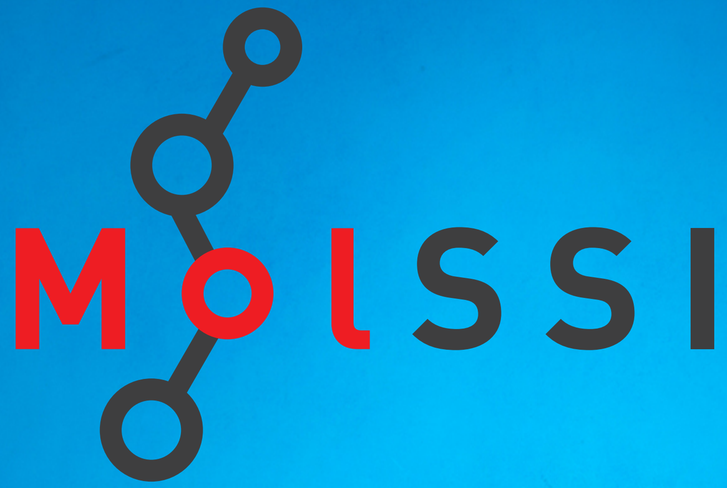
PsiCon 2018



Daniel G. A. Smith

The Molecular Sciences Software Institute

@dga_smith

molssi.org

MolSSI Education Initiatives

How do we change the software practices of an entire field?

- **Primary objectives:**
  - Teamwork and collaborative projects
  - MolSSI Best Practices
  - Open-source paradigms
  - Modern tools for modern software
  - Teach content to increase both a students scientific capability and industrial marketability
  - The "true" cost of ownership

Inspiration:

NUMFOCUS
OPEN CODE = BETTER SCIENCE

better scientific software

THE APACHE SOFTWARE FOUNDATION

open source initiative

- **Core Components:**
  - Software Carpentry style courses
  - Learning groups and hands on practice
  - Taught by experts in the field

software carpentry

Motivation

- Many terms thrown around:
  - **Best Practice**
  - **DevOps**
  - **Open Source**
  - **Software Sustainability**
  - **"Incorrect Software"**
  - **Manage Code**
  - **Build and Package**
  - **"Just get the physics working"**

"**Best**" is subjective and depends on your own community.

Lets discuss practical applications and how to make our lives easier!

Open Source

Open Source software is software that can be freely accessed, used, changed, and shared (in modified or unmodified form) by anyone. Open source software is made by many people, and distributed under licenses that comply with the Open Source Definition. – Open Source Initiative

- **Free Redistribution** – Program is free of charge and distributable
- **Source Code** – Source code is freely available
- **Derived Works** – Allows modification of source in derived works

