TREX libraries:

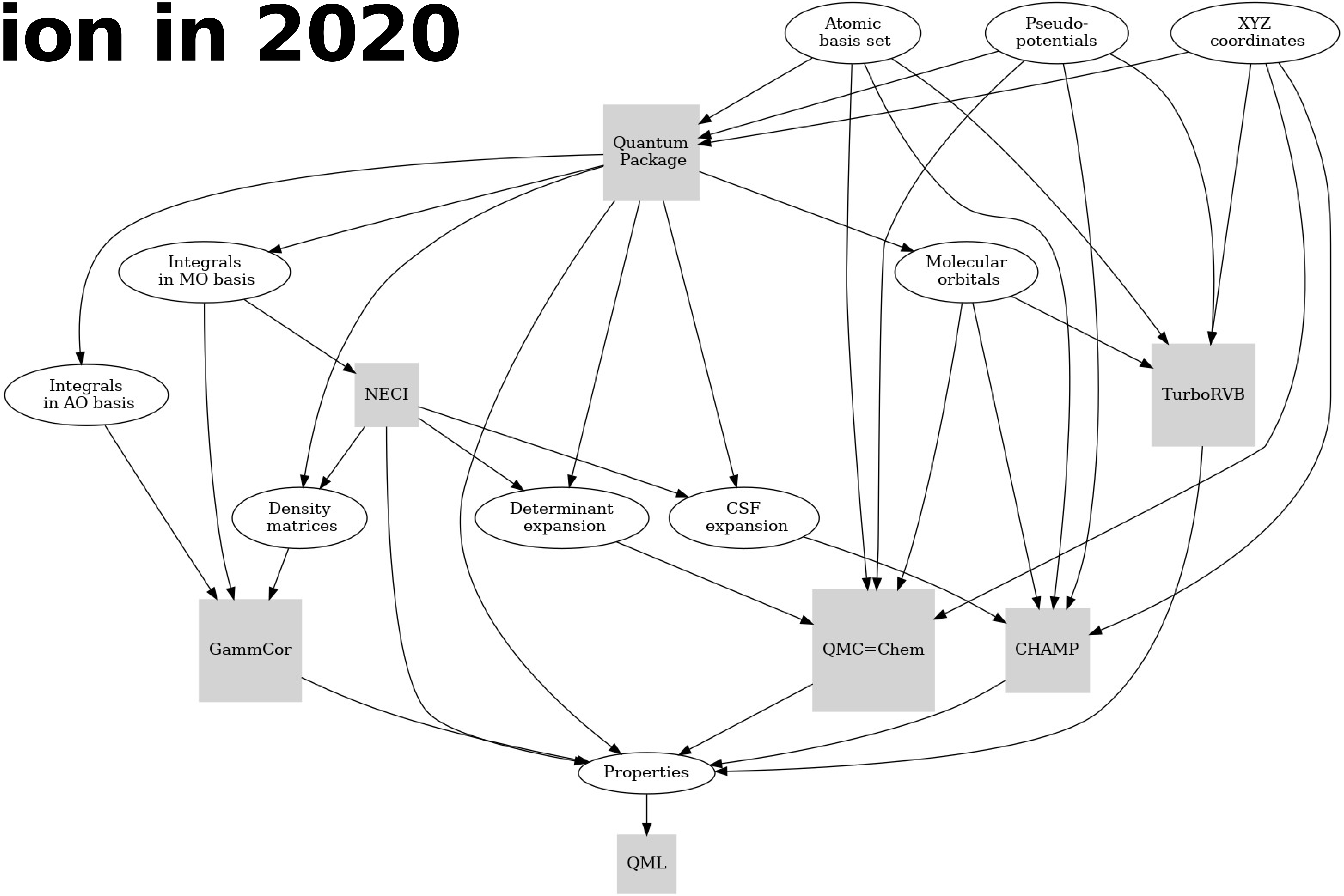
TREXIO & QMCkl

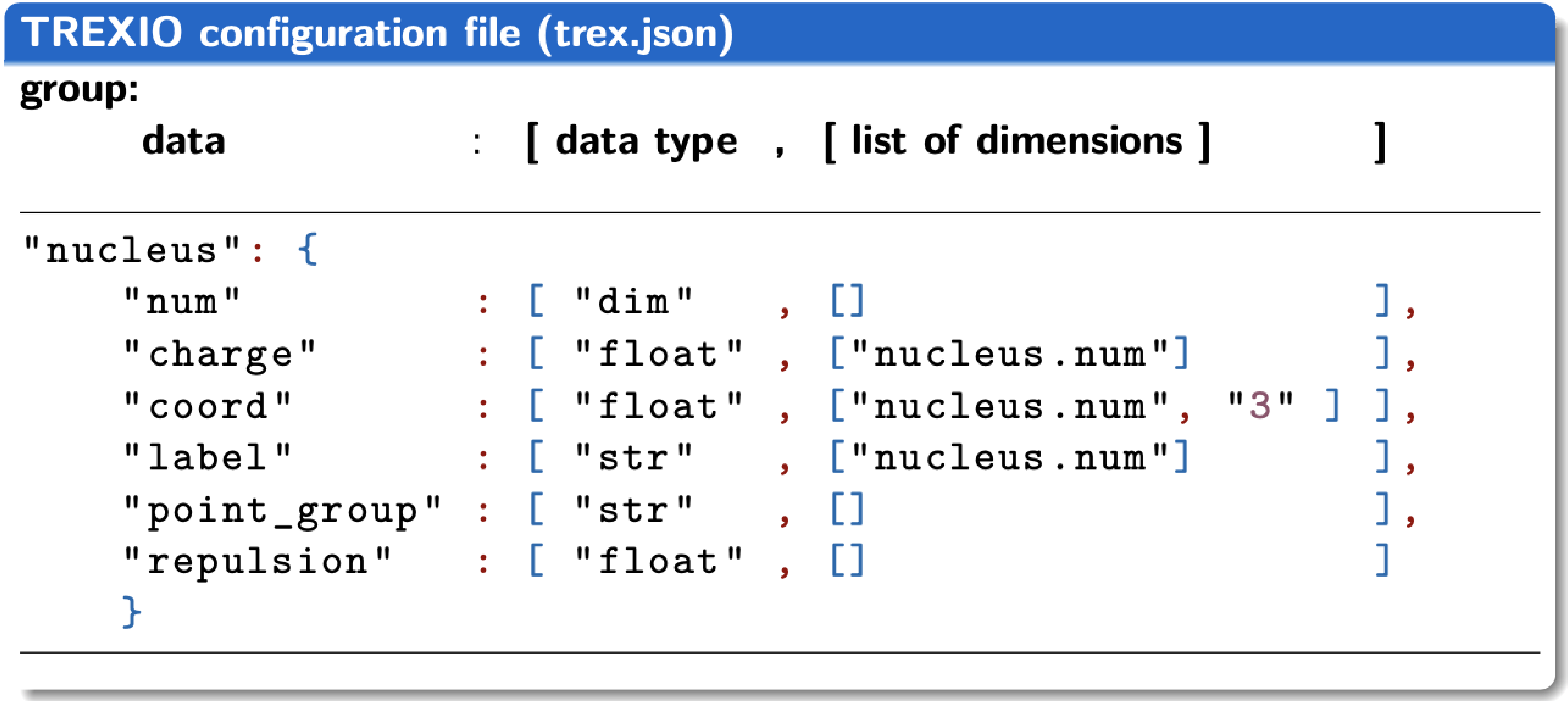
Evgeny Posenitskiy,* Anthony Scemama
Laboratoire de Chimie et Physique Quantiques (LCPQ)
@ CNRS and University of Toulouse, France
***currently @ Qubit Pharmaceuticals, France**

08/02/2023

TREXIO as I/O format

Situation in 2020





TREXIO configuration file (trex.json)

```
group:
      data               :  [ data type  ,  [ list of dimensions ]        ]
```

```
"nucleus": {
    "num"          : [ "dim"   , []                          ],
    "charge"       : [ "float" , ["nucleus.num"]             ],
    "coord"        : [ "float" , ["nucleus.num", "3" ]       ],
    "label"        : [ "str"   , ["nucleus.num"]             ],
    "point_group"  : [ "str"   , []                          ],
    "repulsion"    : [ "float" , []                          ]
    }
```

