OPTIMADE

Open Databases Integration for
Materials Design

AL4MS Tutorial, Aalto University
28th February 2023

Matthew Evans

ml-evs.science

UCLouvain

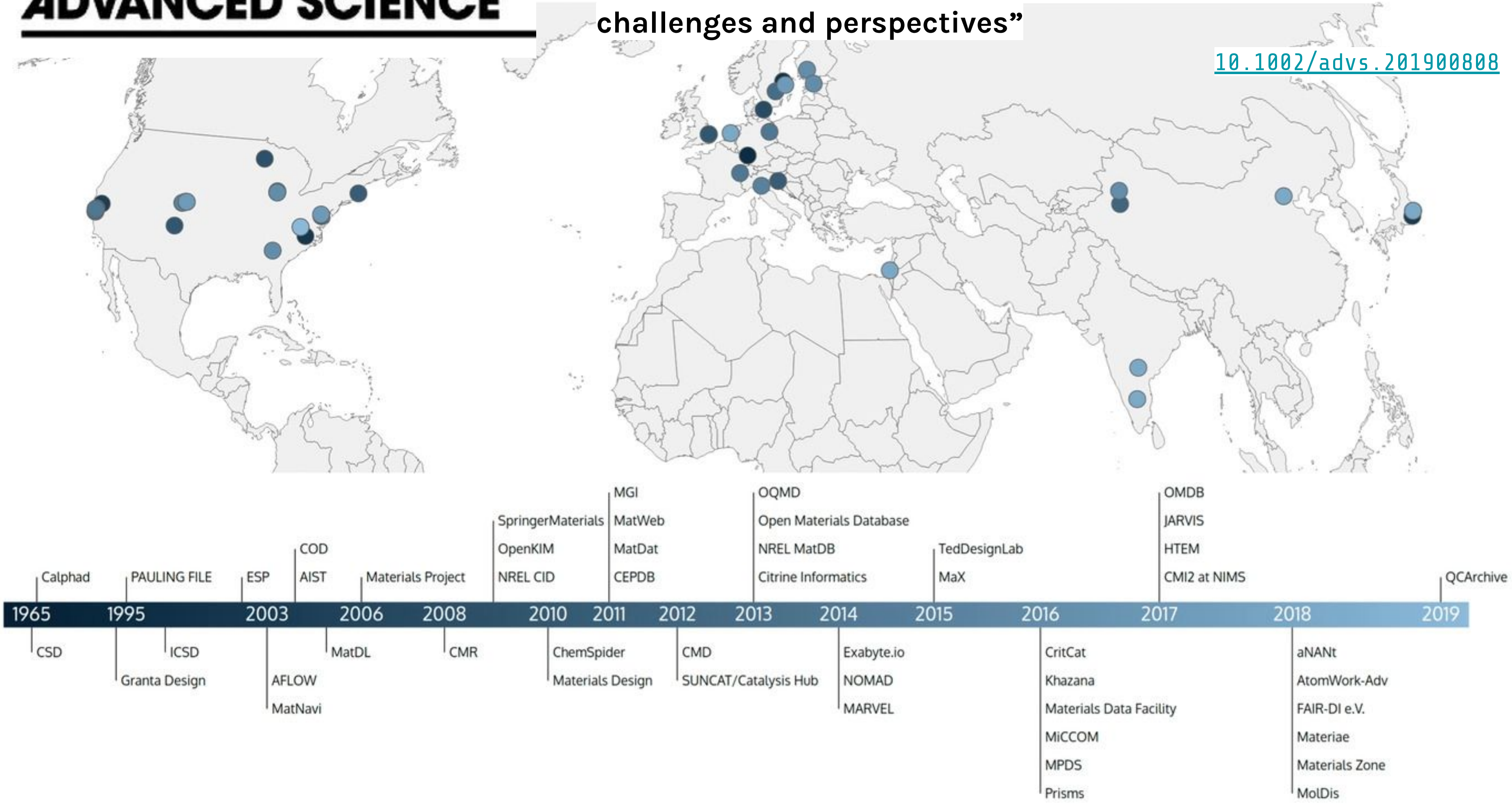
Matgenix

https://www.mentimeter.com/app/presentation/al7jhw95xxvoradntxt7egcbp46ba185/src3as461z6i

ADVANCED SCIENCE

Himanen *et al.* “Data-driven materials science: status, challenges and perspectives”

