PSI Center for Scientific Computing, Theory and Data

AiiDAlab Quantum ESPRESSO App

Streamlining Core-Level Spectroscopy Calculations

Xing Wang
Postdoc, Materials Software and Data Group,
Paul Scherrer Institute, Switzerland
RACXS workshop, 18-06-2024

Outline of one-hour code tutorial

PSI

Part 1: Streamlining scientific workflows with AiiDA

Target Audience: Computational Scientists

Goals:

- Understand the idea of creating workflows using AiiDA
- Learn about checkpoints, provenance tracking (database), and remote execution

Part 2: AiiDAlab Quantum ESPRESSO App

Target Audience:

- Experimental Scientists needing DFT calculations on material properties.
- Computational Scientists want to providing DFT services

Goals:

- Easy-to-use graphical interface
- Accessible, turn-key solution for non-specialists
- Facilitates collaboration between experimentalists and theorists

Code tutorial guidelines

I will conduct live calculation during the session to showcase practical applications.

Part 1: Only a basic understanding of Python is needed.

Your Role:

- You are not required to perform the calculations during the session.
- Full instructions will be provided so you can independently run all demonstrated calculations at your convenience.

Interactive Sessions:

- Each part will have ~ 5 minutes for questions and discussions.
- Please feel free to ask questions or share your thoughts during these intervals, or anytime during the conference.

Small request: speak slowly and maybe repeat your question

Part 1 AiiDA tutorial

AiiDA: Automated **I**nteractive **I**nfrastructure and **Da**tabase for Computational Science

An open-source Python package to help researchers with automating, managing, persisting, sharing and reproducing the complex workflows associated with modern computational science and all associated data.

Key features:

- Automated workflows
- High-throughput
- Data provenance
- Database and advanced queries
- Plugin interface
- HPC interface
- Open science
- Open source

Challenges in high-throughput HPC

- **Workflow automation**
  - Need tools to define complex workflows with advanced error handling
  - An automated, robust and scalable engine to run the workflows
- **Data management**
  - Data should be stored reliably and efficiently
  - Stored data should be interoperable and queryable, and easy sharing
- **Reproducibility**
  - All produced data should be reproducible by storing the full provenance

Data provenance

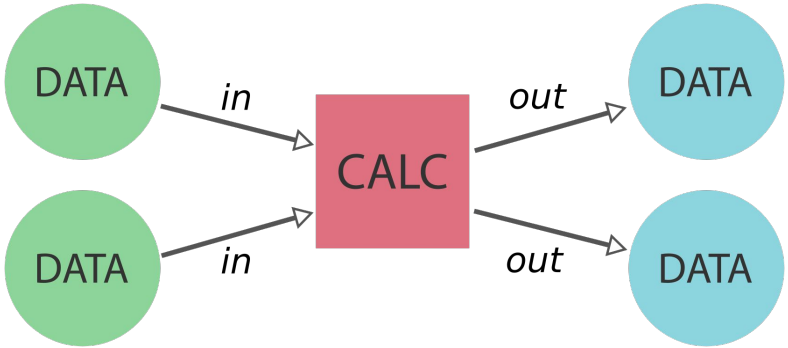
PSI

Simple recipe

- Store the '**calculations**'
- Store its **inputs** and their metadata
- Store its **outputs** and their metadata
- Most **crucially** store the **inter-connections**

Calculation: x+y

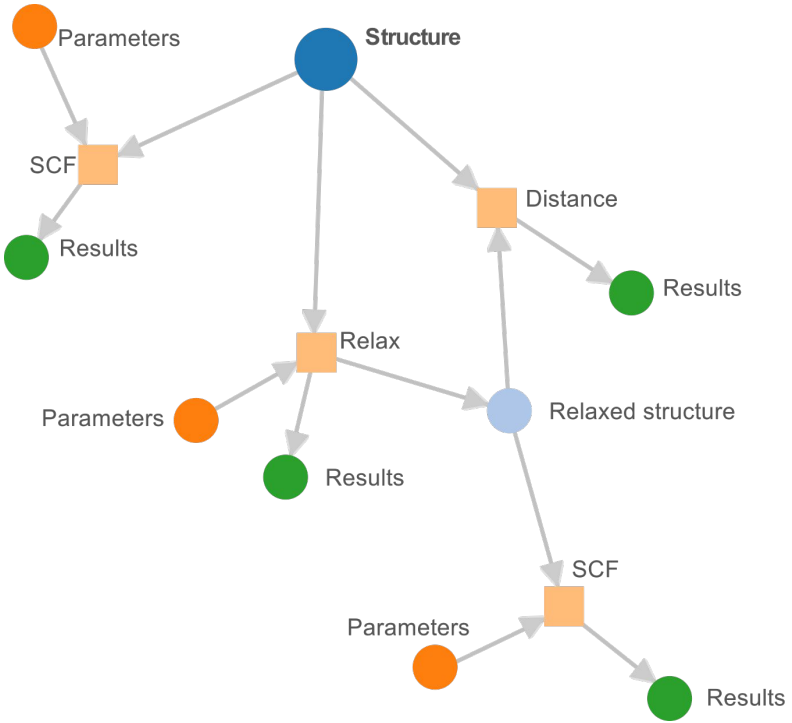
```
def add(x, y):
    return x + y

add(1, 2)
```