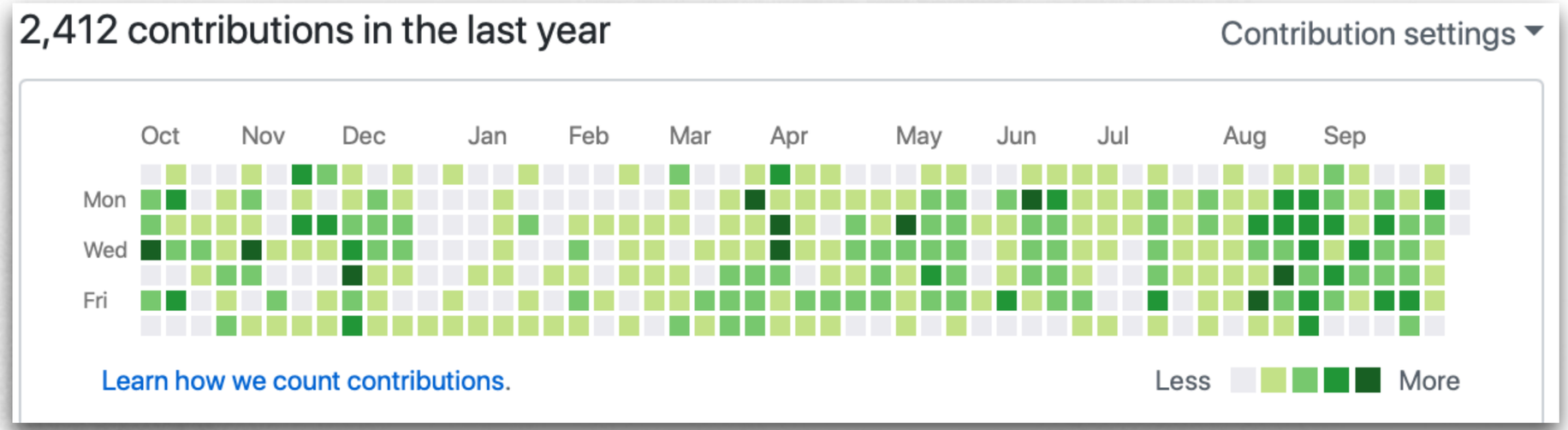
Open Source

- **Community** – Building engaged cooperative communities
- **Open Collaboration**– Anyone is free to join the community
- **Development Costs** – Allows common goal execution

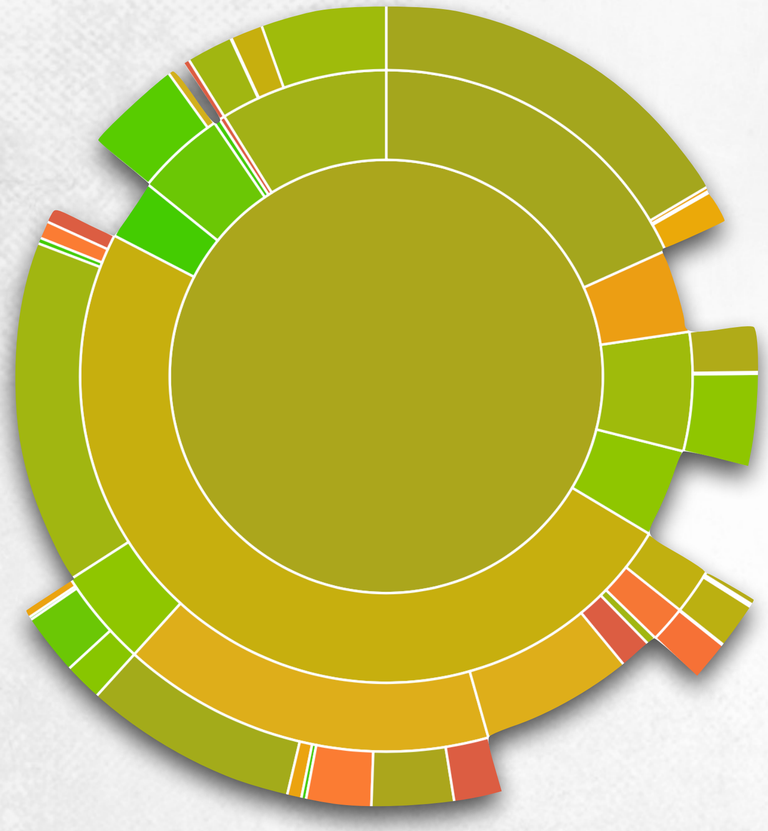
Open Source

Personal Reasons

- **Improves Coding Skills** - Code reviews and feedback
- **New Technologies**- Large base of users to provide unique experience
- **Peer Recognition** - Complete portfolio of what a person has done
- **Job Prospects** - Excellent for resumes



Code Coverage



codecov.io

- Tendency to write many tests that cover a single area while completely neglecting others.
- 100% code coverage does not imply perfect code!
- General metric that informs testing competency that is difficult to fake.

Code Formatters

- Automatic application of a standard format
  - Most language have a single standard (Python)
  - Several language have multiple standard (C++)
- Provides consistent code
  - Within your own software stack
  - And amongst other software stacks
- Tools: clang-format (C/C++), yapf (Python)
- Most languages (C, C++, Python, Java, Julia, etc)

example.py

```
x = {  'a':37,'b':42,

'c':927}

y = 'hello ''world'
z = 'hello '+'world'
a = 'hello {}'.format('world')
class foo  (     object  ):
  def f    (self   ):
    return        37*-+2
  def g(self, x,y=42):
      return y
def f  (   a ) :
  return      37+-+a[42-x :  y**3]

```

YAPF

Code Quality

lgtm.co

- Statically examine the code for a variety of errors and potential errors
- Catch bugs that do not have unit tests for them.
- Automation of code review:
  - Always catches “known” errors
  - Bad news is easier from a bot

Code Quality

Actual errors

```
2150        for (iop = 0; iop < nop; ++iop) {
2151            int axis, op_ndim, op_axis;
```

Local variable 'axis' hides a `parameter of the same name`.