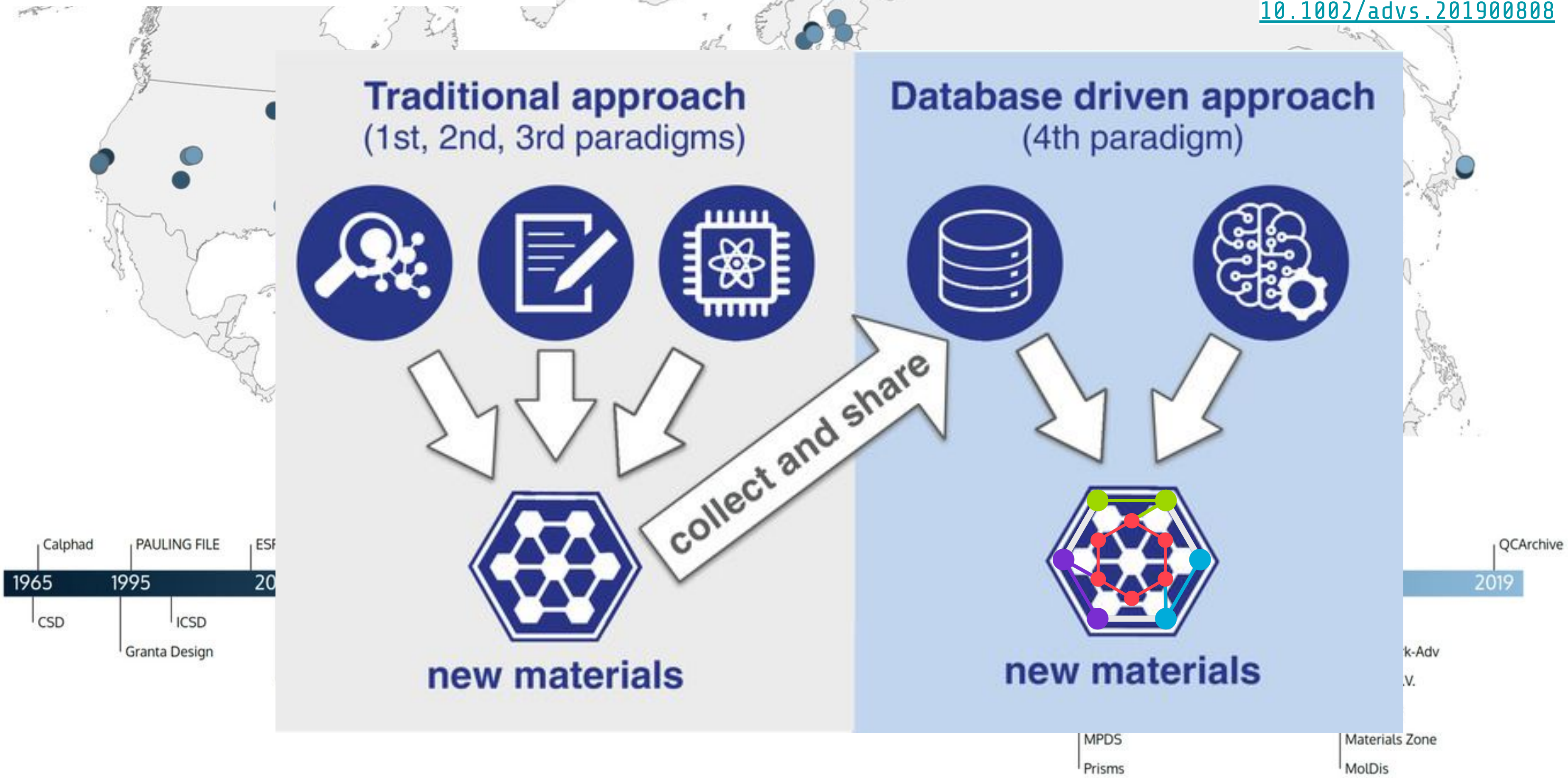
10.1002/advs.201900808



ADVANCED SCIENCE

Himanen *et al.* “Data-driven materials science: status, challenges and perspectives”

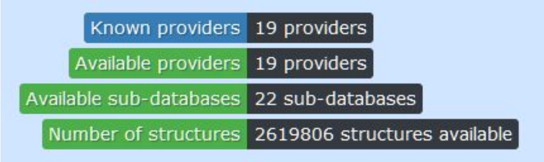
10.1002/advs.201900808



What is OPTIMADE?

"The Open Databases Integration for Materials Design (OPTIMADE) consortium aims to make materials databases interoperational by developing a common REST API."

- ❖ 6 annual workshops (est. 2016)
- ❖ 60+ authors/attendees
- ❖ 25,701 words
- ❖ 19 registered providers, 26M+ crystal structures



Known providers	19 providers
Available providers	19 providers
Available sub-databases	22 sub-databases
Number of structures	2619806 structures available

https://www.optimade.org/providers-dashboard

Why OPTIMADE?

Access to data can vary in:

- syntax
  - Structure of URLs, response formats
- taxonomy
  - Similar/identical concepts have different names
- semantics
  - What it *means* to be an entry in a given database

Why OPTIMADE?

Each database has its own niche:

- Materials Project
  - Curated DFT calculations of stability and properties for “all” reported materials
- NOMAD
  - An archive of any atomistic or electronic structure code output file
- COD
  - A long-running curated set of experimental crystal structures from the literature, primarily from XRD but also other methods
- odbx
  - Some stuff from my PhD

Why OPTIMADE?

Decentralized
data
unification

A simply query: SiO_2

http://aflow.org/API/aflux/?species(Si,O),nspecies(2)

https://materialsproject.org/rest/v2/materials/SiO2/vasp/structure

https://www.crystallography.net/cod/result?formula=O2 Si&format=json

A simply query: SiO_2

http://aflow.org/API/aflux/?species(Si,O),nspecies(2)

https://materialsproject.org/rest/v2/materials/SiO2/vasp/structure

https://www.crystallography.net/cod/result?formula=O2 Si&format=json

A simply query: SiO_2

http://aflow.org/API/aflux/?species(Si,O),nspecies(2)

https://materialsproject.org/rest/v2/materials/SiO2/vasp/structure

https://www.crystallography.net/cod/result?formula=O2 Si&format=json

A simply query: SiO_2

http://aflow.org/API/aflux/?species(Si,O),nspecies(2)

https://materialsproject.org/rest/v2/materials/SiO2/vasp/structure

https://www.crystallography.net/cod/result?formula=O2 Si&format=json

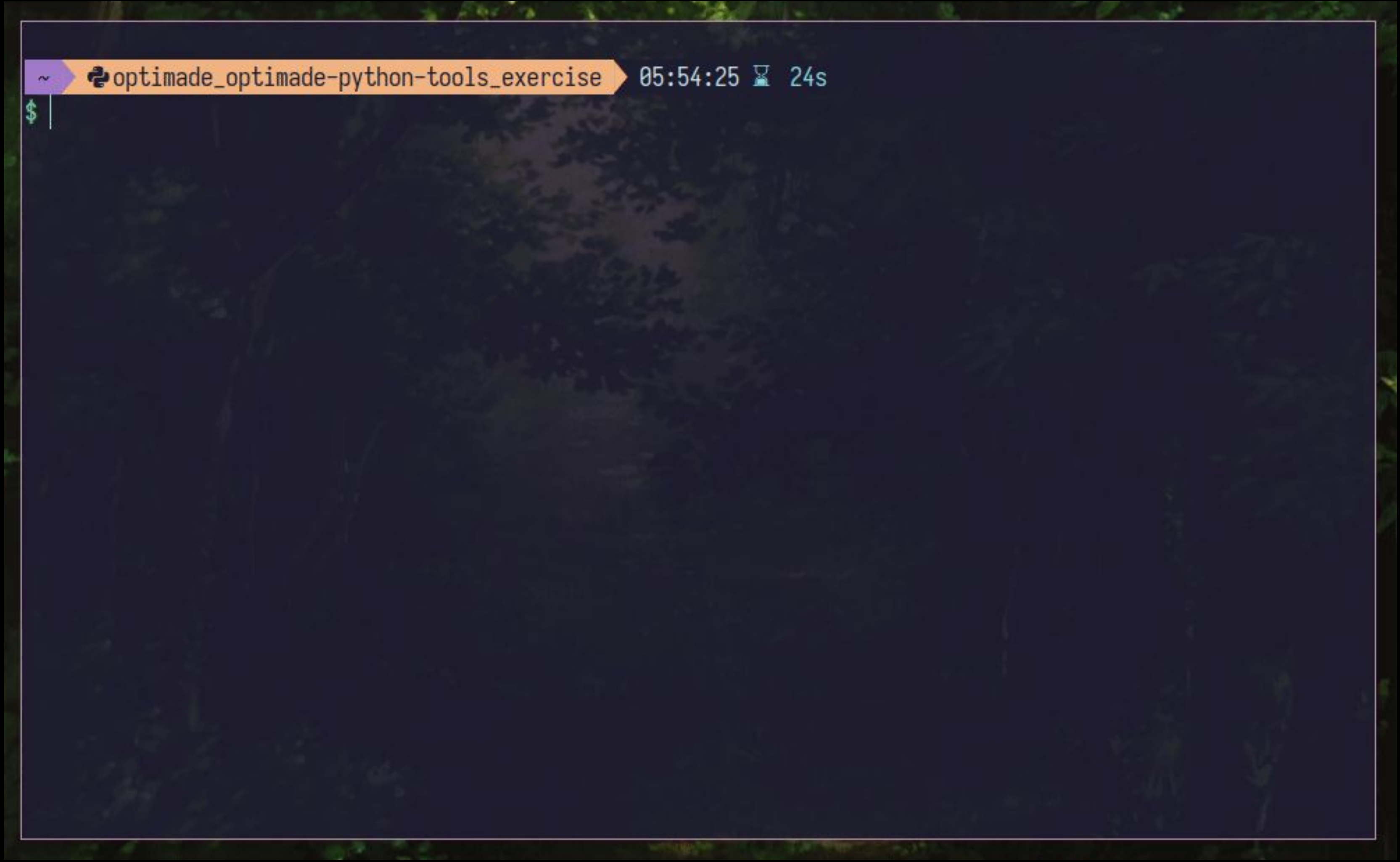
http://aflow.org/API/aflux/?species(Si,O),nspecies(2)