Data provenance

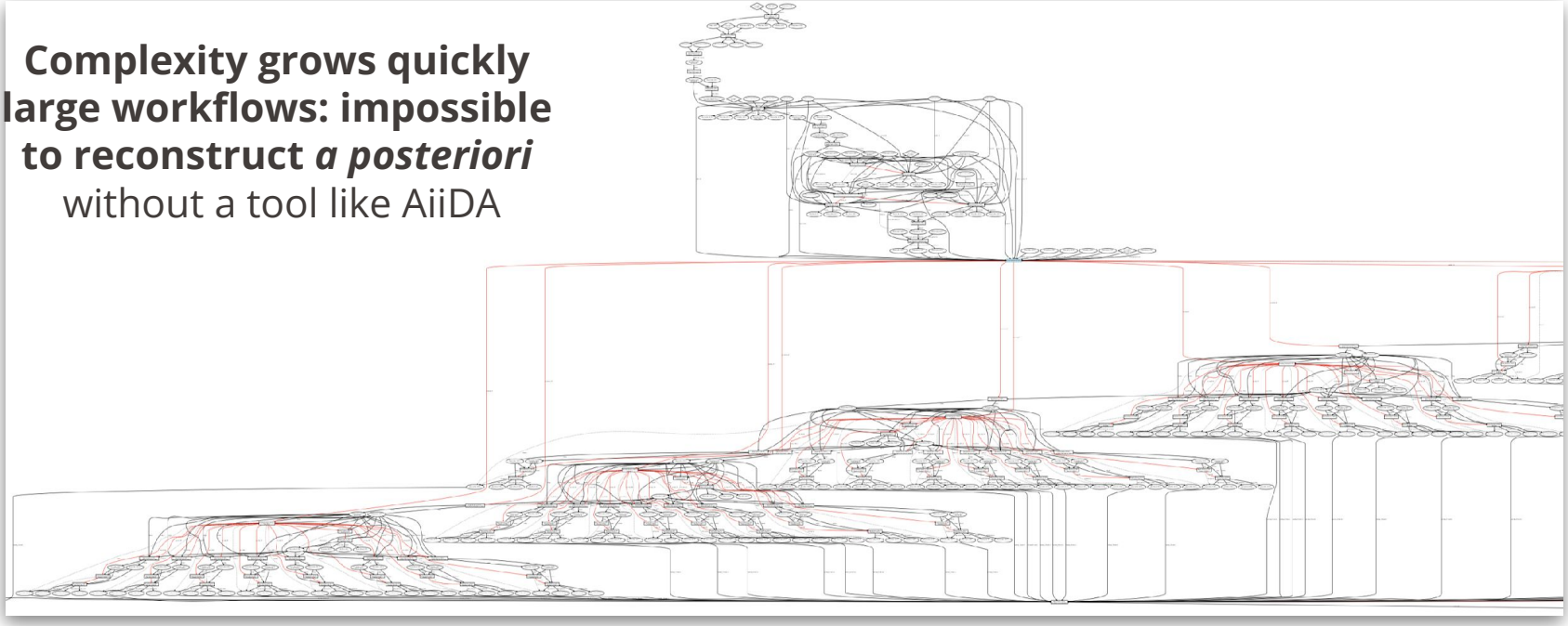
Provenance graphs

- When data gets reused, a directed graph is created
- That quickly grow in complexity for workflows



Data provenance

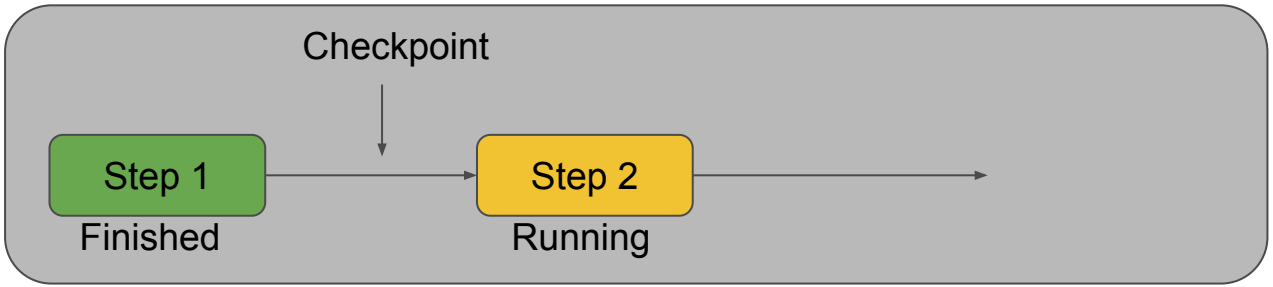
Complexity grows quickly large workflows: impossible to reconstruct *a posteriori* without a tool like AiiDA



Molecular dynamics study of Lithium in a solid electrolyte

Multi-steps Workflow

Automatic checkpointing, which guarantees that work between steps is saved, thus can be can be restored after it had been interrupted. Important for long-running calculation, like DFT.



- **WorkChain**
- **WorkGraph**: flexible and easy to use

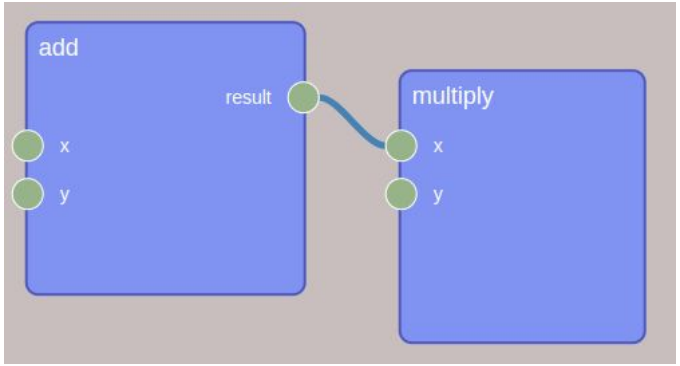
AiiDA-WorkGraph

step 1 step 2

Workflow example: (x + y) * z

1) Create a empty workgraph
2) Add the tasks (the add and multiple functions)
3) Link the task

```
from aiida_workgraph import WorkGraph

@calcfunction()
def add(x, y):
    return x + y

@calcfunction()
def multiply(x, y):
    return x*y

wg = WorkGraph("add_multiply")
wg.tasks.new(add, name="add")
wg.tasks.new(multiply, name="multiply",
                    x=wg.tasks["add"].outputs["result"])
```



Remote calculation

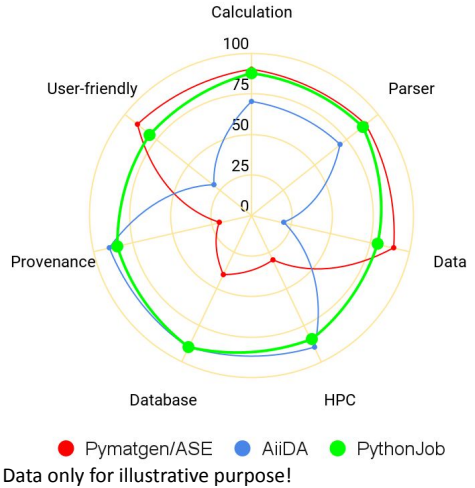
- **CalcJob: run code (e.g. DFT code) on remote computer.**
- **ShellTask: run shell command on a remote computer**
- **PythonTask: run Python function on a remote computer (Ongoing)**

```python
@task
def add(x, y):
    return Int(x + y)

wg = WorkGraph("second_workflow")
wg.tasks.new(add, name="add", run_remotely=True)
```

