
```
# Create system topology
# Import module
from biobb_md.gromacs.pdb2gmx import Pdb2gmx

# Create inputs/outputs
output_pdb2gmx_gro = pdbCode+'_pdb2gmx.gro'
output_pdb2gmx_top_zip = pdbCode+'_pdb2gmx_top.zip'

# Create and launch bb
Pdb2gmx(input_pdb_path=fixed_pdb,
        output_gro_path=output_pdb2gmx_gro,
        output_top_zip_path=output_pdb2gmx_top_zip).launch()
```

```
2019-12-03 15:49:16,485 [MainThread  ] [INFO ]  GROMACS Pdb2gmx 20191 version detected
2019-12-03 15:49:16,491 [MainThread  ] [INFO ]  Not using any container
2019-12-03 15:49:17,750 [MainThread  ] [INFO ]  gmx -nobackup -nocopyright pdb2gmx -f 1AKI_fixed.pdb -o 1AKI_pdb2gmx.
gro -p p2g.top -water spce -ff amber99sb-ildn -i posre.itp

2019-12-03 15:49:17,756 [MainThread  ] [INFO ]  Exit code 0

2019-12-03 15:49:17,759 [MainThread  ] [INFO ]
Using the Amber99sb-ildn force field in directory amber99sb-ildn.ff

going to rename amber99sb-ildn.ff/aminoacids.r2b
going to rename amber99sb-ildn.ff/dna.r2b
going to rename amber99sb-ildn.ff/rna.r2b
Reading 1AKI_fixed.pdb...
Read '', 1001 atoms
Analyzing pdb file
Splitting chemical chains based on TER records or chain id changing.
There are 1 chains and 0 blocks of water and 129 residues with 1001 atoms

  chain  #res #atoms
```

Visualizing 3D structure

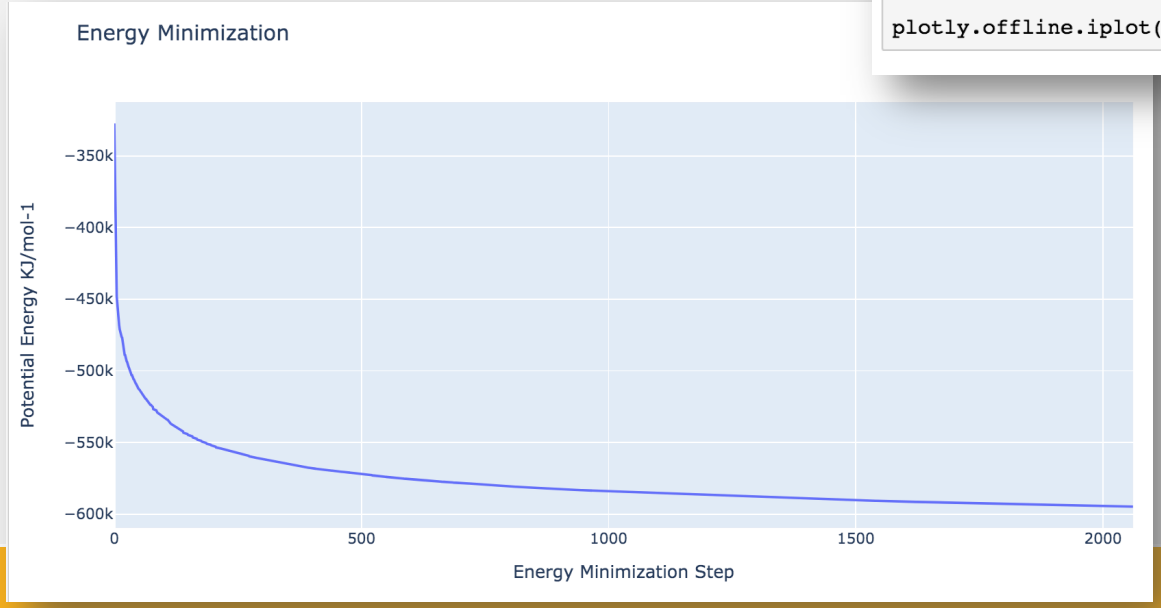
Visualizing the generated **GRO structure** using **NGL**. Note that **hydrogen atoms** were added to the structure by the **pdb2gmx GROMACS tool** when generating the **topology**.

In [12]:
```
# Show protein
struct_file = nglview.FileStructure(output_pdb2gmx_gro)
view = nglview.show_file(struct_file)
view.add_representation(repr_type='ball+stick', selection='all')
view._remote_call('setSize', target='Widget', args=['','600px'])
view
```

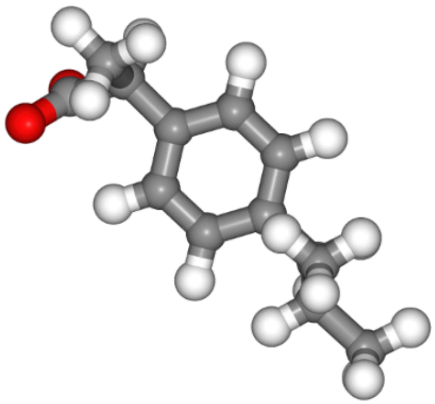


http://nglviewer.org/nglview/latest/

```
import plotly
import plotly.graph_objs as go

#Read data from file and filter energy values higher than 1000 Kj/mol^-1
with open(output_min_ene_xvg,'r') as energy_file:
    x,y = map(
        list,
        zip(*[
            (float(line.split()[0]),float(line.split()[1]))
            for line in energy_file
            if not line.startswith(("#","@"))
            if float(line.split()[1]) < 1000
        ])
    )

plotly.offline.init_notebook_mode(connected=True)

fig = {
    "data": [go.Scatter(x=x, y=y)],
    "layout": go.Layout(title="Energy Minimization",
                        xaxis=dict(title = "Energy Minimization Step"),
                        yaxis=dict(title = "Potential Energy KJ/mol-1")
                       )
}

plotly.offline.iplot(fig)
```



BioBB Workflows: Conda Environments



Automatic Ligand parameterization

Branch: master ▾ **biobb_wf_ligand_parameterization** / conda_env / **environment.yml**