https://materialsproject.org/rest/v2/materials/SiO2/vasp/structure

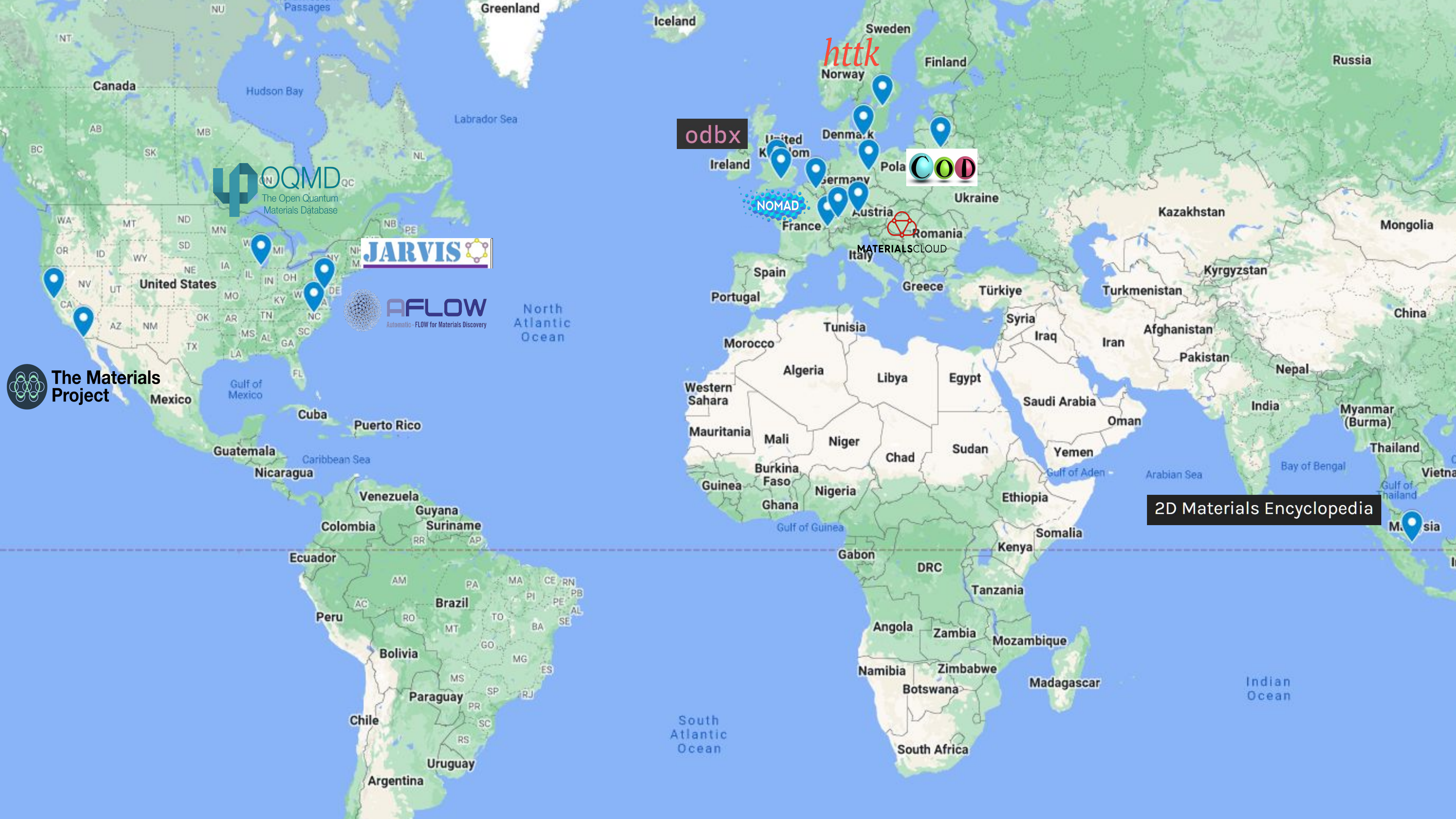
https://www.crystallography.net/cod/result?formula=O2 Si&format=json