Turn it into a PythonTask, and run this function on a remote computer

```
# run on CSCS eiger cluster
metadata = {
    "options": {
        'prepend_text' : """
module load cray/22.05  cpeIntel/22.05 QuantumESPRESSO/7.0
eval "$(/users/xingwang/miniconda3/bin/conda shell.posix hook)"
conda activate py3.11
export OMP_NUM_THREADS=1
        """,
        'resources': {
                    'num_machines' : 1,
                    'num_mpiprocs_per_machine' : 128,
        }
    }
}
```

Integrate Your Favorite Tools with AiiDA

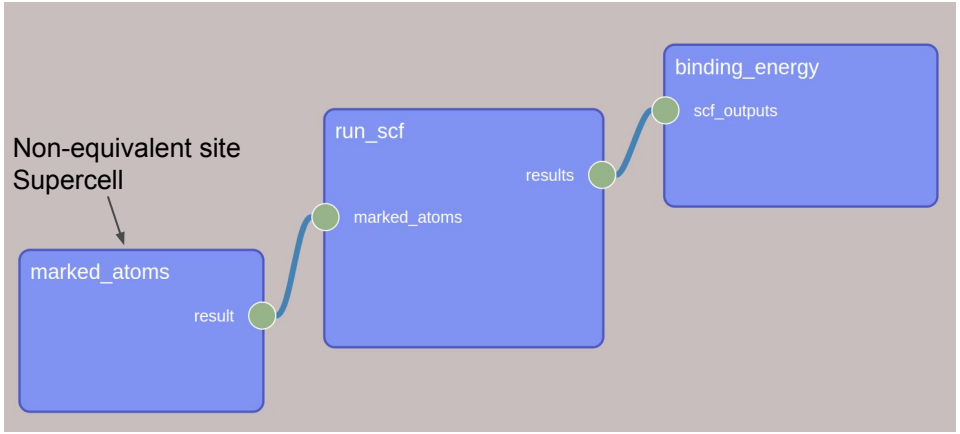


Data only for illustrative purpose!

Xing Wang

WorkGraph examples

- Espresso calculator
  - Calculator
  - Atomization energy
  - Equation of state (EOS)
  - Elastic constants
  - Bands structure
  - Projected density of states (PDOS)
  - Bader Charge
  - X-ray photoelectron spectroscopy (XPS)



Summary

AiiDA: Automated **I**nteractive **I**nfrastructure and **Da**tabase for Computational Science

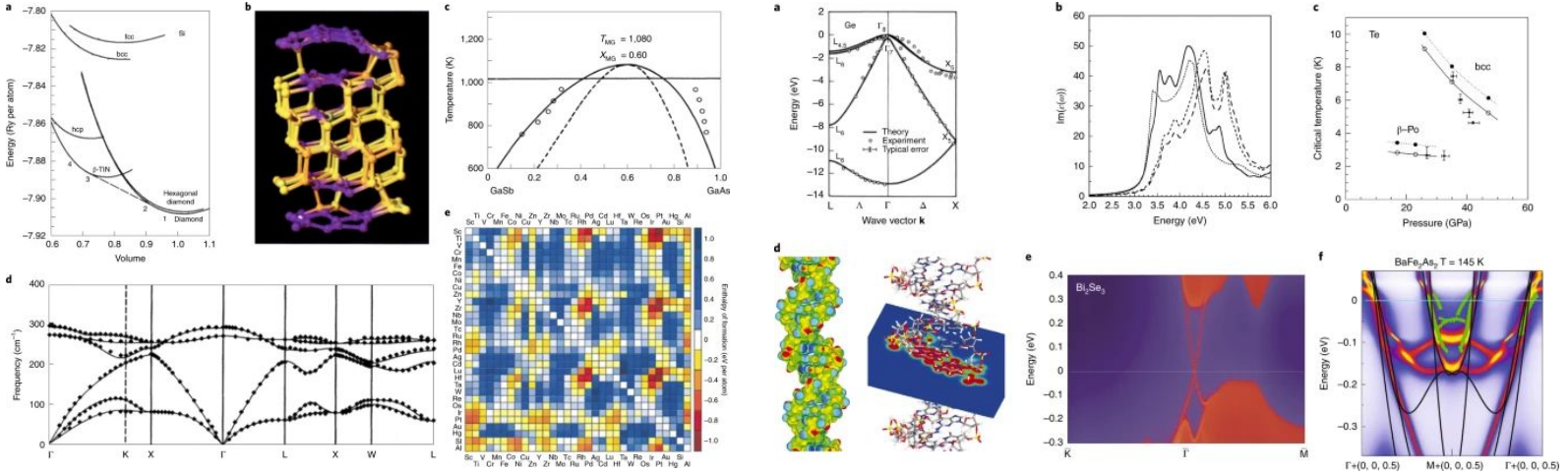
Key Features:

- **Provenance:** Track the provenance of the workflow.
- **Easy to build workflow:** Create workflows by linking the input and output socket of different tasks.
- **Flexible:** Extend (modify) the workflow by adding (editing) tasks and links, or combine multiple workflows together.
- **Interactive GUI:** Visualize and interact with the workflow using the GUI. (Ongoing)
- **Checkpoints:** Save the workflow state, and restore the workflow from the a interrupted state.
- **Remote execution:** Execute the task (Python function, Shell command) on a remote machine.

Part 2 AiiDAlab Quantum ESPRESSO App

Power of first-principles calculations

- Remarkable predictions in the wide range of properties and spectroscopies



Marzari, N., Ferretti, A. & Wolverton, C. Nat. Mater. 20, 736–749 (2021).

Bring the first-principle tools to other communities

Broadly available simulation codes and workflows.

Barrier

Software Installation and Configuration

Input File Preparation

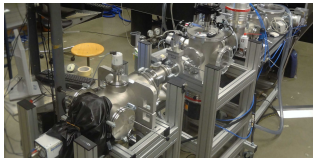
Parameter Tuning

Interpreting Results

Computational Resources

Validation and Benchmarking

Big research and industry communities



Bring the first-principle tools to big communities

Broadly available simulation codes and workflows.