gbayarri Removing python as a environment dependency

1 contributor

11 lines (11 sloc) | 211 Bytes

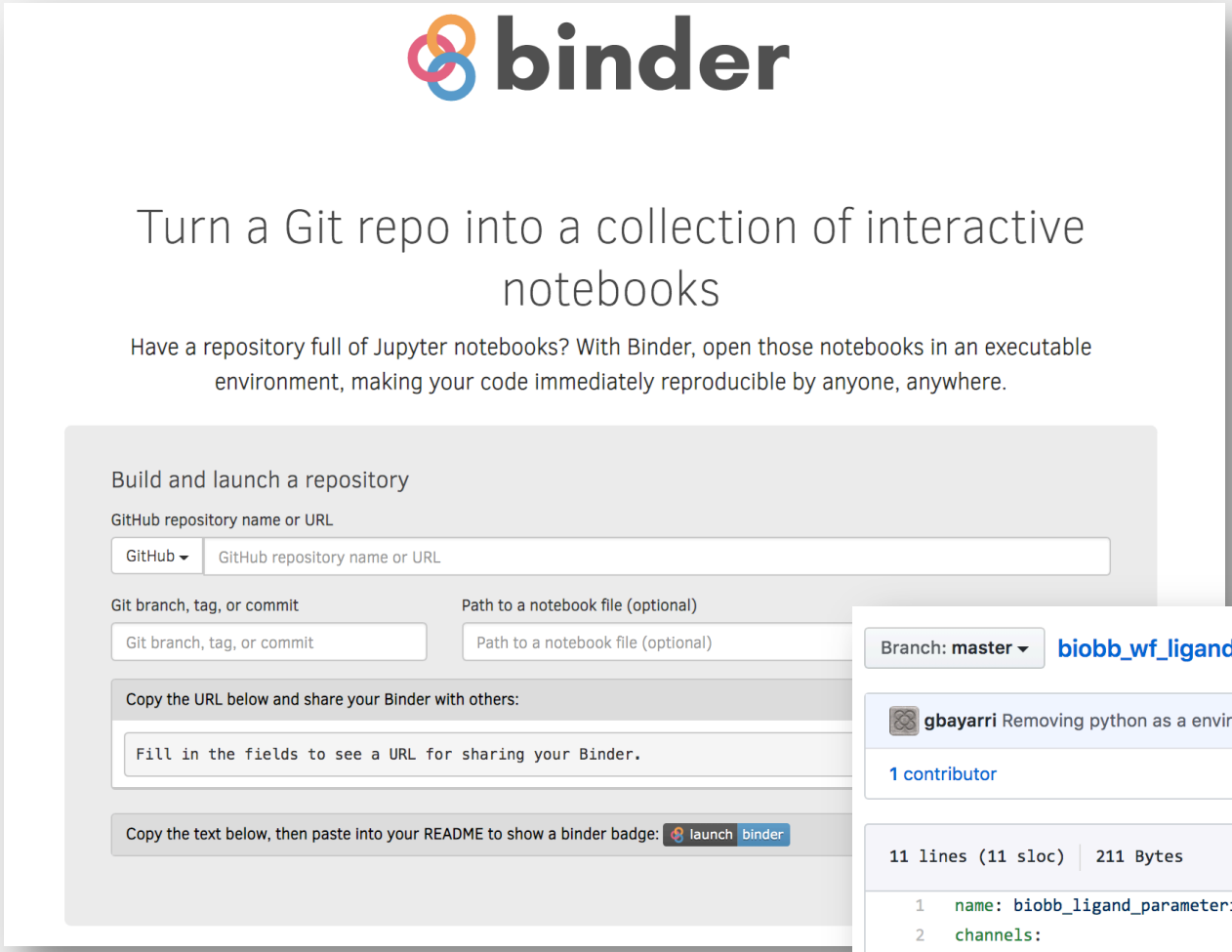
```yaml
name: biobb_ligand_parameterization_tutorial
channels:
  - conda-forge
  - bioconda
dependencies:
  - biobb_common==3.0.0
  - biobb_io==3.0.0
  - biobb_chemistry==3.0.1
  - nb_conda_kernels
  - nglview
  - conda
```

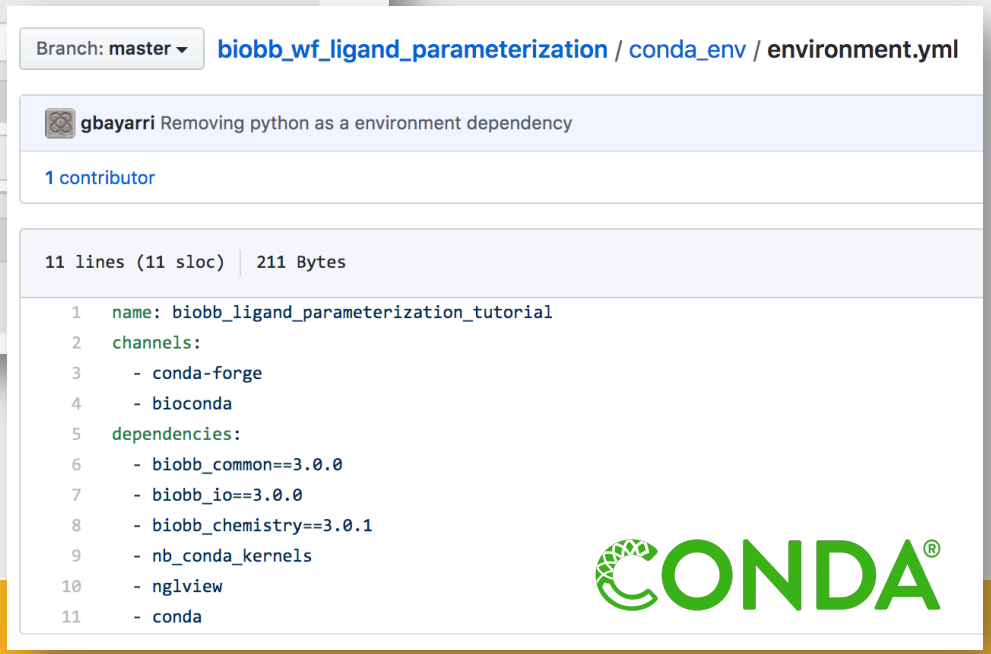
CONDA®

Conda Installation and Launch

```
git clone https://github.com/bioexcel/biobb_wf_ligand_parameterization.git
cd biobb_wf_ligand_parameterization
conda env create -f conda_env/environment.yml
conda activate biobb_ligand_parameterization_tutorial
jupyter-nbextension enable --py --user widgetsnbextension
jupyter-nbextension enable --py --user nglview
jupyter-notebook biobb_wf_ligand_parameterization/notebooks/biobb_ligand_parameterization_tutorial.ipynb
```

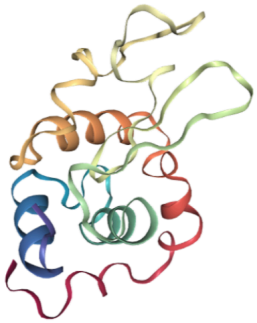
BioBB Workflows: myBinder



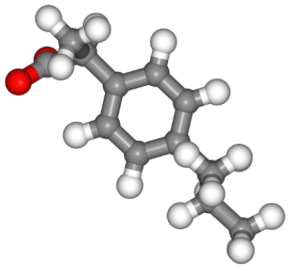


BioBB Workflows: Demonstration workflows

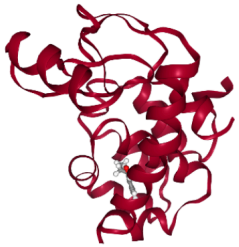
DEMONSTRATION WORKFLOWS TUTORIALS



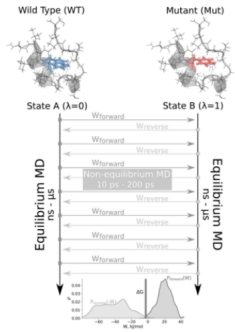
Protein MD Setup



Automatic Ligand parameterization

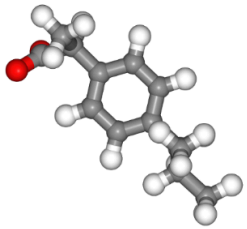


Protein-Complex MD Setup



Mutation Free Energy Calculations

AUTOMATIC LIGAND PARAMETERIZATION



This tutorial aims to illustrate the process of ligand parameterization for a small molecule, step by step, using the BioExcel Building Blocks library (biobb). The particular example used is the Ibuprofen small compound (3-letter code IBP, Drugbank code DB01050), a non-steroidal anti-inflammatory drug (NSAID) derived from propionic acid and it is considered the first of the propionics.