Possible errors

```
1440        printf("| IterIndex: %d\n", (int)NIT_ITERINDEX(iter));
1441        printf("| Iterator SizeOf: %d\n",
1442                                (int)NIT_SIZEOF_ITERATOR(itflags, ndim, nop));
```

Multiplication result may overflow 'int' before it is converted to 'unsigned long'.

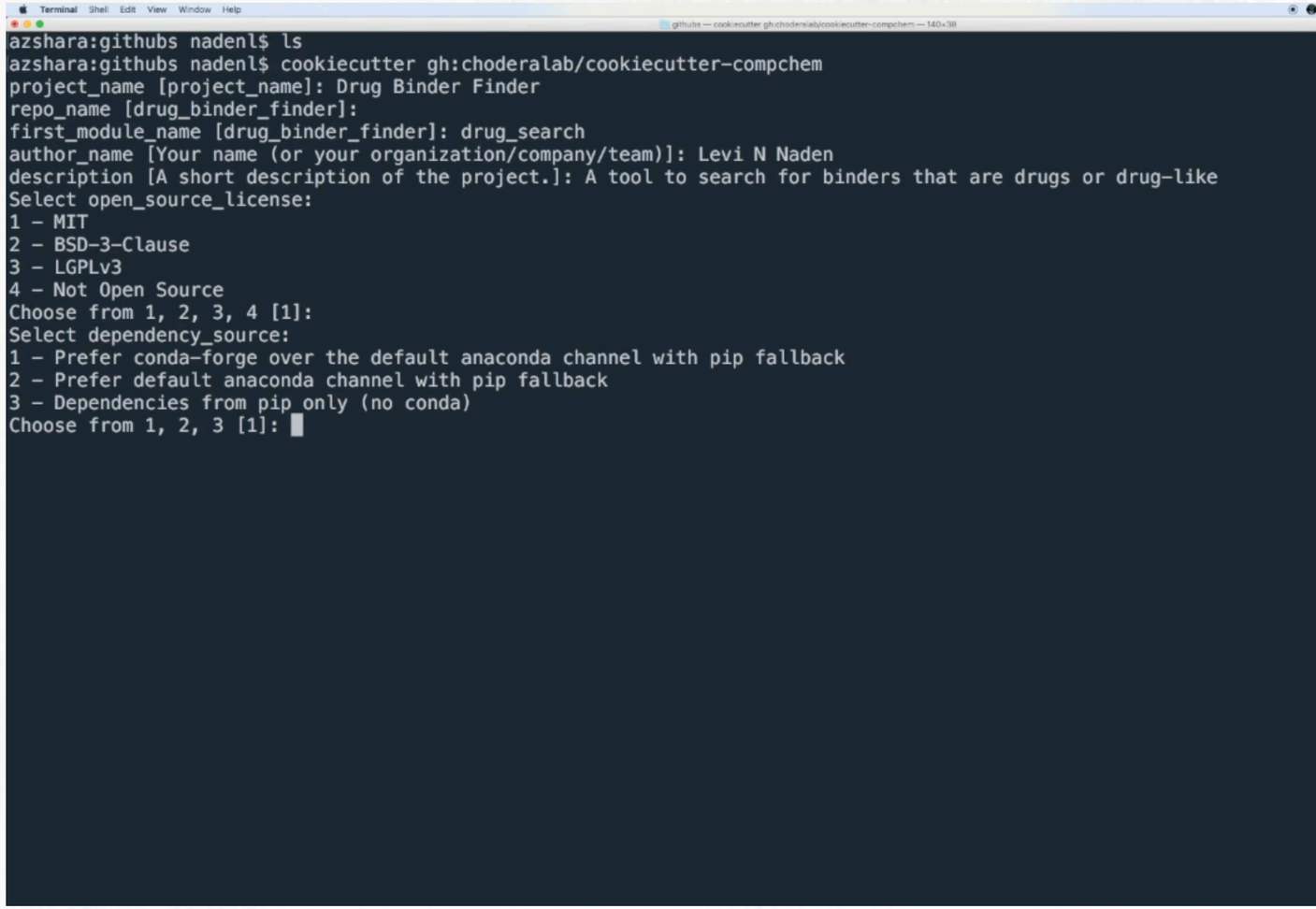
Potential code duplication

```
186     def list_final_energies(self, fragments=None, refresh_cache=False):
```