More details in the TREXIO documentation*

* *https://trex-coe.github.io/trexio/trex.html*

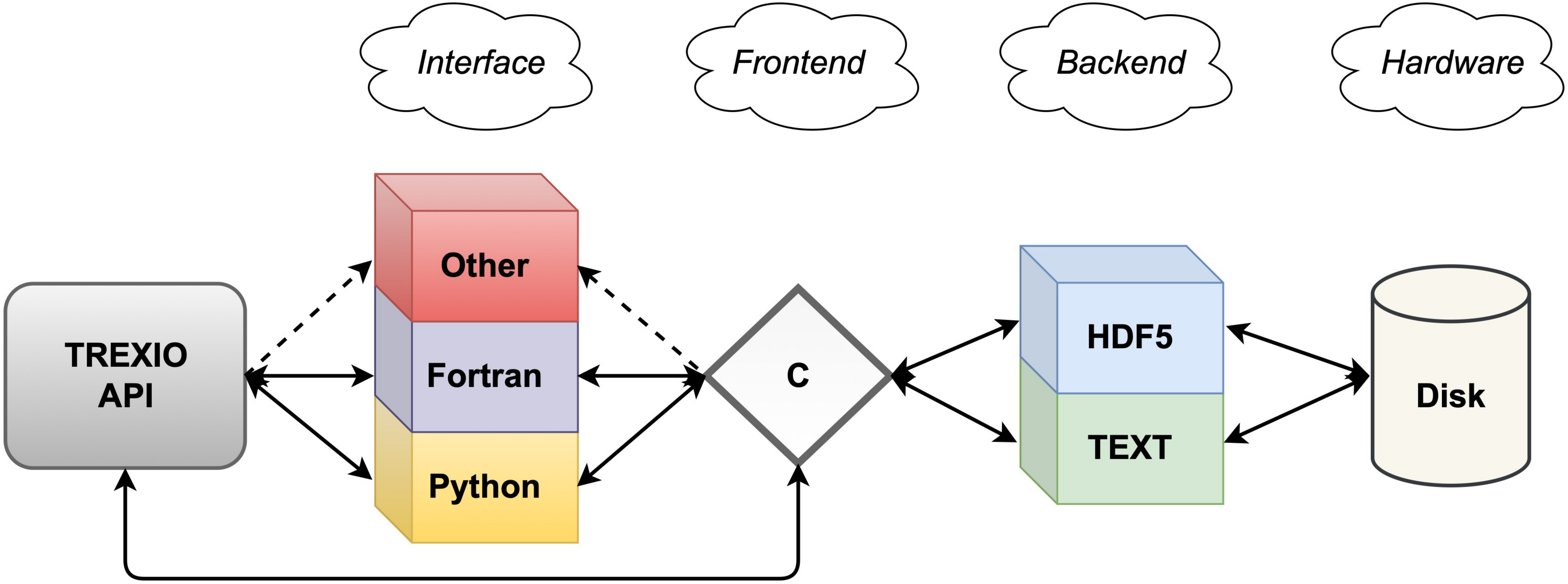
Enhancements compared to other wave function formats

- Fully self-consistent: no code-specific knowledge is required
- Normalization parameters cover all existing ambiguities
- Compact storage of sparse quantities like 2-electron integrals

- **No custom text-based formatting**, forget about typos!