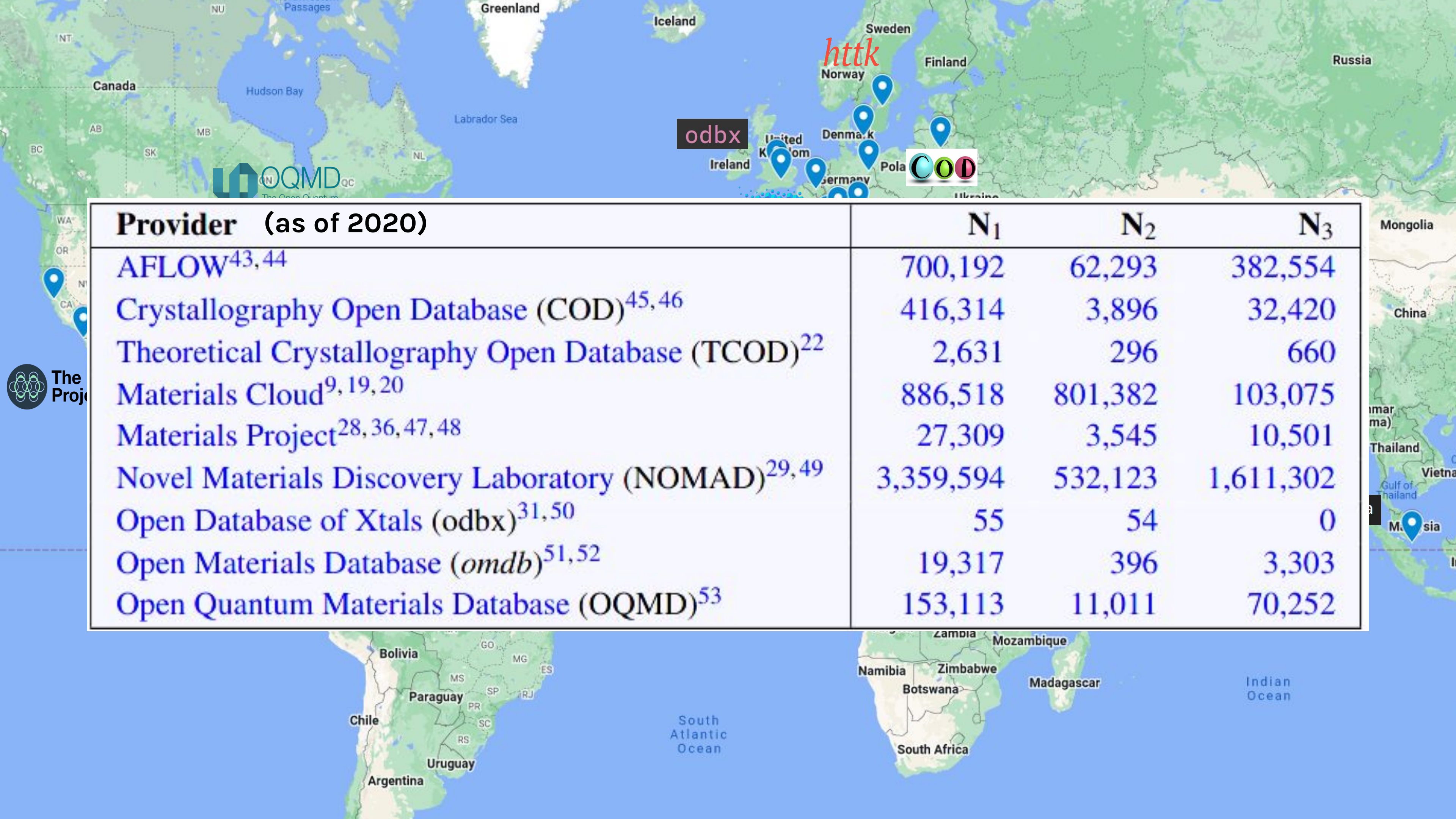
Can we make this machine-actionable?

https://chat.openai.com/chat



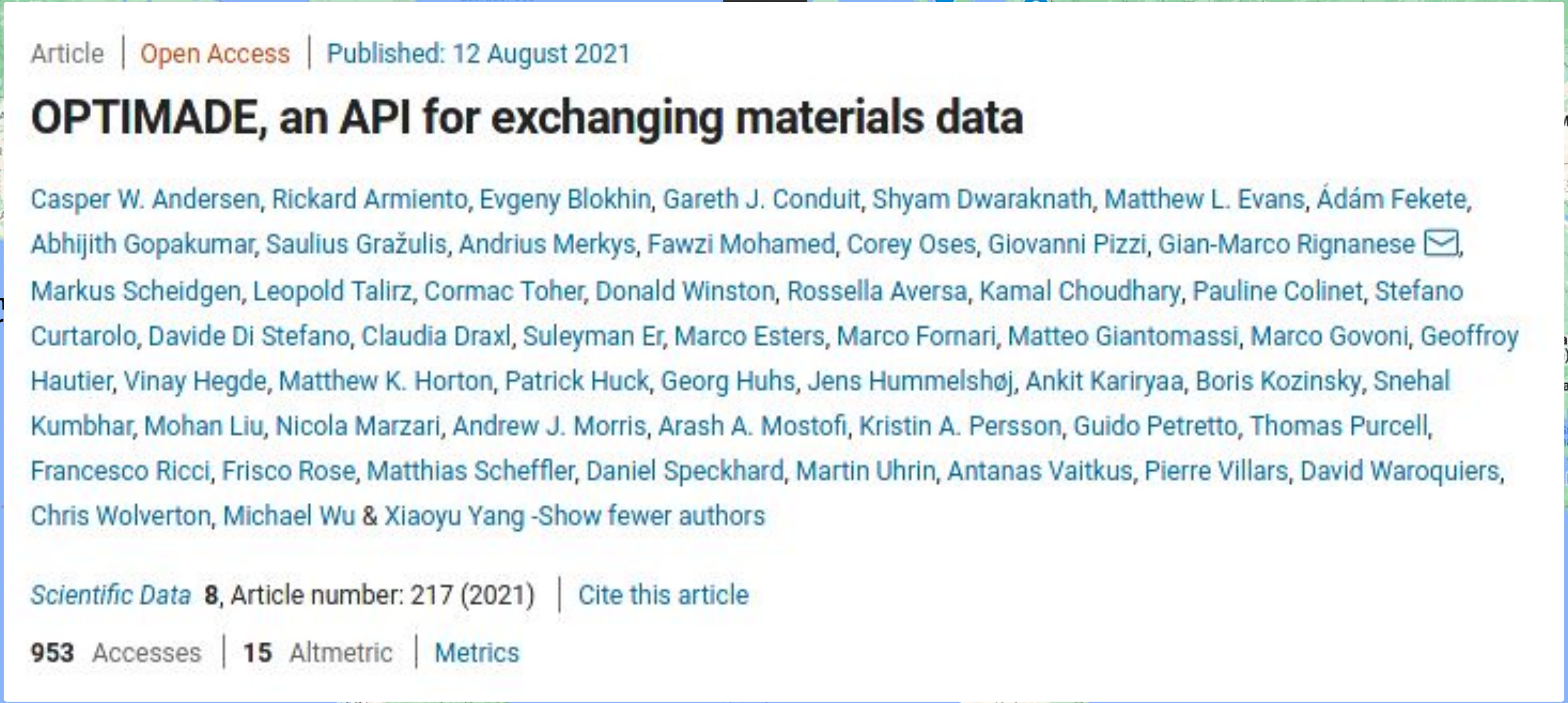
```
~  optimade_optimade-python-tools_exercise  05:54:25 ⌛ 24s
$ 
```