Our Goal

Automated, reliable, "turn-key" simulations tool

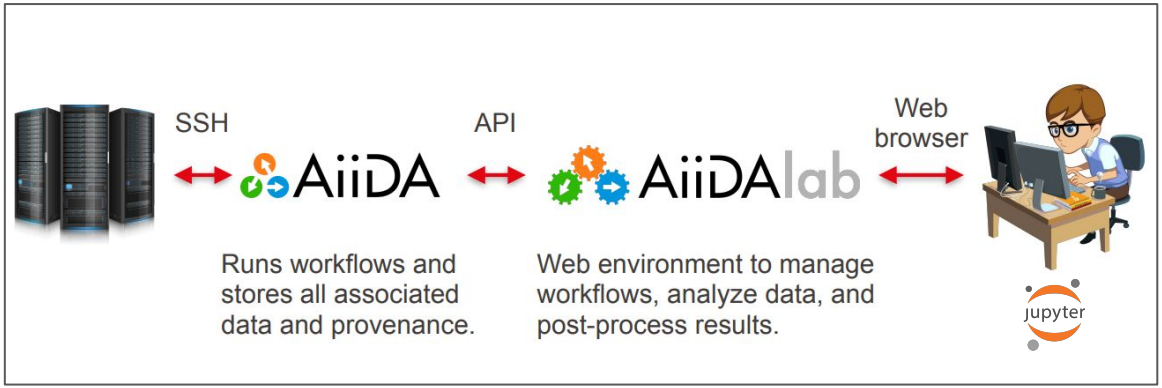
Barrier

Big research and industry communities

AiiDAlab-QuantumESPRESSO app

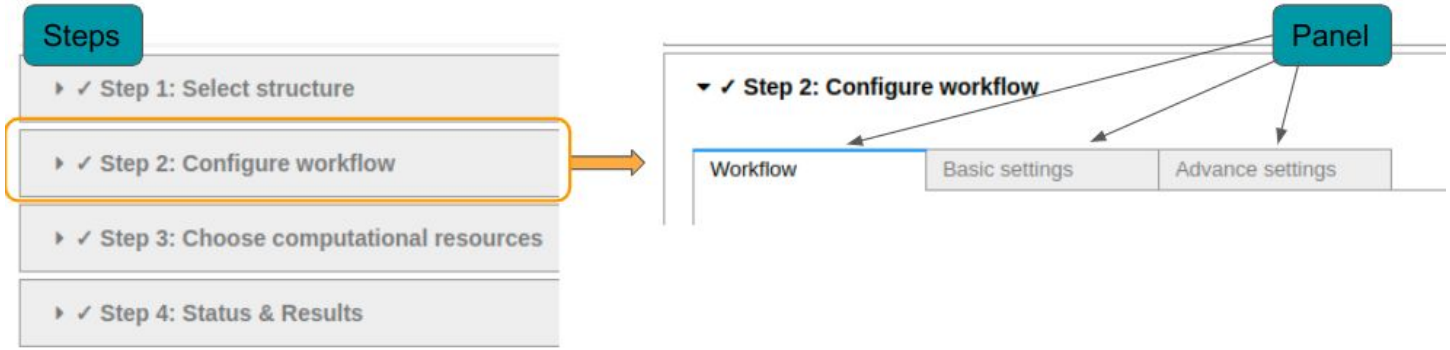
- Quantum ESPRESSO (QE) app is an AiiDAlab web-based graphical user interface allowing users to conduct first-principle calculations on the web browser directly without writing any code.

  - Automated
  - Reliable
  - Intuitive user interface
  - Easy to share and maintain



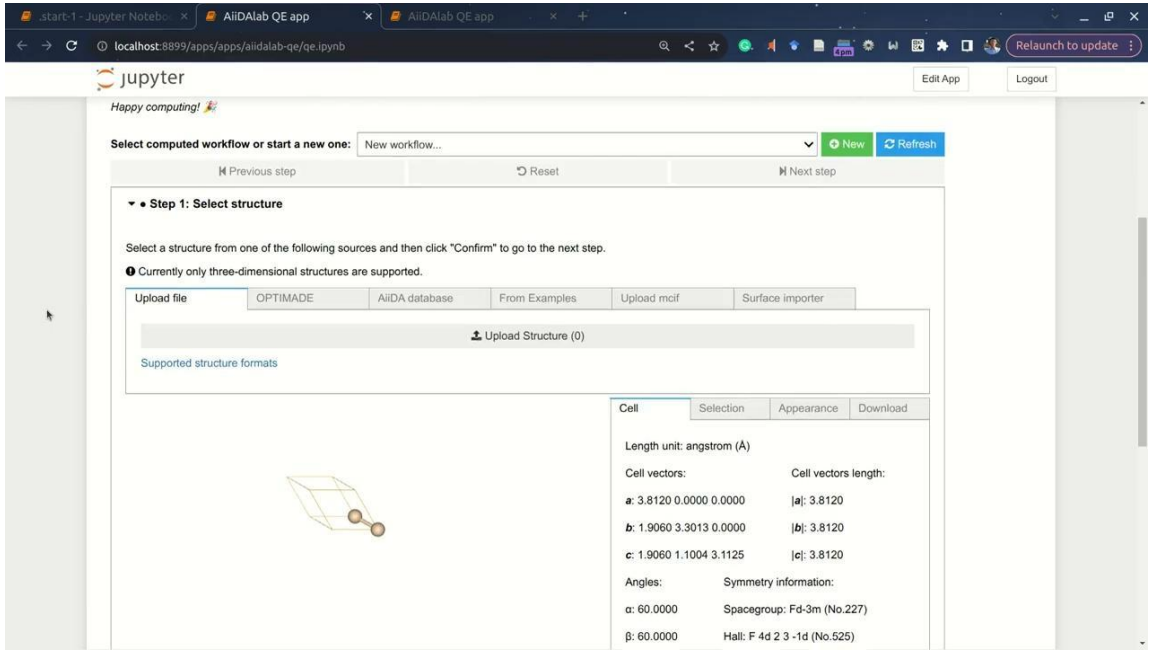
Quantum ESPRESSO app

- Quantum ESPRESSO app uses the Wizards UI, which divides one calculation job into four steps.
- Each step contains several panels.