Execute in binder | View tutorial | Open Github repository | Open documentation

Biomolecular workflows using biobb

HPC Workflows

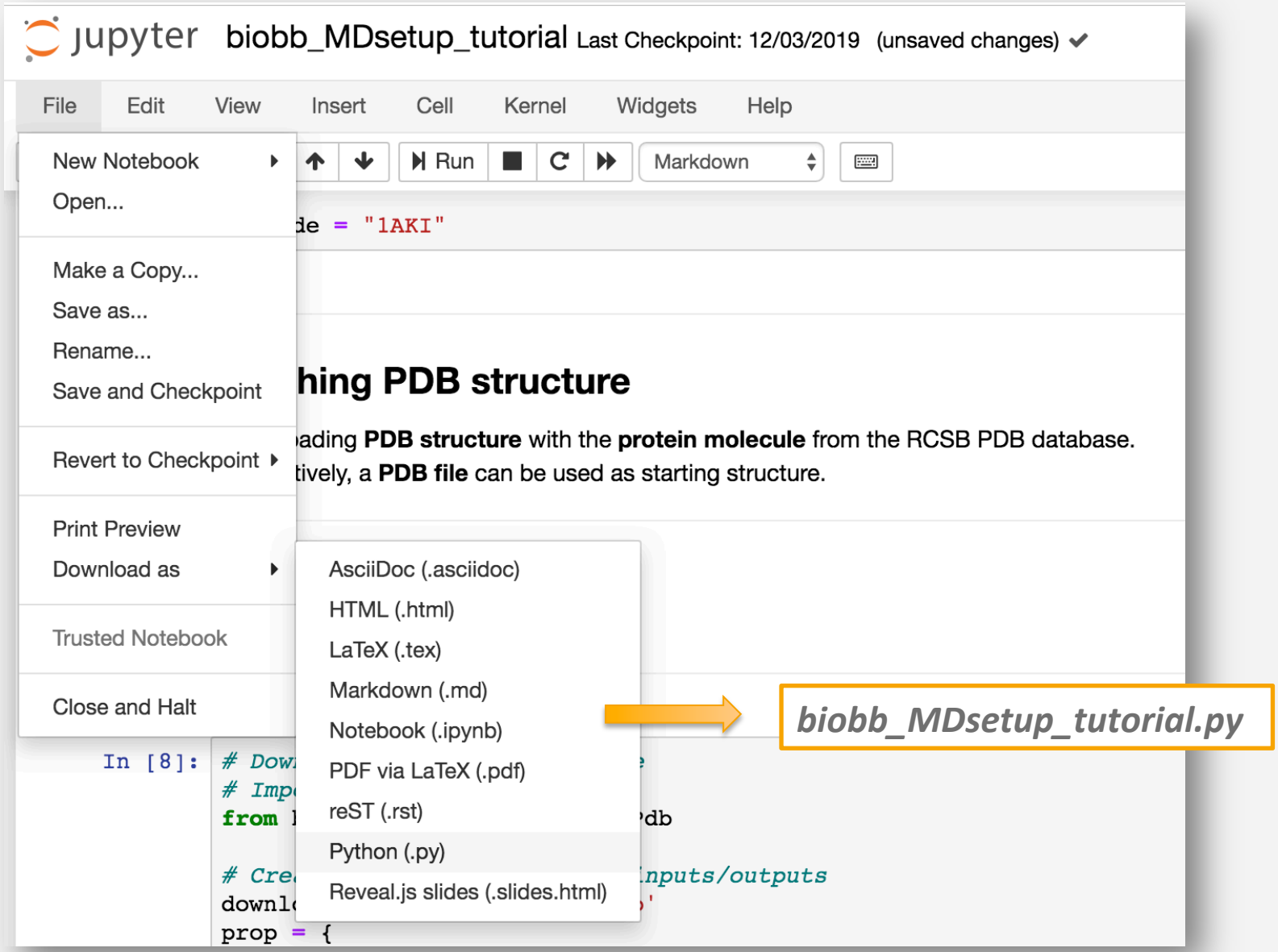
Enabling scalability

Making biomolecular simulation workflows exascale ready

pyCOMPSs

200 * 192 cores(4 nodes)
38,400 cores

BioBB Workflows: CLI – from Jupyter Notebook



BioBB Workflows: CLI – from Jupyter Notebook

```
(biobb_MDsetup_tutorial) Adams-MacBook-Pro:Python hospital$ python biobb_MDsetup_tutorial.py
/anaconda3/envs/biobb_MDsetup_tutorial/lib/python3.6/site-packages/biobb_common/tools/file_utils.py:327: UserWarning: Warni
ng: filter is not a recognized property. The most similar property is: filt
  error_property, close_property))
2019-12-11 08:26:44,183 [MainThread  ] [INFO ]  Downloading: 1aki from: https://files.rcsb.org/download/1aki.pdb
2019-12-11 08:26:44,891 [MainThread  ] [INFO ]  Writting pdb to: /Users/hospital/biobb_tutorials/biobb_wf_md_setup/biobb_wf
_md_setup/Python/1AKI.pdb
2019-12-11 08:26:44,891 [MainThread  ] [INFO ]  Filtering lines NOT starting with one of these words: ['ATOM']
2019-12-11 08:26:45,386 [MainThread  ] [INFO ]  check_structure -i 1AKI.pdb -o 1AKI_fixed.pdb --force_save fixside --fix AL
L

2019-12-11 08:26:45,387 [MainThread  ] [INFO ]  Exit code 0
```

Disadvantages:

- Graphical cells not showing.
- Losing interactivity

Advantages:

- Gaining High Throughput (automation, repetition)

Problem:

- Modify parameters for a certain step

  → modify the Python script

BioBB Workflows: CLI - from Python+Yaml

Workflow script

- Building blocks
- Python code
- Loops / conditionals
- Global log
- Output folders hierarchy

Workflow parameters

- Steps Inputs / Outputs
- Steps Dependencies
- Steps Properties

- Workflow inputs
- Workflow parameters

BioBB Workflows: CLI - from Python+Yaml

```python
global_log.info("step1_pdb: Download the initial Structure")
Pdb(**global_paths["step1_pdb"], properties=global_prop["step1_pdb"]).launch()

global_log.info("step2_fixsidechain: Modeling the missing heavy atoms in the structure side chains")
FixSideChain(**global_paths["step2_fixsidechain"], properties=global_prop["step2_fixsidechain"]).launch()

global_log.info("step3_pdb2gmx: Generate the topology")
Pdb2gmx(**global_paths["step3_pdb2gmx"], properties=global_prop["step3_pdb2gmx"]).launch()

global_log.info("step4_editconf: Create the solvent box")
Editconf(**global_paths["step4_editconf"], properties=global_prop["step4_editconf"]).launch()

global_log.info("step5_solvate: Fill the solvent box with water molecules")
Solvate(**global_paths["step5_solvate"], properties=global_prop["step5_solvate"]).launch()

global_log.info("step6_grompp_genion: Preprocess ion generation")
Grompp(**global_paths["step6_grompp_genion"], properties=global_prop["step6_grompp_genion"]).launch

global_log.info("step7_genion: Ion generation")
Genion(**global_paths["step7_genion"], properties=global_prop["step7_genion"]).launch()

global_log.info("step8_grompp_min: Preprocess energy minimization")
Grompp(**global_paths["step8_grompp_min"], properties=global_prop["step8_grompp_min"]).launch()

global_log.info("step9_mdrun_min: Execute energy minimization")
Mdrun(**global_paths["step9_mdrun_min"], properties=global_prop["step9_mdrun_min"]).launch()

global_log.info("step10_energy_min: Compute potential energy during minimization")
GMXEnergy(**global_paths["step10_energy_min"], properties=global_prop["step10_energy_min"]).launch(
```