httk
odbx
COD
NOMAD
MATERIALSCLOUD
OQMD
The Open Quantum Materials Database
JARVIS
AFLOW
Automatic - FLOW for Materials Discovery
The Materials Project
2D Materials Encyclopedia



Provider (as of 2020)	N_1	N_2	N_3
AFLOW[43, 44]	700,192	62,293	382,554
Crystallography Open Database (COD)[45, 46]	416,314	3,896	32,420
Theoretical Crystallography Open Database (TCOD)[22]	2,631	296	660
Materials Cloud[9, 19, 20]	886,518	801,382	103,075
Materials Project[28, 36, 47, 48]	27,309	3,545	10,501
Novel Materials Discovery Laboratory (NOMAD)[29, 49]	3,359,594	532,123	1,611,302
Open Database of Xtals (odbx)[31, 50]	55	54	0
Open Materials Database (*omdb*)[51, 52]	19,317	396	3,303
Open Quantum Materials Database (OQMD)[53]	153,113	11,011	70,252

httk



Article | Open Access | Published: 12 August 2021

OPTIMADE, an API for exchanging materials data

Casper W. Andersen, Rickard Armiento, Evgeny Blokhin, Gareth J. Conduit, Shyam Dwaraknath, Matthew L. Evans, Ádám Fekete, Abhijith Gopakumar, Saulius Gražulis, Andrius Merkys, Fawzi Mohamed, Corey Oses, Giovanni Pizzi, Gian-Marco Rignanese ✉, Markus Scheidgen, Leopold Talirz, Cormac Toher, Donald Winston, Rossella Aversa, Kamal Choudhary, Pauline Colinet, Stefano Curtarolo, Davide Di Stefano, Claudia Draxl, Suleyman Er, Marco Esters, Marco Fornari, Matteo Giantomassi, Marco Govoni, Geoffroy Hautier, Vinay Hegde, Matthew K. Horton, Patrick Huck, Georg Huhs, Jens Hummelshøj, Ankit Kariryaa, Boris Kozinsky, Snehal Kumbhar, Mohan Liu, Nicola Marzari, Andrew J. Morris, Arash A. Mostofi, Kristin A. Persson, Guido Petretto, Thomas Purcell, Francesco Ricci, Frisco Rose, Matthias Scheffler, Daniel Speckhard, Martin Uhrin, Antanas Vaitkus, Pierre Villars, David Waroquiers, Chris Wolverton, Michael Wu & Xiaoyu Yang -Show fewer authors

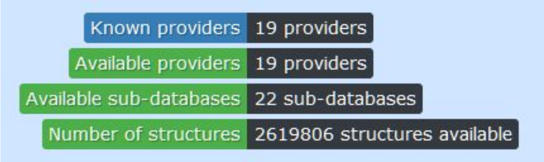
Scientific Data **8**, Article number: 217 (2021) | Cite this article

953 Accesses | **15** Altmetric | Metrics

What is OPTIMADE?

"The Open Databases Integration for Materials Design (OPTIMADE) consortium aims to make materials databases interoperational by developing a common REST API."

- 6 annual workshops (est. 2016)
- 60+ authors/attendees
- 25,701 words
- 19 registered providers, 26M+ crystal structures



Known providers	19 providers
Available providers	19 providers
Available sub-databases	22 sub-databases
Number of structures	2619806 structures available

Technical aspects

Jargon: APIs, HTTP, REST and JSON:API

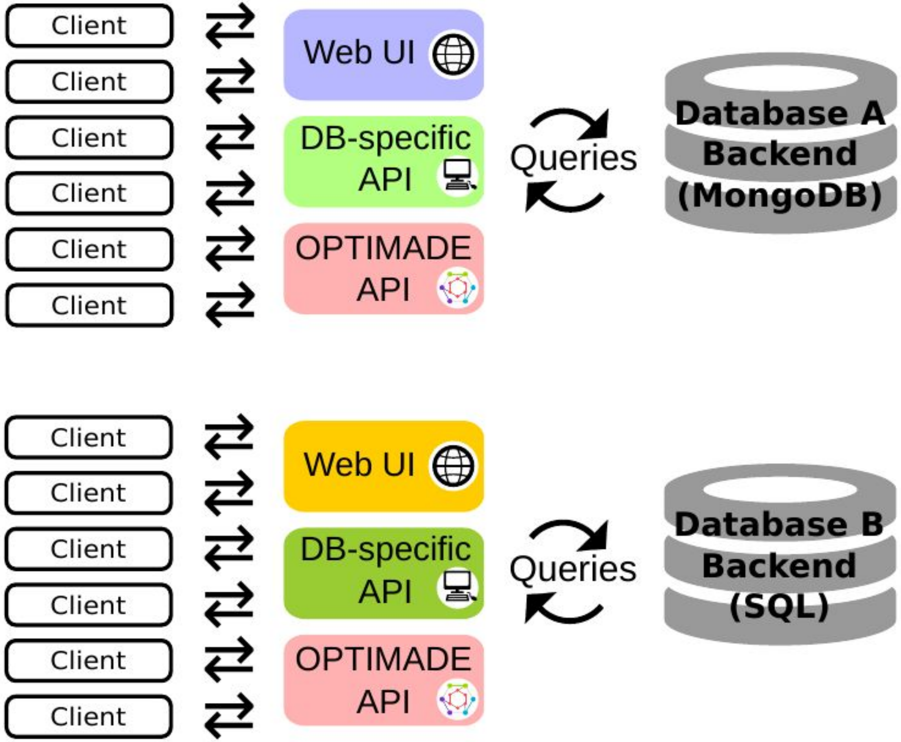
- **Application Programming Interfaces (APIs) your code call someone else's code**
  - They aim to expose functionality not implementation
  - An API specification defines a set of expectations/promises about what happens when a particular method is called
  - Examples:
    - `scikit_learn`
      - `numpy`
        - `BLAS`