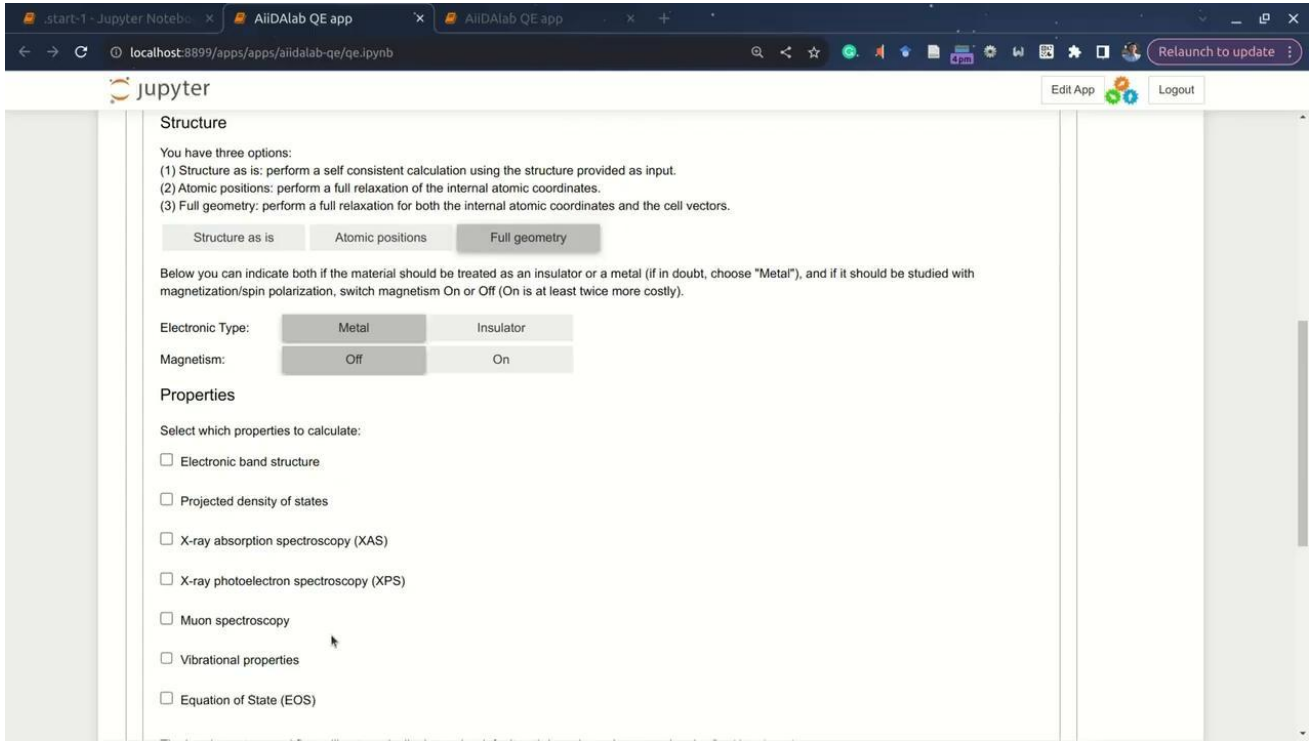
Step 1 Select a structure

- Structures can be uploaded as a file in any standard format (XYZ, CIF, etc.) or be downloaded from online databases (e.g., Materials Cloud, Materials Project) using the [OPTIMADE API].



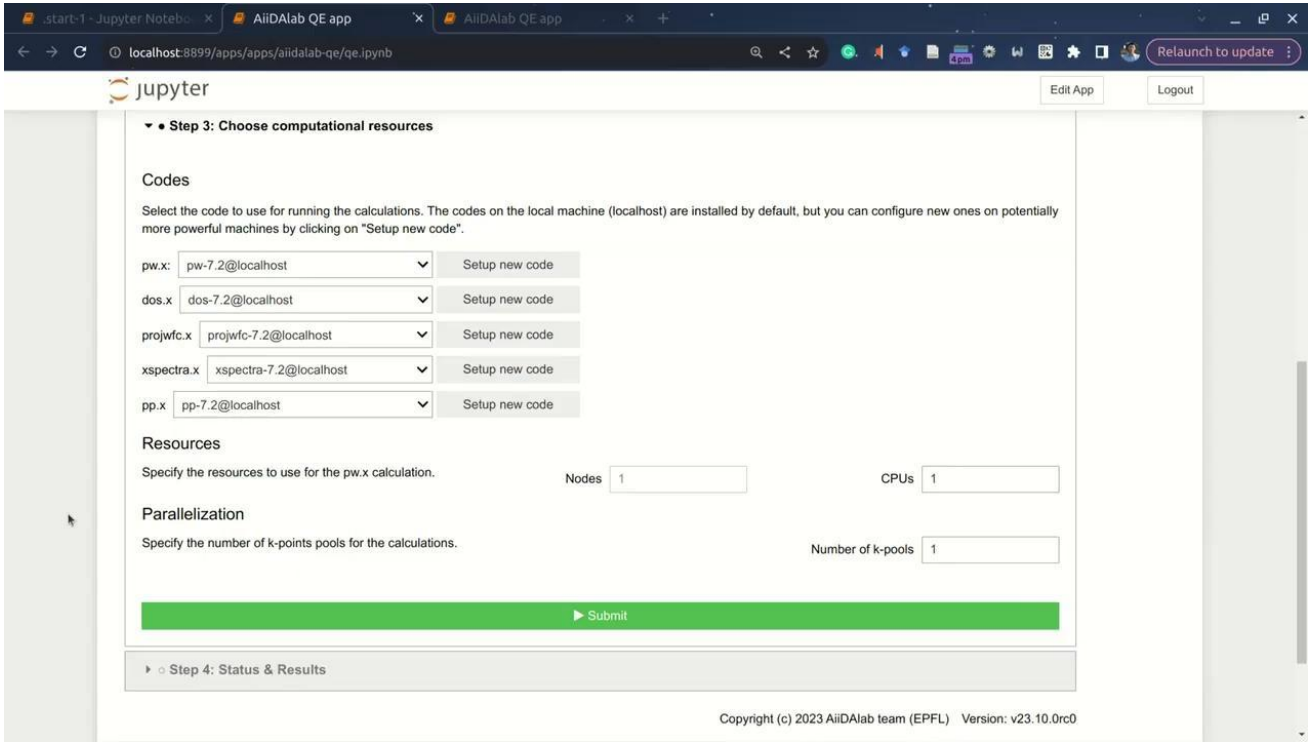
Step 2: Set the parameters for DFT calculation

- Easy parameter selection for non-experts by choosing the protocols, which are developed and tested for a large database of structures.