Workflow script

Workflow parameters

```yaml
# Example of a YAML configuration file for a BioExcel building blocks workflow

working_dir_path: md_tutorial     # Folder to write i/o files of the workflow steps
can_write_console_log: False      # Verbose writing of log information
restart: False                     # Skip steps already performed
remove_tmp: True

step1_pdb:
  paths:
    output_pdb_path: structure.pdb
  properties:
    pdb_code: 1aki

step2_fixsidechain:
  paths:
    input_pdb_path: dependency/step1_pdb/output_pdb_path
    output_pdb_path: fixsidechain.pdb

step3_pdb2gmx:
  paths:
    input_pdb_path: dependency/step2_fixsidechain/output_pdb_path
    output_gro_path: pdb2gmx.gro
    output_top_zip_path: pdb2gmx_top.zip

step4_editconf:
  paths:
    input_gro_path: dependency/step3_pdb2gmx/output_gro_path
    output_gro_path: editconf.gro

step5_solvate:
  paths:
    input_solute_gro_path: dependency/step4_editconf/output_gro_path
    output_gro_path: solvate.gro
    input_top_zip_path: dependency/step3_pdb2gmx/output_top_zip_path
    output_top_zip_path: solvate_top.zip
```

BioBB Workflows: CLI - Mutations

```python
# Loading the biobb configuration reader
conf = settings.ConfReader(sys.argv[1])

global_log, _ = fu.get_logs(path=conf.get_working_dir_path())
global_prop = conf.get_prop_dic(global_log=global_log)
global_paths = conf.get_paths_dic()

global_log.info("step1_pdb: Download the initial Structure")
Pdb(**global_paths["step1_pdb"], properties=global_prop["step1_pdb"]).launch()

global_log.info("step2_fixsidechain: Modeling the missing heavy atoms in the structure side chains")
FixSideChain(**global_paths["step2_fixsidechain"], properties=global_prop["step2_fixsidechain"]).launch()

for mutation in conf.properties['mutations_list']:

    mut_paths = conf.get_paths_dic(mutation)
    mut_prop = conf.get_prop_dic(mutation, global_log=global_log)

    mut_paths['step3_mutate']['input_pdb_path'] = global_paths['step2_fixsidechain']['output_pdb_path']

    global_log.info("step3_mutate: Modeling a particular residue mutation")
    Mutate(**mut_paths["step3_mutate"], properties=mut_prop["step3_mutate"]).launch()

    global_log.info("step4_pdb2gmx: Generate the topology")
    Pdb2gmx(**mut_paths["step4_pdb2gmx"], properties=mut_prop["step4_pdb2gmx"]).launch()
```

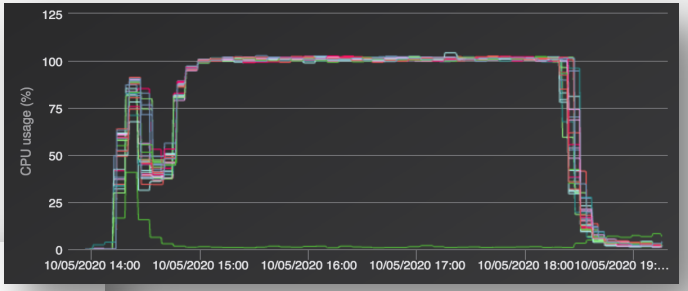

```yaml
mutations_list: ["A:Arg5Ala", "A:Arg5Gly", "A:Arg5Lys"]

step1_pdb:
  paths:
    output_pdb_path: structure.pdb
  properties:
    pdb_code: 1aki
```

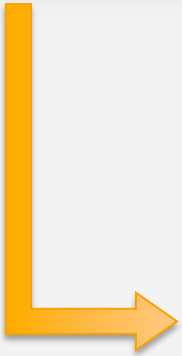
BioBB Workflows: CLI – Mutations

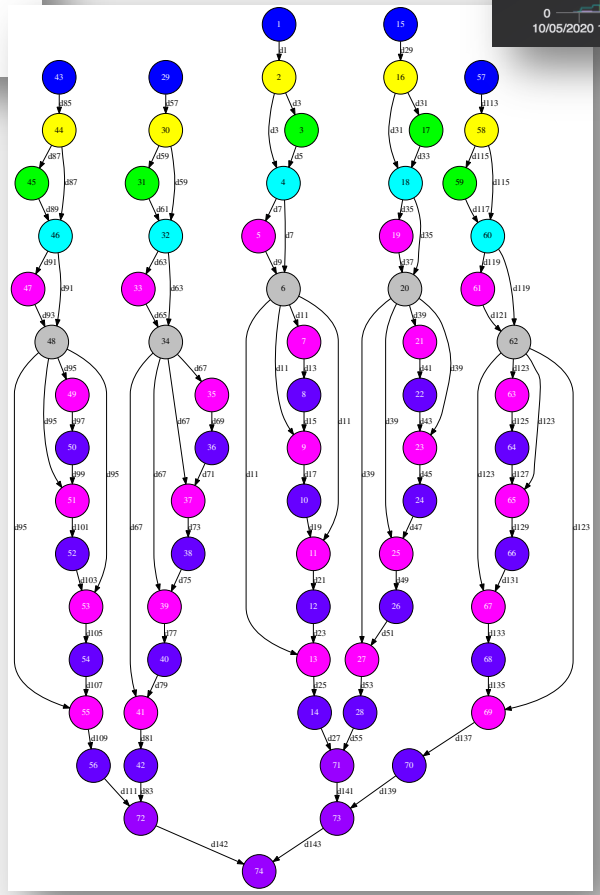
```
(biobb_Protein-Complex_MDsetup_tutorial) Adams-MacBook-Pro:Yaml hospital$ ls
biobb_MDsetup_tutorial-AlaScan.py   biobb_MDsetup_tutorial-lite.yaml    biobb_MDsetup_tutorial.py
biobb_MDsetup_tutorial-AlaScan.yaml biobb_MDsetup_tutorial-mut.py       biobb_MDsetup_tutorial.yaml
biobb_MDsetup_tutorial-lite.py      biobb_MDsetup_tutorial-mut.yaml     md_tutorial_mut
(biobb_Protein-Complex_MDsetup_tutorial) Adams-MacBook-Pro:Yaml hospital$ ls -lrht md_tutorial_mut/
total 64
-rw-r--r--   1 hospital  staff     0B Dec 11 22:43 log.err
drwxr-xr-x   5 hospital  staff   170B Dec 11 22:43 step2_fixsidechain
drwxr-xr-x   5 hospital  staff   170B Dec 11 22:43 step1_pdb
dr-xr-xr-x  24 hospital  staff   816B Dec 11 22:52 A:Arg5Ala
dr-xr-xr-x  24 hospital  staff   816B Dec 11 23:01 A:Arg5Gly
dr-xr-xr-x  24 hospital  staff   816B Dec 11 23:11 A:Arg5Lys
-rw-r--r--   1 hospital  staff    30K Dec 11 23:16 log.out
(biobb_Protein-Complex_MDsetup_tutorial) Adams-MacBook-Pro:Yaml hospital$ ls -rlht md_tutorial_mut/A\:Arg5Ala/
total 0
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:43 step5_editconf
dr-xr-xr-x  6 hospital  staff   204B Dec 11 22:43 step4_pdb2gmx
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:43 step3_mutate
dr-xr-xr-x  6 hospital  staff   204B Dec 11 22:43 step8_genion
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:43 step7_grompp_genion
dr-xr-xr-x  6 hospital  staff   204B Dec 11 22:43 step6_solvate
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:43 step9_grompp_min
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:45 step12_grompp_nvt
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:45 step11_energy_min
dr-xr-xr-x  8 hospital  staff   272B Dec 11 22:45 step10_mdrun_min
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:47 step14_energy_nvt
dr-xr-xr-x  9 hospital  staff   306B Dec 11 22:47 step13_mdrun_nvt
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:47 step15_grompp_npt
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:50 step18_grompp_md
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:50 step17_energy_npt
dr-xr-xr-x  9 hospital  staff   306B Dec 11 22:50 step16_mdrun_npt
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:52 step21_rmsexp
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:52 step20_rmsfirst
dr-xr-xr-x  9 hospital  staff   306B Dec 11 22:52 step19_mdrun_md
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:52 step24_dry
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:52 step23_image
dr-xr-xr-x  5 hospital  staff   170B Dec 11 22:52 step22_rgyr
```