- **Web-based APIs let you (or your machine) call someone else's code on THEIR computer**
  - Well-defined interfaces that allow for interactions between remote systems without user intervention
  - Examples: online payments, OAuth, database queries, offloading intensive operations to the cloud, e.g., face recognition, translation…

Jargon: APIs, HTTP, REST and JSON:API

- **Web-based APIs are typically accessed via HTTP requests**
  - Defines a set of methods ("verbs"), e.g., `GET`, `POST`, `PUT`, `DELETE` that a server can respond to
  - Accessible via URLs in the browser, or with tools like curl, wget
  - Examples:
    - `GET /images/logo.png HTTP/1.1`
    -
      ```
      POST /test HTTP/1.1
      Host: foo.example
      Content-Type:
      application/x-www-form-urlencoded
      Content-Length: 27
      field1=value1&field2=value2
      ```

- **REST APIs (Representational State Transfer)**
  - A style of "**stateless**" web API that uses HTTP methods to manipulate "**resources**" on the server
  - OPTIMADE is based on a specific flavour of REST APIs called **JSON:API**
  - Can write API specification itself in a machine-readable way using e.g., **OpenAPI** and **JSONSchema**



Client

Client

Client

Client

Client

Client

Web UI

DB-specific API

OPTIMADE API

Queries

Database A Backend (MongoDB)

Client

Client

Client

Client

Client

Client

Web UI

DB-specific API

OPTIMADE API

Queries

Database B Backend (SQL)

The OPTIMADE API Format

OPTIMADE 1.0 (released July 2020)

- **REST API specification for common access to crystal structure databases**
  - Human-readable specification (~20k words)
  - Based on **JSON:API**
  - Machine-readable **OpenAPI 3.0** schema
- **Enables unified access to 26M crystal structures from 19 providers***
  - Federated providers list and discoverability mechanisms
  - API validation beyond OpenAPI through associated tooling

- **Features:**
  - Defines **resource types** for crystal structures and bibliographic references
  - Well-defined grammar for **`?filter=`** language to enable multi-provider queries
  - Introspective **`/info`** endpoint for extensibility
  - API version negotiation
  - Strict response format, but data models are flexible where necessary
  - Hopefully not “`just another format`”

https://github.com/Materials-Consortia/OPTIMADE

Response format and entry types

- **Follow JSON:API v1 (https://jsonapi.org) for all entry types.**
  - Anti-bikeshedding tool with its own registered MIME type

    Every response is broken down into `data`, `links`, `included` and `meta`

    Every member of `data` response is further broken down into: `type`, `id`, `attributes`, `relationships`

- **OPTIMADE entry types:**
  - No attached semantics, simply defines a data representation
  - `/structures`:
    - Crystal structure with only *one* required field, `structure_features`
    - Many “**SHOULD**” level fields, as you might expect.
  - `/references`:
    - Bibliographic entry type that follows the BibJSON specification.

Querying and filtering

- **Each entry type can be filtered at the entry listing endpoints**:
  - Single entry: `/<entry_type>/<id>`
    - e.g., `/structures/12345`
  - Multi-entry: `/<entry_type>?filter=<query>`
    - e.g., `/structures?filter=nelements=2`

- **Filtering uses a custom grammar written for OPTIMADE**:
  - You can find examples in the specification, our paper and in the exercises
  - Need to be careful encoding the filter as a URL, depending on the tool you are using:
    - e.g., no spaces allowed, quotes must be escaped and so on.
  - Filters should be **transferable** between databases, and will hopefully become more powerful going forward

Auxiliary endpoints and discoverability

- `/links`
  - Link to other implementations or arbitrary web resources
- `/info`
  - Introspective endpoint for provider-specific metadata, schemas and contact details
  - Crucial for extensibility without requiring consensus
- `/versions`
  - Version negotiation, for generating version section of URLs, e.g. `example.org/v1/structures`

- **Index meta-databases:**
  - Serves links to child databases of a given provider, rather than data itself
  - Can be registered or even hosted by the OPTIMADE “federation”.

Questions?

Then live demo

- Finding the list of OPTIMADE APIs:
  - on the dashboard
- Explore some OPTIMADE APIs:
  - in the browser,
  - from the command-line,
  - with Python.

Links visited in demo:

- https://www.optimade.org
- https://www.optimade.org/providers-dashboard
- https://optimade.odbx.science/v1/info
- https://optimade.odbx.science/v1/structures
- https://optimade.odbx.science/v1/structures?filter=elements HAS “Li”
- https://optimade.materialsproject.org/v1/structures?filter=elements HAS “Li”

Common pitfalls

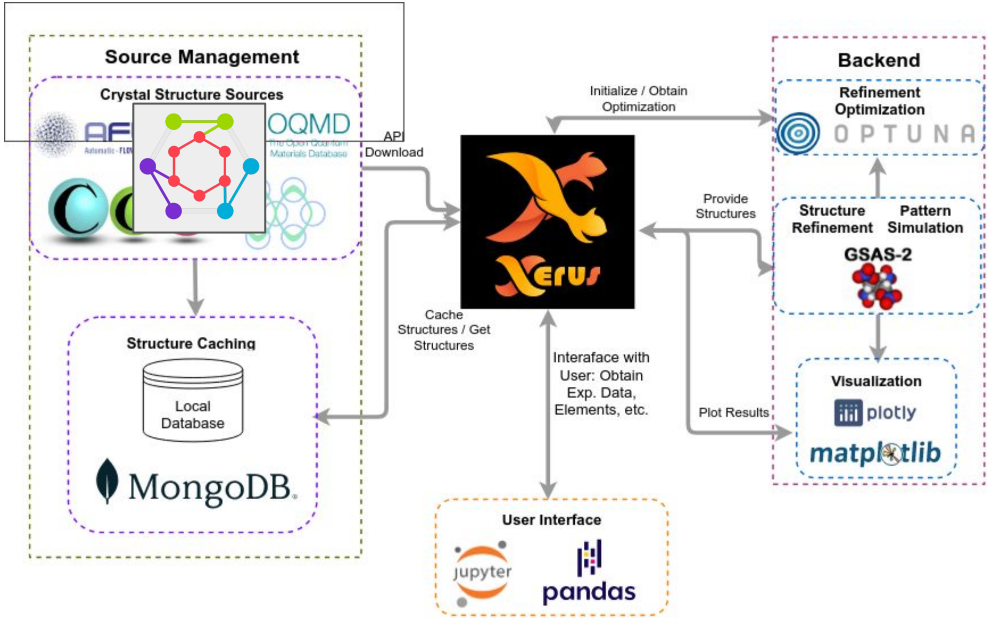
- Trying just the base URL: e.g., `https://optimade.odbx.science`
  - Need to add a version/endpoint, as the base URL can serve anything (or nothing!)
- Visiting the wrong base URL, e.g., the one used for indexing
  - Some providers serve multiple databases, and the index lists them all.
  - Need to make sure you are querying the correct base URL!
- URL encoding: spaces and quotes need to be “escaped”/encoded within some tools otherwise you will get errors
  - e.g., using Python’s `urllib.parse.quote`
- APIs should be really, really boring