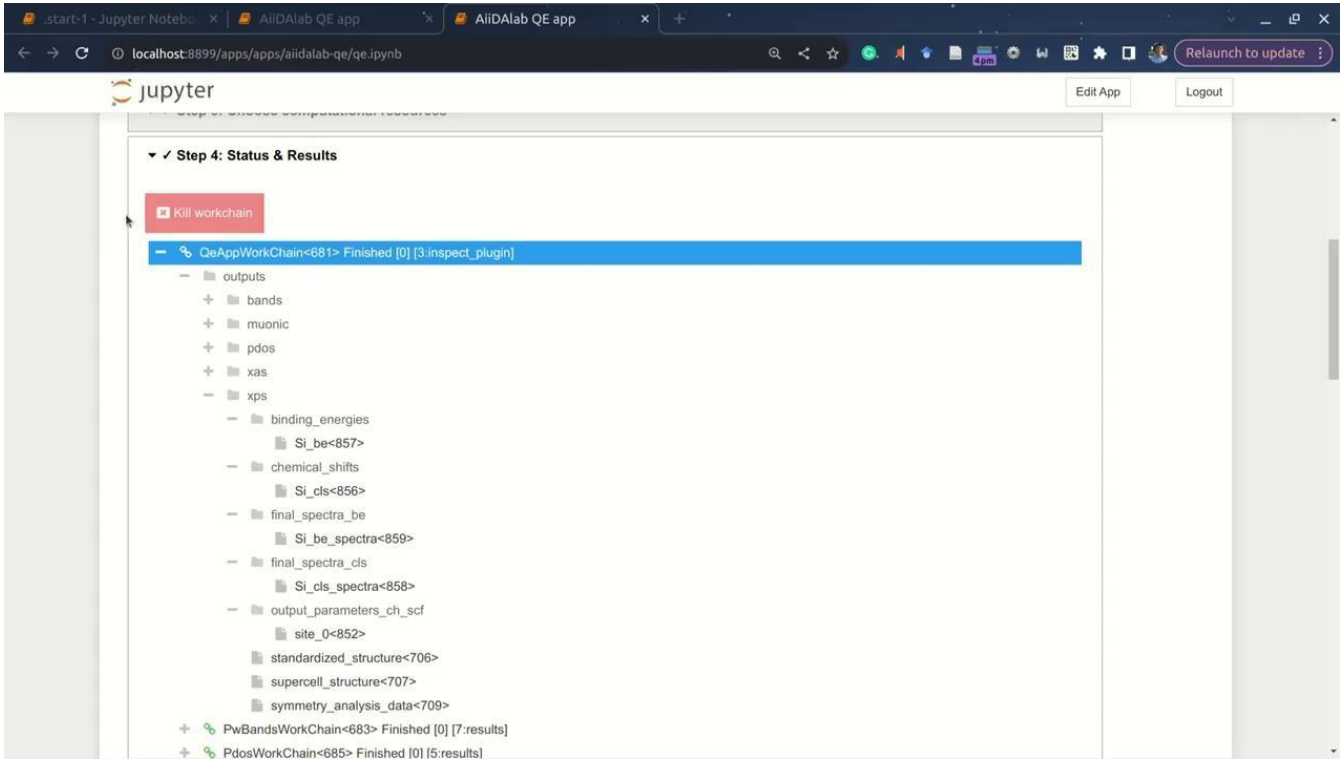
Step 3: Set up the computational resources

- Set up the code on the local machine or on the remote HPC.



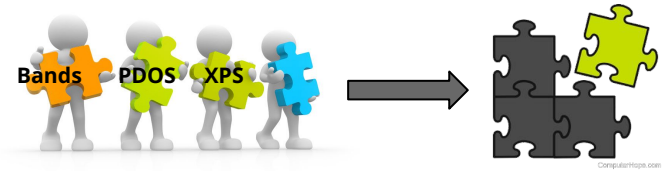
Step 4: Visualize the results

- Visualize the result interactively.

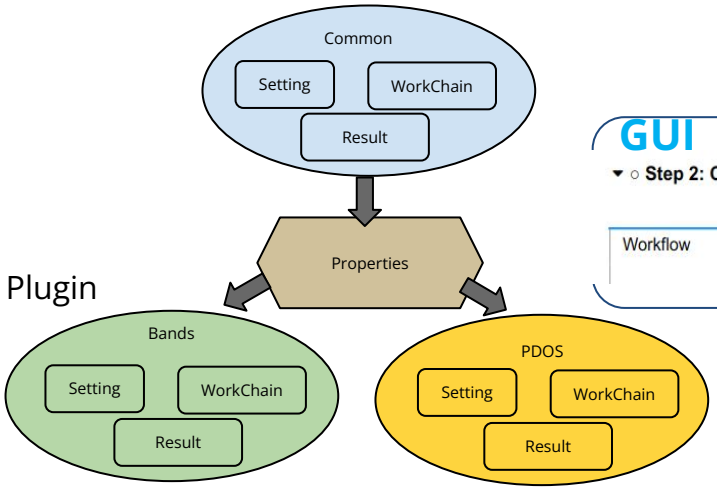


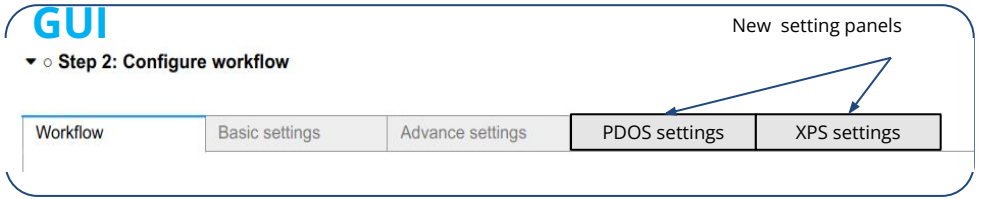
Architecture: Plugin

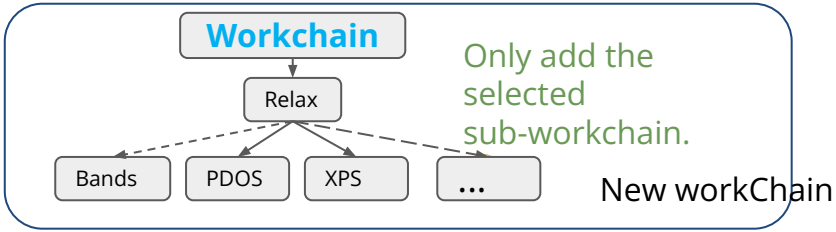
- Running multiple calculations (bands, pdos, XPS, etc) in one app using the same interface.