BioBB Workflows + PyCOMPSs

```
print 'step2: mmbuniprot -- Get mutations'
mmbuniprot = uniprot.MmbVariants(prop['pdb_code'])
mutations = mmbuniprot.fetch_variants()

for mut in mutations:
    mut_path = cdir(wd, mut)

    print 'step3: scw -- Model mutation'
    scw_path = cdir(mut_path, 'step3_scw')
    scw_pdb = opj(scw_path, prop['mutated_pdb'])
    scw = scwrl.Scwrl4(mmbpdb_pdb, scw_pdb, mut, scwrl_path=scwrl_path)
    scw_pdb2 = scw.launchPyCOMPSs()
```

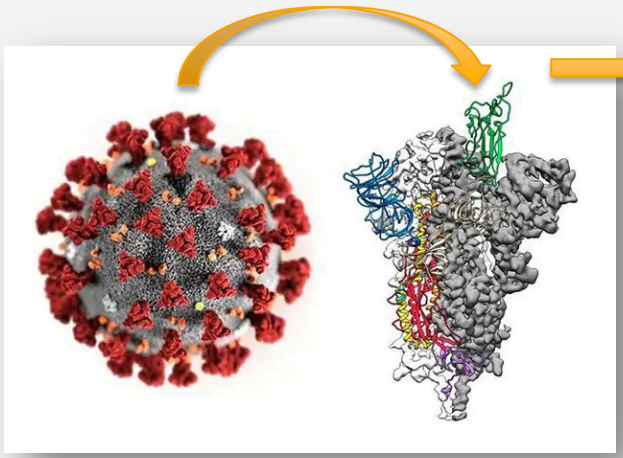


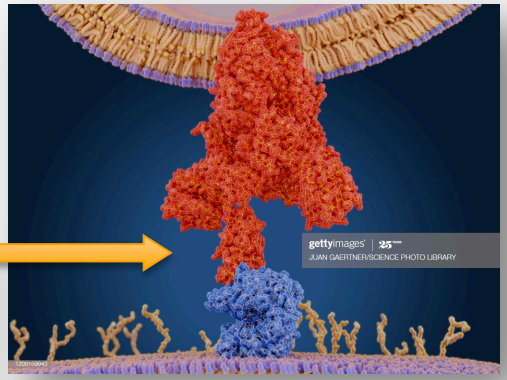


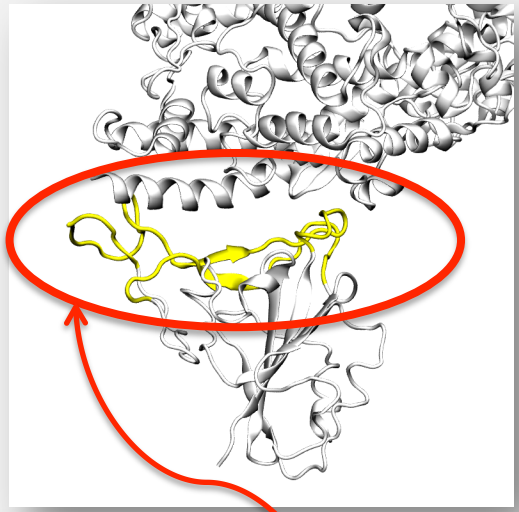


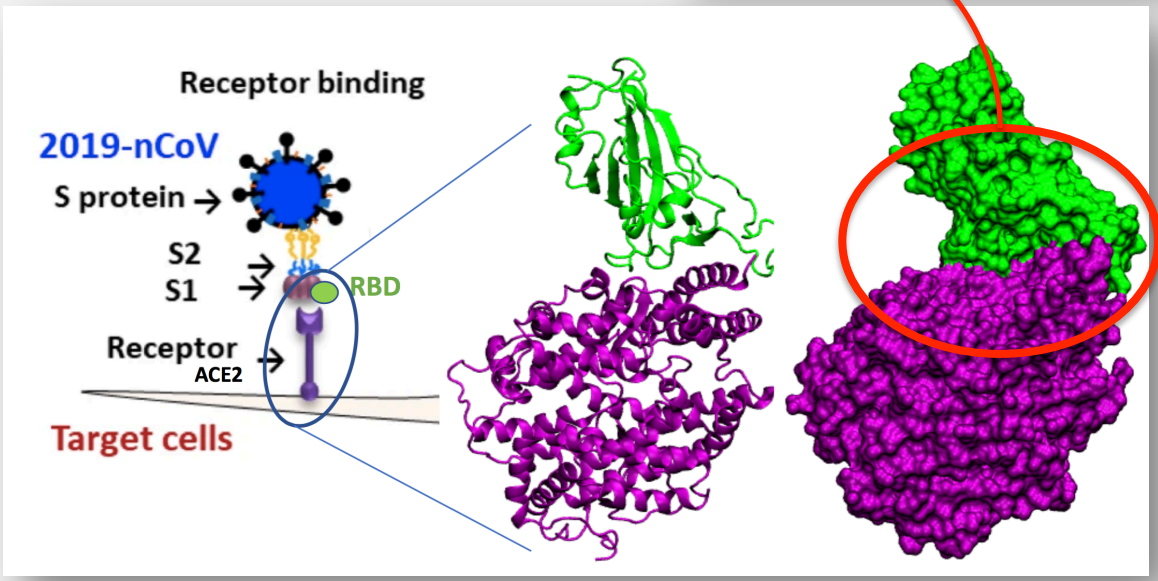
Example:
pre-exascale COVID-19 biobb workflow