All statements in list_final_energies are similar in `list_final_molecules`.

Cookiecutters [Python]

- Automatic setup:
  - Version Control (git)
  - License (MIT/BSD/LGPL/Other)
  - CI (Travis CI/Appveyor)
  - Testing (PyTest)
  - Code Coverage (Codecov)
  - Documentation (Sphinx)
- Takes ~5 minutes to integrate



https://github.com/MolSSI/cookiecutter-cms

Final Thoughts

- Make libraries with command line interfaces
  - Think multi-stage runs (workflows)
- Try not to use **stdout**
- If you are writing CUDA you are probably doing it wrong
- Recommend YAML over custom ASCII files
- Recommend JSON/msgpack/YAML/HDF5 for cross platform serialization
- Use the cookiecutter

https://molssi.org/education/best-practices/

MolSSI Integral Reference Project

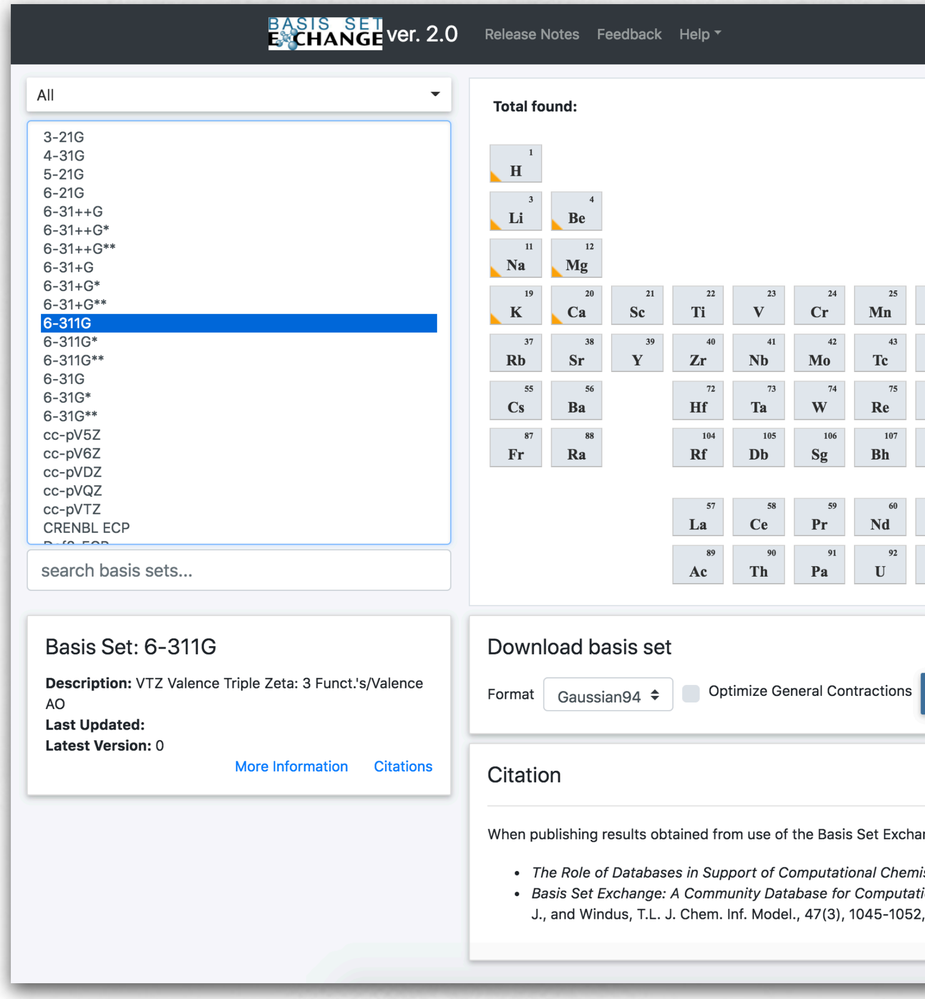
- Reference implementation and values
- Utilizes arbitrary-precision interval arithmetic (ball arithmetic)
- **Very slow**, but relatively simple implementation

```
4.78506540470550297026366517126315309034777632299183246390095
52057465005515845927490470528135254482526 +/- 4.63e-101
```