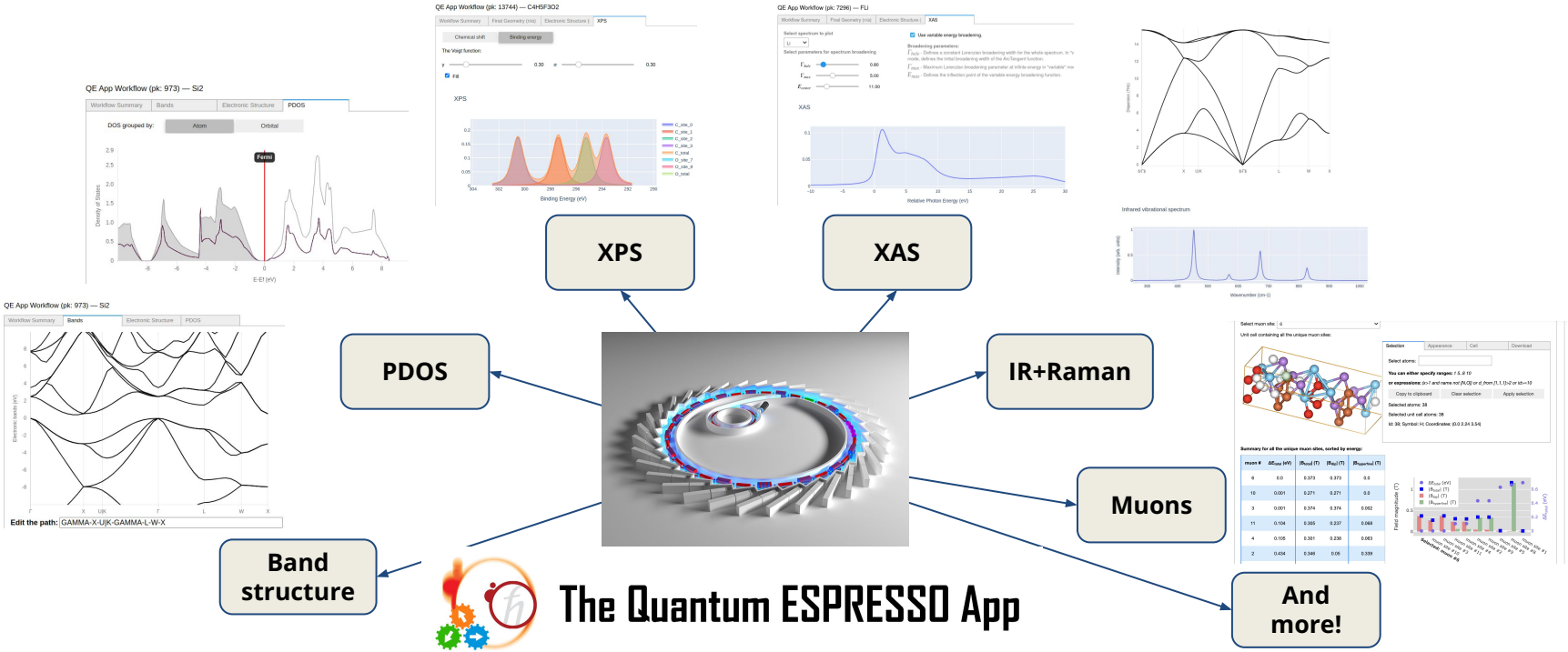
Plugin







Power of Quantum ESPRESSO app



Contributors: E. Bainglass (PSI), M. Bercx (PSI), M. Bonacci (PSI), D. Dou (EPFL), P. Gillespie (CNR-NANO Modena), M. A. Hernández-Bertrán (CNR-NANO Modena), D. Hollas (Bristol), N. Marzari (EPFL, PSI), A. Ortega-Guerrero (Empa), C. Pignedoli (Empa), D. Prezzi (CNR-NANO Modena), G. Pizzi (PSI), I. Timrov (EPFL, PSI), J. Yu (PSI), A. Yakutovich (Empa)

XPS inside the AiiDAlab-Quantum ESPRESSO app

PSI

Theory

ΔKS

$$E_B = E_+ - E_0$$

Pseudo-potential with core-hole

Molecular systems:

$$E_B = E_{\mathrm{ch}}^{+} - E_0$$

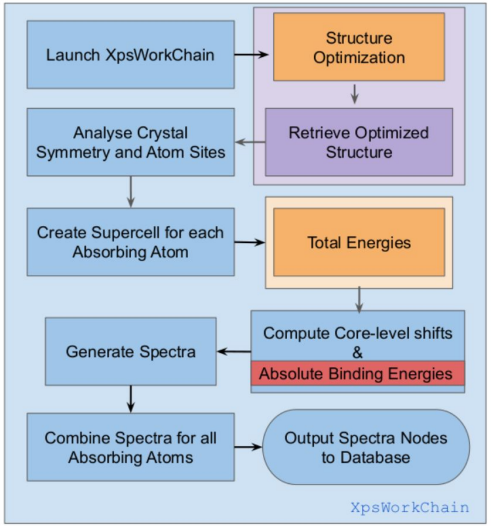
Solid systems:

$$E_B^F = E_{\mathrm{ch}}^{+} - E_0 + \epsilon_F$$

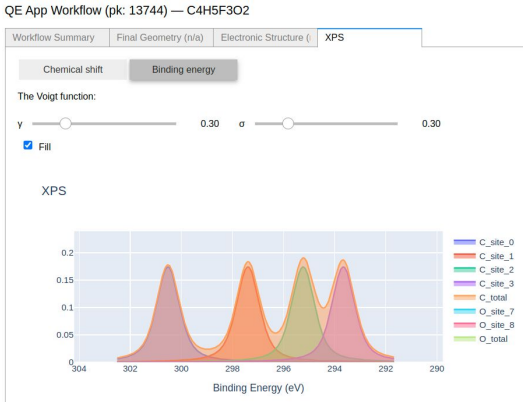
$$E_{\mathrm{ch}}^{0} = E_{\mathrm{ch}}^{+} + \epsilon_L$$