COVID-19: SARS-CoV-2 RBD & hACE2



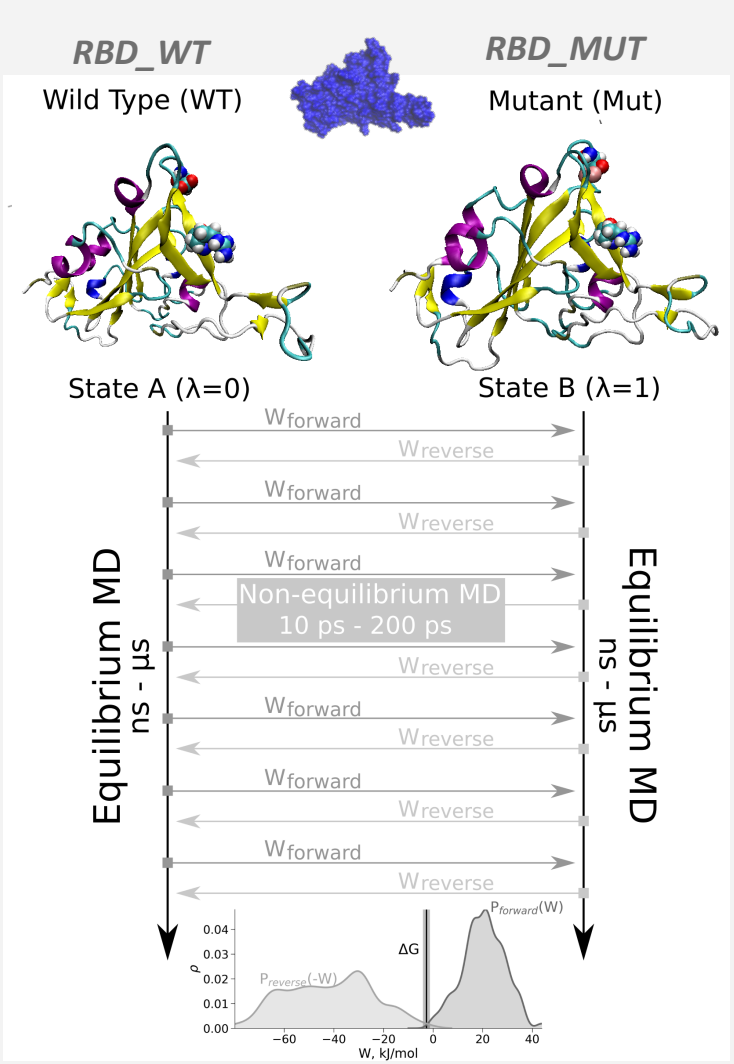


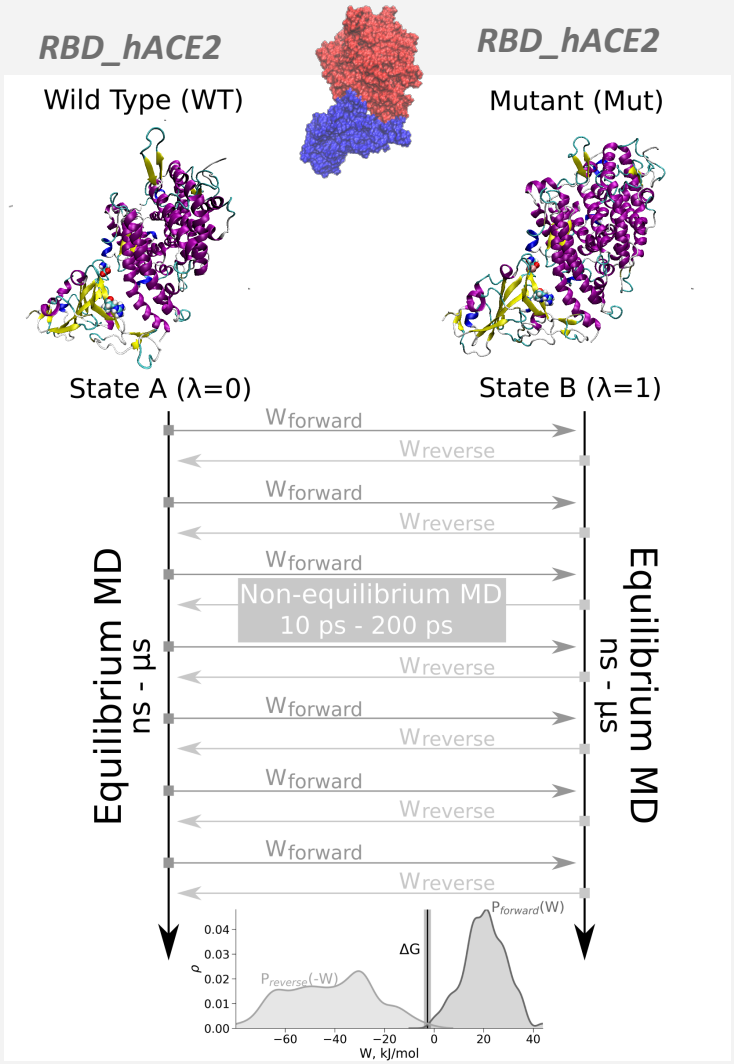




Alchemical free energy calculations of **relative ligand binding free energy** difference:
$\Delta\Delta G = \Delta G4 - \Delta G1$

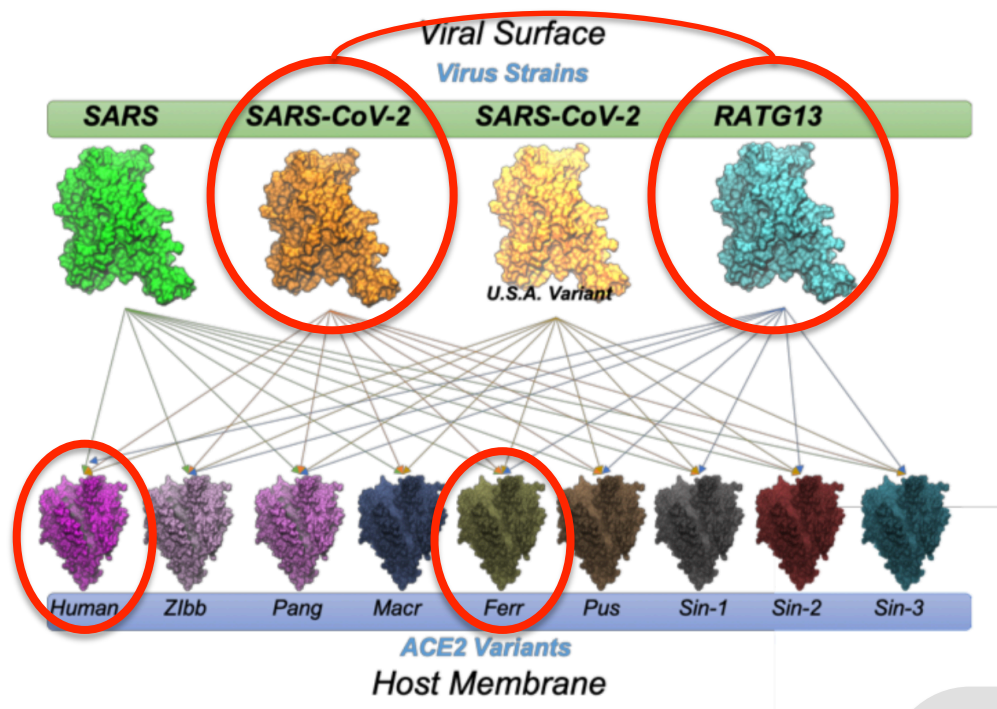
Alchemical free energy calculations
of relative protein binding free energy difference





ΔG1

ΔG4



PRACE

Big Data Study:
PRACE computational power
Massive amount of data

Molecular Dynamics simulation data

- For each mutation:
  - MD Simulations (RBD + ACE2 + Complex)
  - Free energy calculations

Impact of mutations in binding affinity

- **Fast-growth Thermodynamic Integration**
- **1000** independent **short MD simulations** (500 forward + 500 reverse)
- **GROMACS + pmx**
- **Extremely parallelizable**

GROMACS
FAST. FLEXIBLE. FREE.