Applications of OPTIMADE

- What would you do with unified access to the largest dataset of hypothetical inorganic crystal structures?
- Field-dependent but driven by the participating databases:
    - Surfaces, Interfaces and amorphous cells are missing
    - Molecular crystals, MOFs

Materials Discovery

- Why only filter on known materials?
- Case study: Xerus
  - An automated XRD refinement program that operates out of crystal structure databases
  - Cheap to screen XRD patterns, filtering enables precision in which structures are actually included
  - Automated screening against tens of millions of hypothetical structures
- If you have a database of things you think exist, but struggle to get them in front of experimentalists
- Ongoing implementations in FullProf and others



ADVANCED THEORY AND SIMULATIONS

https://github.com/pedrobcst/Xerus

Baptista de Castro *et al.*, "XERUS: An Open-Source Tool for Quick XRD Phase Identification and Refinement Automation" 10.1002/adts.202100588

Materials Discovery-ability (discoverability)

- Federated databases encourages early and piecemeal sharing
- Barrier to hosting data and getting an audience is drastically lowered
- Material proposed: DFT files archived in NOMAD/Materials Cloud can be referred to but not easily **discovered**
- Now, with OPTIMADE API, anyone filtering for specific chemical spaces will see such results
- Small group/project-specific databases of very few structures
  - Can accelerate ingestion into larger curated sets
  - Whilst retaining enhanced discoverability
  - Baked data: static datasets that are filterable

Pitfalls:

- GIGO
- Analysis methods will need to accommodate **multiple fidelities**, high uncertainty entries
- OPTIMADE API format needs to keep up with such use cases,
  - e.g., new `structure_origin` field will allow filtering between hypothetical, experimentally reported and theoretically unstable structures as reported by the database

Materials Design

- Re-funnelling high-throughput workflows with new hypothetical materials
- Making use of all hypothetical materials:
  - Exploring the feature space of 26M structures is difficult!
  - Active learning could help decide which compositions to sample
- Growing all the time:
  - Can we make our workflows adaptable to this?
  - Can filter by date across all databases and have a living HT workflow that continually screens new materials

optimade-python-tools

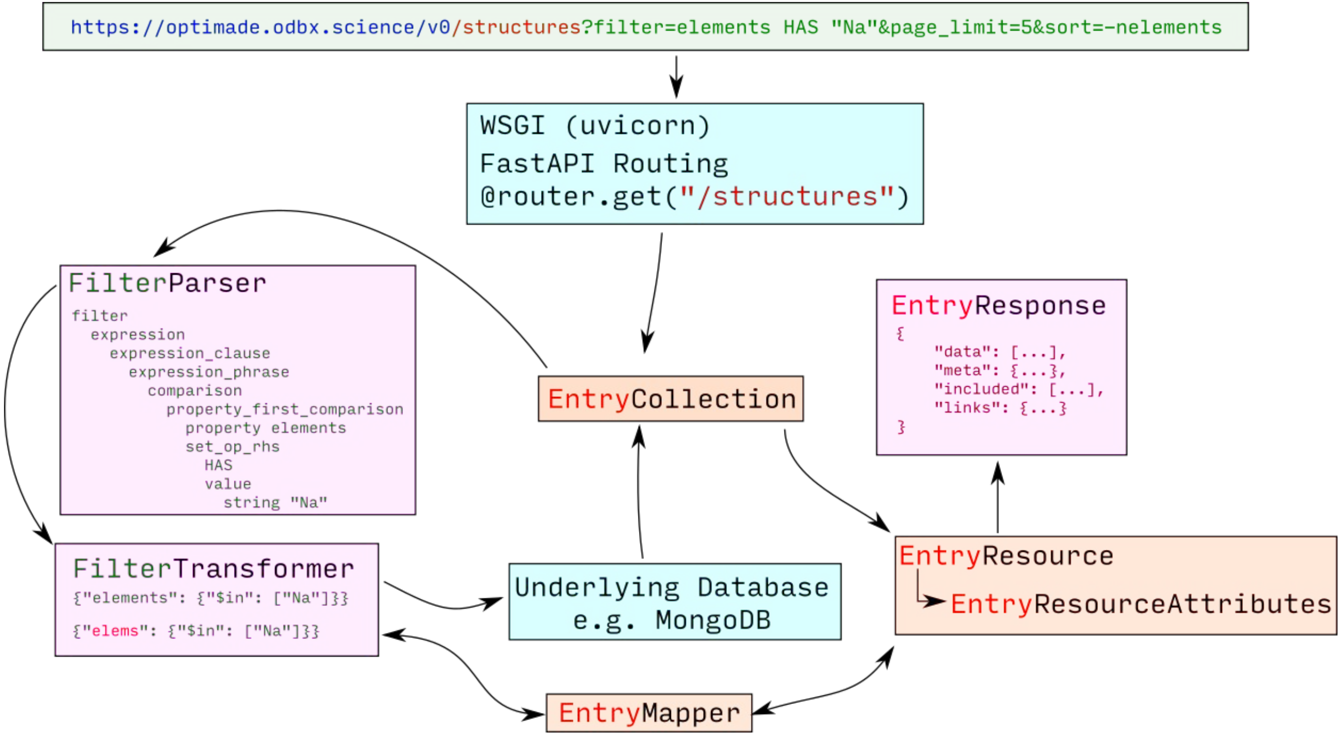
An open source Python package for consuming and implementing OPTIMADE APIs.