$$E_B^F = E_{\mathrm{ch}}^{0} - E_0 + \epsilon_F - \epsilon_L$$

AiiDA workflow



AiiDAlab GUI



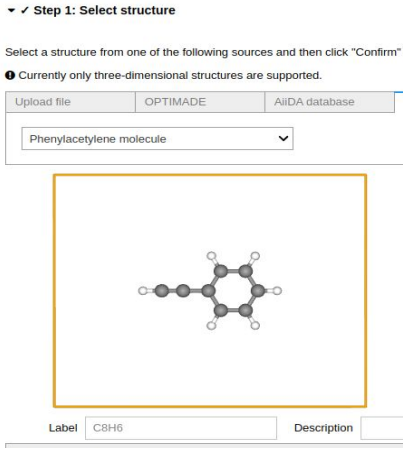
Collaborate with Dr. Prezzi's group (CNR-NANO Modena), Dr. Iurii Timrov (PSI)

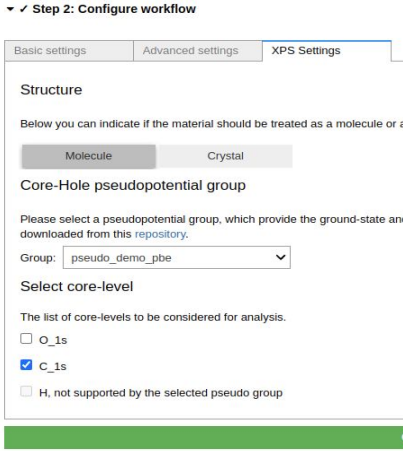
[1] M. Walter *et al.*, Phys. Rev. B. 94, 041112(R) (2016)

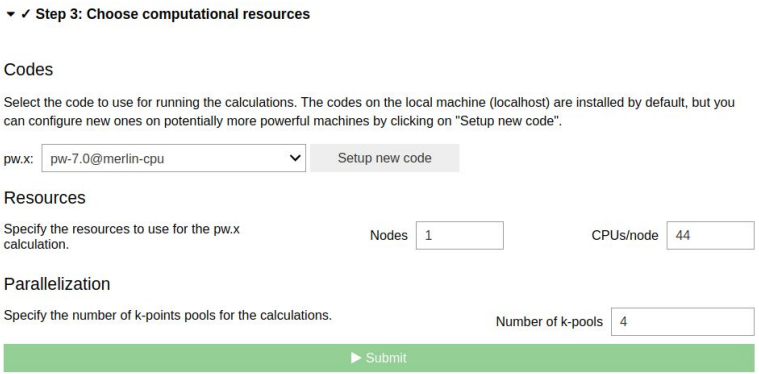
Xing Wang

XPS demo

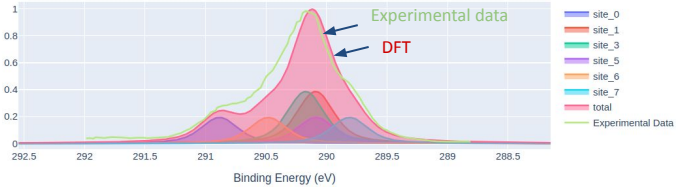
Example: Phenylacetylene (PA) molecule







Step 4 Result



https://aiidalab-qe.readthedocs.io/howto/xps.html

How to access?

- Docker image: [aiidalab/qe:latest](), with pre-built environments, e.g., pseudopotentials, code setup.

Single User

1-50 Users

100-1000+ Users

Documentation

GitHub

- We also maintain a few official cloud-backed deployments
  https://www.aiidalab.net/deployments/
  http://aiidalab.psi.ch/

Code tutorial (Part 1)

PSI

Run the docker image

docker run -p 8888:8888 superstar54/qeapp-tutorial

Click File Manager icon
Go to /home/jovyan/tutorial/
Open the tutorial notebook, and follow the guides there.

Code tutorial (Part 2)

PSI

Run the docker image

docker run -p 8888:8888 superstar54/qeapp-tutorial

Then follow this tutorial:
https://aiidalab-qe.readthedocs.io/howto/xps.html

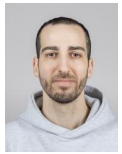
Conclusions

- AiiDAlab Quantum ESPRESSO app allows researchers to arrange DFT calculations on the web browser directly without writing any code.
- QEApp is modularized and pluggable, by using plugin to extend its functionality. Developers can maintain and share their plugin easily.
- QEApp serves as an optimal platform for seamlessly incorporating the theoretical advancements while concurrently facilitating accessibility for both experimentalists and theorists within the scientific community.

Acknowledgements - AiiDA, AiiDAlab, Materials Cloud

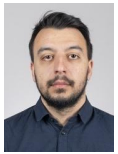
PSI

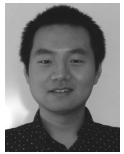
The current AiiDA, AiiDAlab and Materials Cloud teams


Edan Bainglass (PSI)


Roberto Bendinelli (EPFL)


Marnik Bercx (EPFL)


Miki Bonacci (PSI)


Dou Du (EPFL)


Kristjan Eimre (EPFL)


Guillaume Fraux (EPFL)


Valeria Granata (EPFL)


Daniel Hollas (U. Bristol)


Sebastiaan P. Huber (Microsoft)

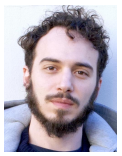

François Liot (EPFL)


Nicola Marzari (EPFL, PSI)


Carlo Pignedoli (Empa)


Giovanni Pizzi (PSI)


Louis Ponet (EPFL)


Francisco F. Ramirez (EPFL)


Thomas Schulthess (ETHZ,CSCS)

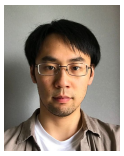

Chris Sewell (EPFL)


Leopold Talirz (Microsoft)


Joost VandeVondele (ETHZ,CSCS)


Xing Wang (EPFL)


Aliaksandr Yakutovich (Empa)


Jusong Yu (EPFL)

XPS and XAS app: Collaborate with Dr. Prezzi's group (CNR-NANO Modena), Dr. Iurii Timrov (PSI)

And all former team members, who actively contributed with ideas and code to make these platforms what they are today.