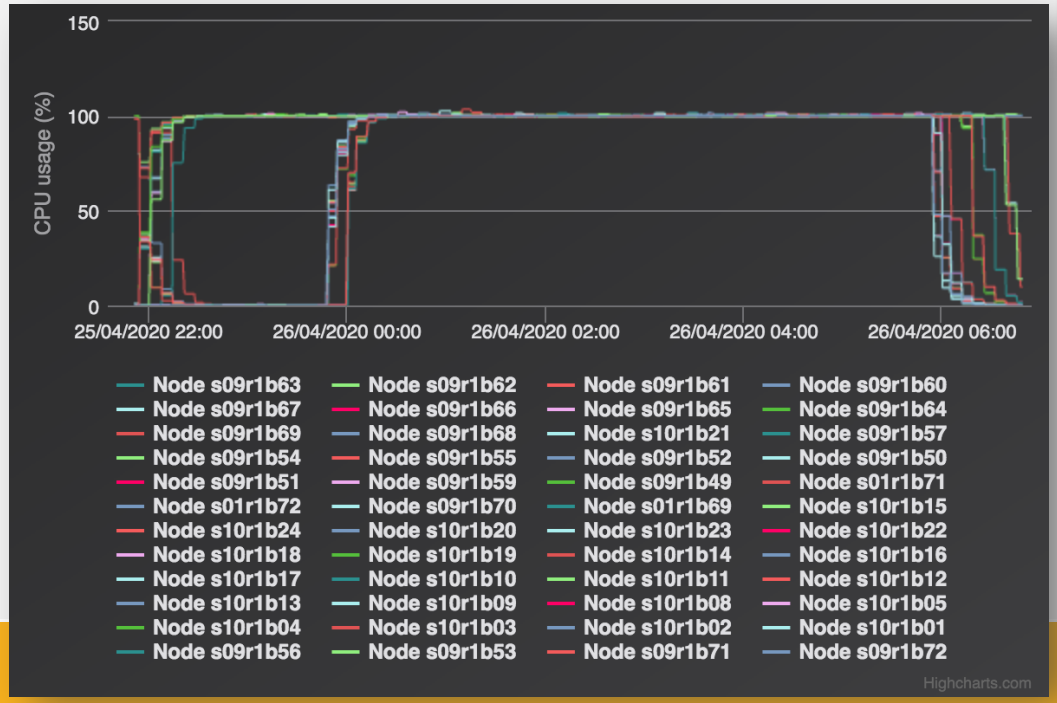
pmx: generate hybrid protein structure and topology
Computational Biomolecular Dynamics Group

Workflow 1: Equilibrium MD simulations

bsc@lg1:~/PMX> **python mdlaunchCV.py** -h

Workflow to **model**, **setup** and **run MD simulations** for a set of **mutations**.

Usage: **mdlaunchCV.py** [-h]
-m MUTATION -wt WT_STRUCTURE
[-q QUEUE] [-t TIME] [-nn NUM_NODES]
[-c CUMULATIVE] [-mpi MPI_NODES]
[-o OUTPUT_DIR] [-jn JOB_NAME] [-l MD_LENGTH]



python **mdlaunchCV.py**
-m WT+B:F756S
-c False
-wt complex.pdb
-l 10000
-nn 8
-mpi 4
-o COMPLEX_MUTS
-q bsc_ls
-t 2800
-jn mds_F756S_covid

48 MareNostrum nodes
2,304 cores → 1 job

12 mutations
10ns-length MDs
GROMACS 4 nodes MPI

Time: 8h

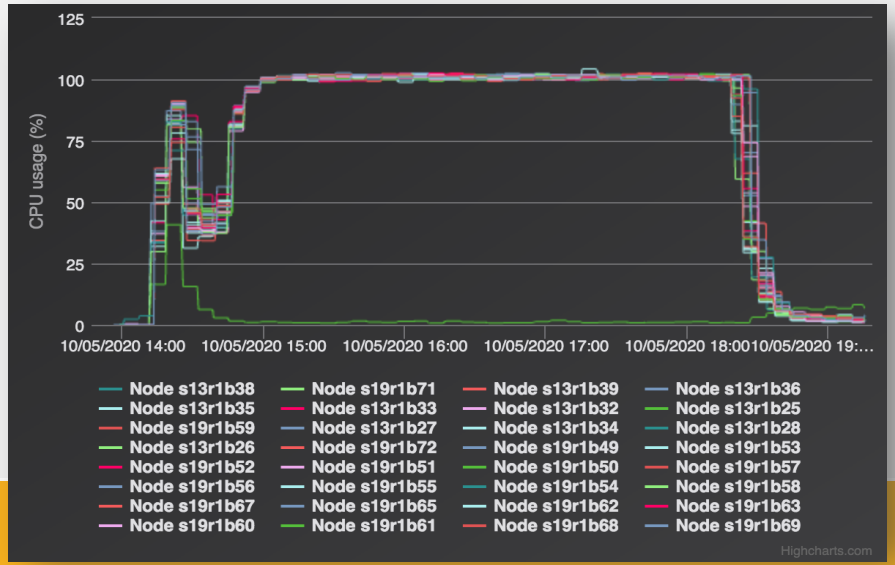
Workflow 2: Mutations free energy calculations

bsc@lg1:~/PMX> **python pmxlaunchCV.py** -h

Free Energy estimation upon amino acid modification with fast growth Thermodynamic Integration using GROMACS and pmx tools.

Usage: pmxlaunchCV.py [-h]
-m MUTATION
-wt_top WT_TOPOLOGY -wt_trj WT_TRAJECTORY
-mut_top MUT_TOPOLOGY -mut_trj MUT_TRAJECTORY
[-q QUEUE] [-t TIME] [-nn NUM_NODES] [-fe FE_LENGTH] [-nf NUM_FRAMES] [-prn PMX_RESNUM]

python **pmxlaunchCV.py**
-m Lys8Arg
-wt_top wt.tpr
-wt_trj wt.xtc
-mut_top mut.tpr
-mut_trj mut.xtc
-nf 500
-q bsc_ls
-jn aceK8R_covid
-o Lys8Arg
-nn 16
-t 2800



32 MareNostrum nodes
1,536 cores → 1 job

1 mutation (RBD-ACE2)
1000 short TI MDs (50ps)
500 forward
+
500 reverse

Time: 5h

BioExcel Building Blocks: Summary

- **BioExcel Building Blocks** software library offers a new layer of interoperability on biomolecular software tools.
- **Biomolecular workflows**, portable and reproducible, can be easily built using the library.
- The additional **workflow manager compatibility layer** allows **workflows** built with the library to be launched and controlled with **different frameworks** and in **different infrastructures**:
  - **Demonstration** workflows (Jupyter Notebooks)
  - **Pre-exascale HPC** workflows (PyCOMPSs)

Bonus Track!

biobb

Home | Availability | Build your own BioBB | Abo

BioExcel Building Blocks, a software librar

Home

BioExcel Building Blocks

Do you want to wrap up your tool with BioBB's?

First off, please read the official documentation available at Read the Docs.

Once you are ready, download from GitHub our biobb_template.

Use the biobb_template to develop your own BioBB.