https://github.com/MolSSI/mirp

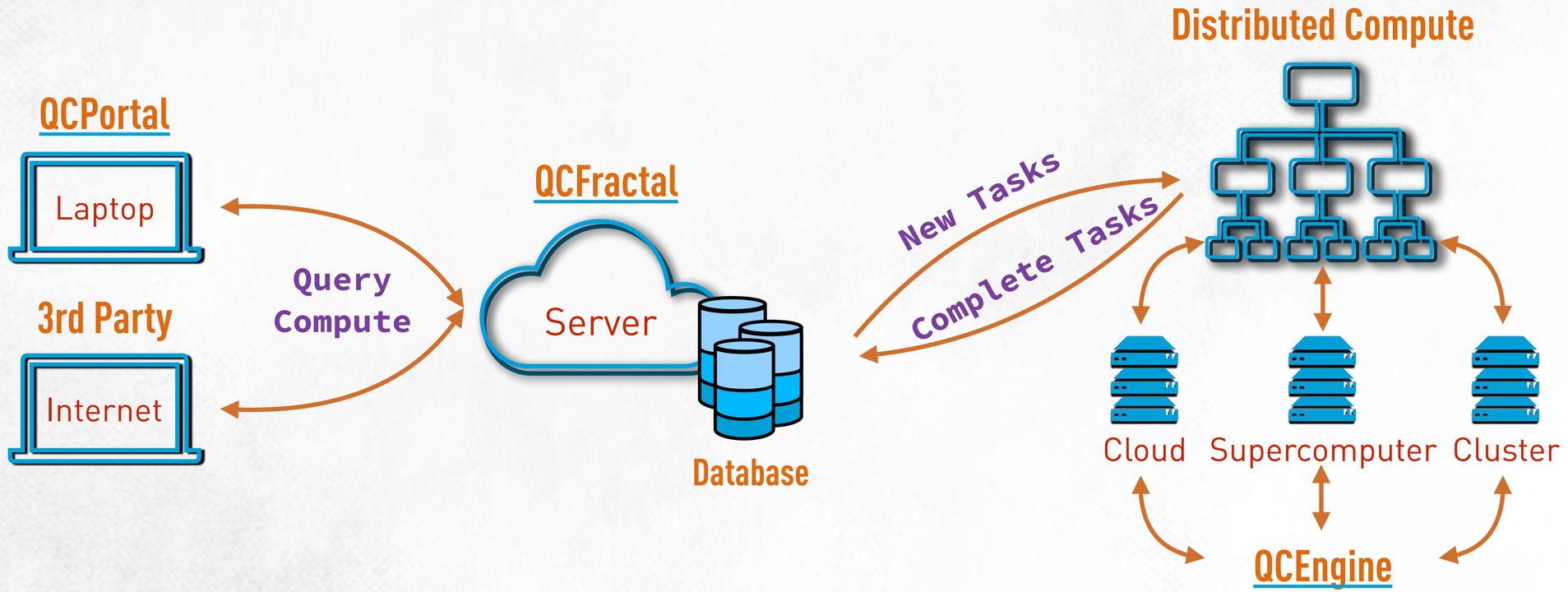
A New and Improved Basis-Set Exchange (BSE)

- Used by both end users and code developers
- Collaboration with original BSE developers (PNNL/EMSL)
- Improved provenance and reproducibility of calculations via unique identifiers
- Improved curation/reliability of the raw data
- Programmatic access via a public API



QCArchive Overview

Provide an open, community-wide quantum chemistry database to both facilitate and capture hundreds of millions of hours of computing time to enable large-scale forcefield construction, physical property prediction, new methodology assessment, and machine learning from data that would otherwise end up siloed or inaccessible.