* *https://trex-coe.github.io/trexio/trex.html*

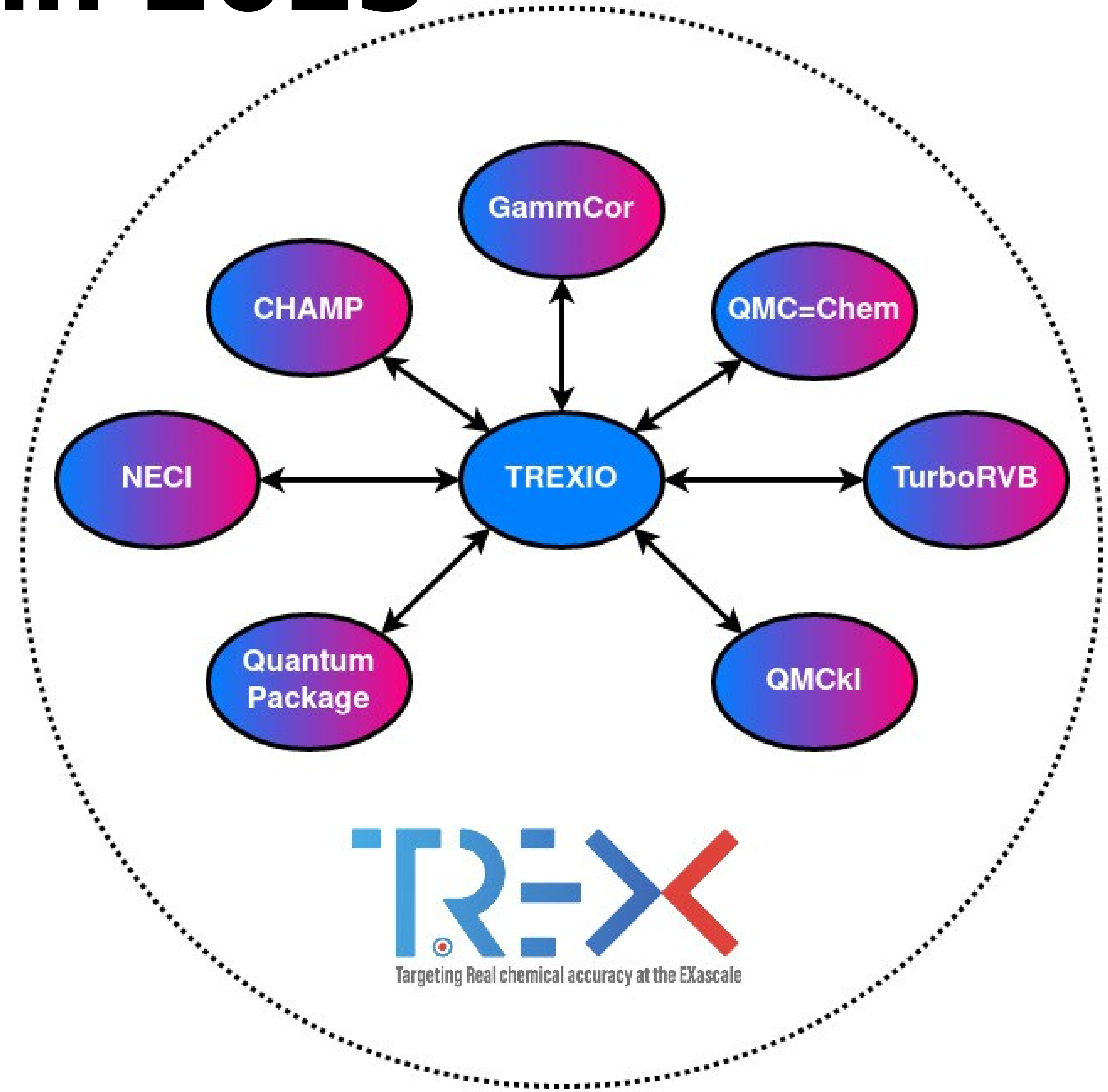
TREXIO as I/O library



- Source code in pure **C** for the best **performance and portability**
- High-performance I/O backend based on the HDF5 library
- Bindings in **Fortran**, **Python**, **OCaml**
- Very **easy to install**: Autotools/CMake, conda, Spack, Guix, pip, apt, opam, you name it :-)