Once you've finished and run the tests, make a pull request to our GitHub.

Our team will check your code and, if necessary, make the appropriate arrangements.

Once the request has been accepted, your BioBB is ready for launching.

Congratulations! Your tool is now part of BioExcel Building Blocks!

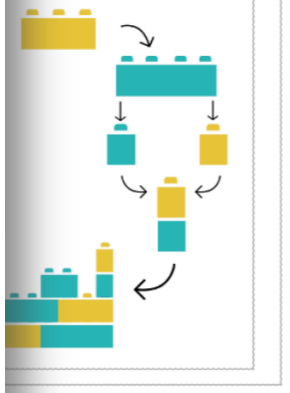
celona.org/biobb/

orkflows

WORKFLOW MANAGERS

Galaxy PROJECT

WORKFLOW



BioExcel Building Blocks: Training

Remote BioExcel Winter School on Biomolecular Simulations

November 30, 2020 Michelle Mendonca

VIRTUAL TRAININGS

Direct interaction with trainers!

Virtual Trainings are training events done in a virtual or simulated environment, with the learner and the instructor being in separate locations. They are designed to simulate the traditional classroom or learning experience. Similarly to the webinars series, **BioExcel Virtual Training series** for computational biomolecular research also cover a broad range of topics related to the biomolecular simulation field. The first **Virtual Trainings** for the PDBe Knowledge Base (PDBe-KB) and for the **BioExcel Building Blocks** have already been run, but new sessions are being planned: PDBe-KB, biobb

- **Training #1**, 09/12/2019 - 12/12/2019: biobb_VT - biobb_VT_wfs - biobb_VT_ex - biobb_VT_cli - examples

BioExcel Center of Excellence for Computational Biomolecular Research

858 subscribers

HOME VIDEOS PLAYLISTS CHANNELS DISCUSSION ABOUT

YouTube

2020 Remote BioExcel Summer School on Biomolecu

BioExcel Center of Excellence for Computational Bio... - 1 / 16

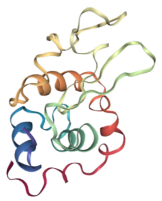
3 Basis of molecular dynamics simulations - Part I 23:54 BioExcel Center of Excellence for Compu...

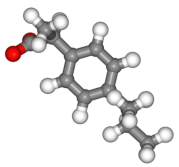
4 Basis of molecular dynamics simulations - Part 2 28:06 BioExcel Center of Excellence for Compu...

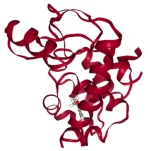
5 BioExcel Building Blocks - Part I 41:25 BioExcel Center of Excellence for Compu...

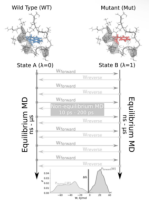
6 BioExcel Building Blocks - Part 2 33:30 BioExcel Center of Excellence for Compu...

DEMONSTRATION WORKFLOWS TUTORIALS


Protein MD Setup


Automatic Ligand parameterization


Protein-Complex MD Setup



Mutation Free Energy Calculations

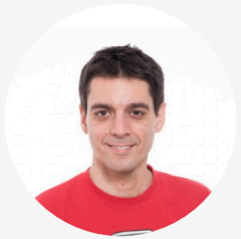
Acknowledgements

Main Developers

Pau Andrio

Software Research Engineer

BSC - CNS - Life Sciences, INB Computational Node 2



Genís Bayarri

Full Stack Developer

IRB Barcelona - Molecular Modeling and Bioinformatics

Laia Codó

Software Research Engineer

BSC - CNS - Life Sciences, INB Computational Node 2

Luis Jordà

Research Student

BSC - CNS - Life Sciences, INB Computational Node 2

Modesto Orozco

Group Leader

IRB Barcelona - Molecular Modeling and Bioinformatics

Josep Lluís Gelpí

INB - Computational Group Manager

BSC - CNS - Life Sciences, INB Computational Node 2

Jorge Ejarque

Position: RESEARCHER

Computer Sciences - Workflows and Distributed Computing

email: jorge.ejarque@bsc.es

Tel: +34 934137248

Access to ORCID Profile

Rosa M Badia

Position: WORKFLOWS AND DISTRIBUTED COMPUTING GROUP MANAGER

Computer Sciences - Workflows and Distributed Computing

email: rosa.m.badia@bsc.es

Tel: +34 934134075

Access to ORCID Profile

Questions

biobb

http://mmb.irbbarcelona.org/biobb/

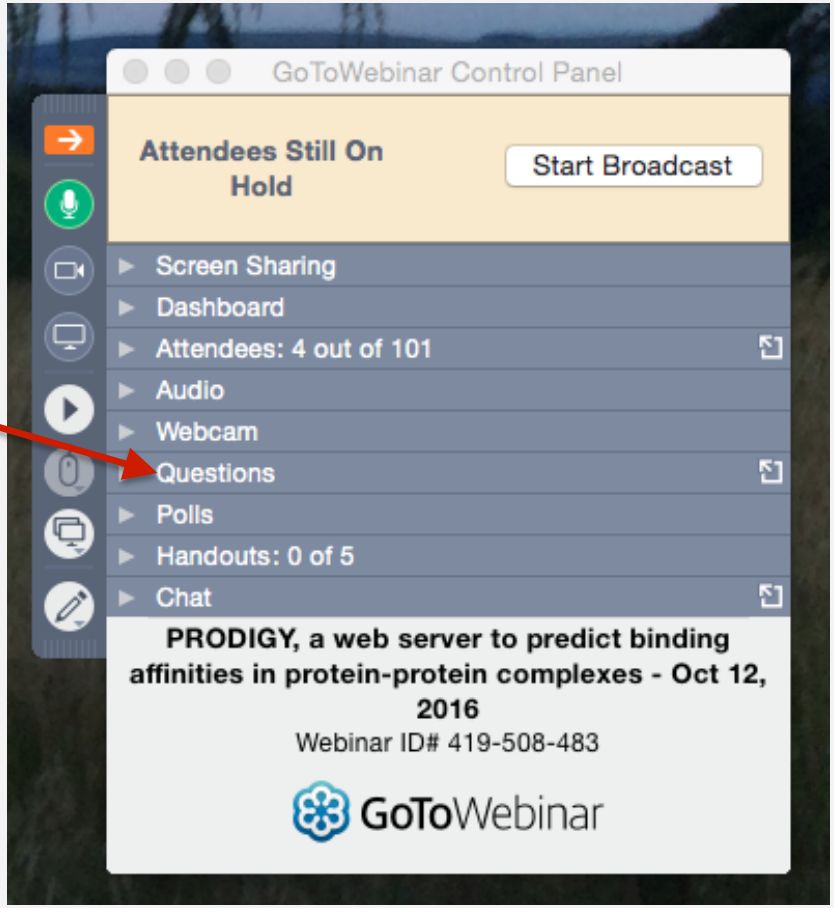
adam.hospital@irbbarcelona.org

Audience Q&A session

Please use the **Questions** function in **GoToWebinar** application

- If you *don't have audio*, please mention that in the question.

Any other questions or points to discuss after the live webinar? Join the discussions at **http://ask.bioexcel.eu**



Special edition Student Webinar: Summer School 2020 Edition (2020-09-15)

Molywood: streamlining the design and rendering of molecular movies

by Miłosz Wieczór (2020-10-22)

MOLYWOOD

See https://bioexcel.eu/