QCSchema: github.com/MolSSI/QCSchema

Stack: QC Schema

https://github.com/MolSSI/QC_JSON_Schema

- Communication channel between all piece of the ecosystem.
- *Community* project useful for many aspects of quantum chemistry.
- Not only JSON, but any key/value/array language (BSON/HDF5/XML/YAML)

```
{
   "molecule": {
        "geometry": [0, 0, 0, 0, 0, 1],
        "atoms": [“He", "He"]
    },
    "driver": "energy",
    “model": {
        "method": "SCF",
        "basis": "sto-3g",
    },
}
```

- Molecule
- Input
- Output
- Optimization Trajectory

```
{
    …Input
    "provenance": {
        "creator": "My QM Program",
        "version": "1.1rc1",
    },
    "properties": {
        “scf_n_iterations”: 2.0,
        "scf_total_energy”: -5.433191881443323,
        "nuclear_repulsion_energy”: 2.11670883436,
        “one_electron_energy”: -11.67399006298957,
    …
    },
    "error": "",
    "success": true,
    "raw_output": "Output storing was not requested."
}
```

Development Efforts

- **Methods:**
  Collections: Datasets, OpenFFWorkflow
  Services: Torsiondrives, geometry optimizations
  Computation: Psi4, RDKit, TorchANI

- **Format:**
  A Quantum Chemistry schema so that different QC programs, AI evaluators, and forcefields can be plugged into the backend.

- **Front-end:**
  Python front end for querying, computing, and visualization. REST API for power users and web front ends.

Front End

pandas

$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$

jupyter

Database Technologies

PostgreSQL

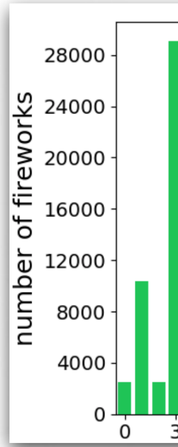
mongoDB

Apache Arrow

Distributed Compute Engines

Parsl

FireWorks

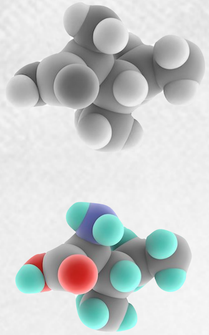


QCSchema: github.com/MolSSI/QCSchema

Production Runs

Molecular Torsion Drives

A. The user runs a client that requests quantum chemical computations for lists of molecules.



H. The molecules with requested quantum chemical data are returned to the user.

B. Clients request results from the server. These are either returned from storage in the server or new computations are initiated.

Client

D. The iterative *torsiondrive* procedure is run on the server and spawns new geomeTRIC computations as previous iterations complete.

Torsion Drive

Server

Database

C. A central sever stores a record of complete computations and can execute “services” such as *torsiondrive*.

Distributed Compute

Local Cluster

Supercomputer

Cloud Compute

E. Compute clients can be run on any resource connected to the internet. These clients request computations and return results.

G. The results of the geomeTRIC / Psi4 quantum computations are formatted via the QC Schema and returned to the database.

Psi4

Geometry Optimization

F. The geometry optimization of a single configuration is completed with geomeTRIC and Psi4.

134 Torsion Drives, **3,582** Optimizations, **76,976** gradient evaluations, **2,200** peak cores, **3** compute clusters

Challenges and Directions

- **Challenges:**
  - Intricate software stack, focusing on simplifying user experience and interaction.
  - Deployment of complex programs on large scale compute resources.
  - Project is ~6 months old at the moment, still early days!

- **Future Directions:**
  - A central database to hold not only Open Force Field data, but data from many molecular science subfields with a web-portal front-end.
  - Enhanced "workflows" or chained quantum chemistry computations.
  - Array-based storage for storing arrays like density matrices.