https://github.com/TREX-CoE/trexio

Situation in 2023



Adoption of TREXIO enabled

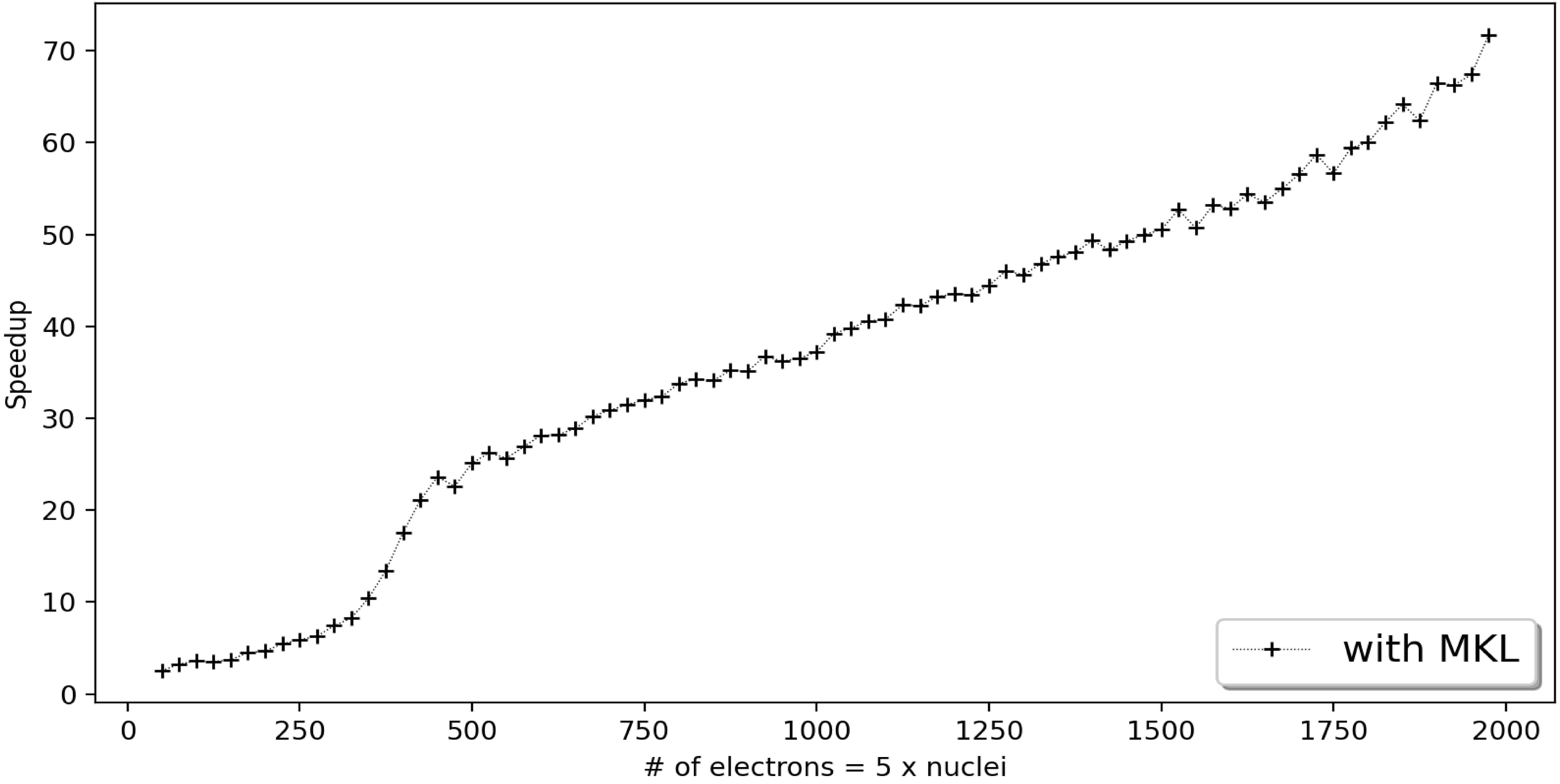
- Enhanced data exchange and I/O performance in TREX codes
- **QP** ⇨ **TREXIO** ⇨ **GammCor** : SAPT with CIPSI density matrices
- **QP** ⇨ **TREXIO** ⇨ **CHAMP** : QMC with CIPSI wave functions
- **trexio_tools** ⇨ **TREXIO** ⇨ **all TREXIO users** are interfaced with external programs like GAMESS, Gaussian, PySCF
- **QP** ⇨ **TREXIO** ⇨ **QMCkl** : user-friendly QMC tutorials in pure Python

QMC Kernel Library: QMCkl

- API for main algorithms of Quantum Monte Carlo
- **Pedagogical** and **high-performance** implementations
- Low-level functions: linear algebra
- High-level functions: domain-specific
- Bindings in **C, Fortran, Python**

https://github.com/TREX-CoE/qmckl

QMCkl use case: Jastrow factor*



*work of Vijay Gopal Chilkuri @ Aix-Marseille University

- **TREXIO repository:** **https://github.com/TREX-CoE/trexio**
- **QMCkl repository:** **https://github.com/TREX-CoE/qmckl**

Thank you for your attention!

The TREX: Targeting Real Chemical Accuracy at the Exascale project has received funding from the European Union’s Horizon 2020 - Research and Innovation program - under grant agreement no. 952165.