Evans, M. L., Andersen, C. W. A. *et al.*,
Journal of Open Source Software, 6(65), 3458, 2021
10.21105/joss.03458

Materials-Consortia/optimade-python-tools

- ❖ **Spin-up OPTIMADE API with “no code”**
  - ➢ Built with pydantic and FastAPI
  - ➢ Annotated data models with data validation
  - ➢ Auto-generated OpenAPI 3.0 and JSONSchema
  - ➢ EBNF grammar implementation with filter transformers for MongoDB & Elasticsearch
  - ➢ Mappers between existing formats (ASE, pymatgen, CIF) and OPTIMADE, supporting aliases etc.
- ❖ **Client for asynchronously querying multiple databases**
- ❖ **Used by Materials Project, NOMAD, *odbx*, 2DMatPedia and Materials Cloud**
- ❖ **Provides tools for validating remote implementations**

optimade-python-tools: what makes an OPTIMADE API?



```
https://optimade.odbx.science/v0/structures?filter=elements HAS "Na"&page_limit=5&sort=-nelements
```

WSGI (uvicorn)
FastAPI Routing
`@router.get("/structures")`

EntryCollection

FilterParser

```
filter
  expression
    expression_clause
      expression_phrase
        comparison
          property_first_comparison
            property elements
            set_op_rhs
              HAS
              value
                string "Na"
```

FilterTransformer

```
{"elements": {"$in": ["Na"]}}
{"elems": {"$in": ["Na"]}}
```

Underlying Database
e.g. MongoDB

EntryMapper

EntryResource
↳ EntryResourceAttributes

EntryResponse

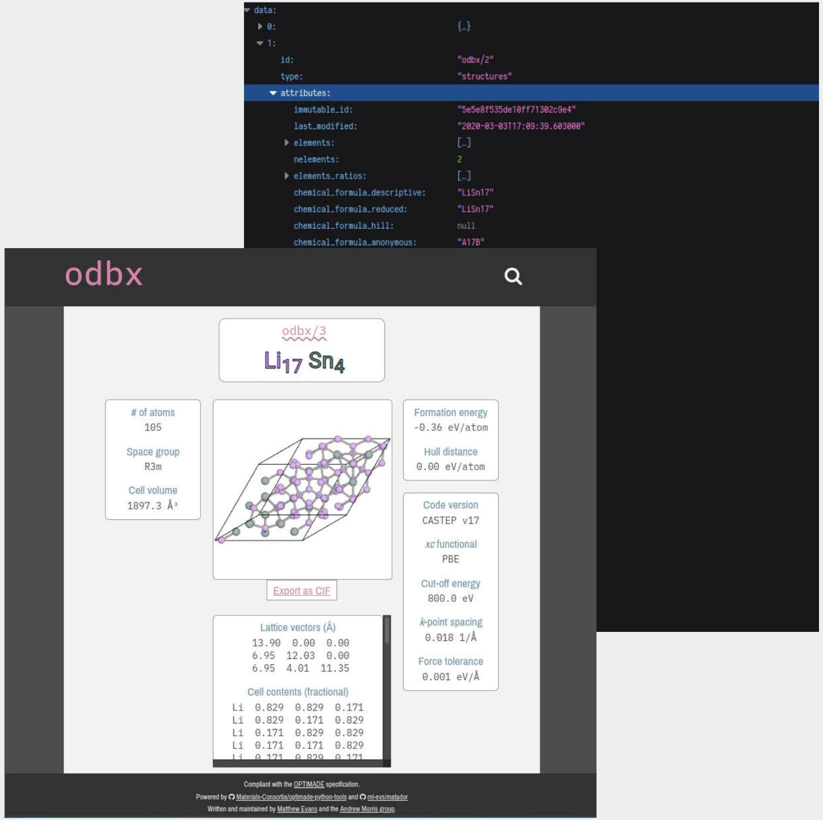
```
{
    "data": [...],
    "meta": {...},
    "included": [...],
    "links": {...}
}
```

odbx

Open Database of Xtals

https://odbx.science

- Andrew Morris Group (ajm143.github.io) at University of Birmingham/University of Cambridge
- HT crystal structure prediction for battery materials, encapsulated nanowires and MOFs.
- Underlying database (MongoDB) with ~1M geometry optimisations
    - ~50 M force/energy configurations)
    - Focussed on ~30 binary and ternary phase diagrams
    - Constructed with matador library (https://github.com/ml-evs/matador)



```
▾ data:
  ▸ 0:                                {…}
  ▾ 1:
      id:                             "odbx/2"
      type:                           "structures"
    ▾ attributes:
        immutable_id:                 "5e5e8f535de10ff71302c9e4"
        last_modified:                "2020-03-03T17:09:39.603000"
      ▸ elements:                     […]
        nelements:                    2
      ▸ elements_ratios:              […]
        chemical_formula_descriptive: "LiSn17"
        chemical_formula_reduced:     "LiSn17"
        chemical_formula_hill:        null
        chemical_formula_anonymous:   "A17B"
```

odbx/3

Li_{17} Sn_4

# of atoms
105

Space group
R3m

Cell volume
1897.3 Å³

Formation energy
-0.36 eV/atom

Hull distance
0.00 eV/atom

Code version
CASTEP v17

xc functional
PBE

Cut-off energy
800.0 eV

k-point spacing
0.018 1/Å

Force tolerance
0.001 eV/Å

Export as CIF

Lattice vectors (Å)

```
13.90   0.00   0.00
 6.95  12.03   0.00
 6.95   4.01  11.35
```

Cell contents (fractional)

```
Li  0.829  0.829  0.171
Li  0.829  0.171  0.829
Li  0.171  0.829  0.829
Li  0.171  0.171  0.829
Li  0.171  0.829  0.171
```

Compliant with the OPTIMADE specification.
Powered by Materials-Consortia/optimade-python-tools and ml-evs/matador
Written and maintained by Matthew Evans and the Andrew Morris group.

- An OPTIMADE-first database
- `optimade-python-tools` runs REST API
  - https://optimade.odbx.science
  - https://odbx.science
  - https://github.com/ml-evs/odbx.science
- Currently a very reduced set: ~50 “polished” structures
- Provider-specific fields bundled as extensions to Python models:
  - energies/derived energies (e.g. distance from hull), forces
  - DFT parameters
- Also ingested some relevant materials discovery datasets

An example implementation: *odbx*

Materials-Consortia/optimade-python-tools

ml-evs/odbx.science

https://optimade.odbx.science

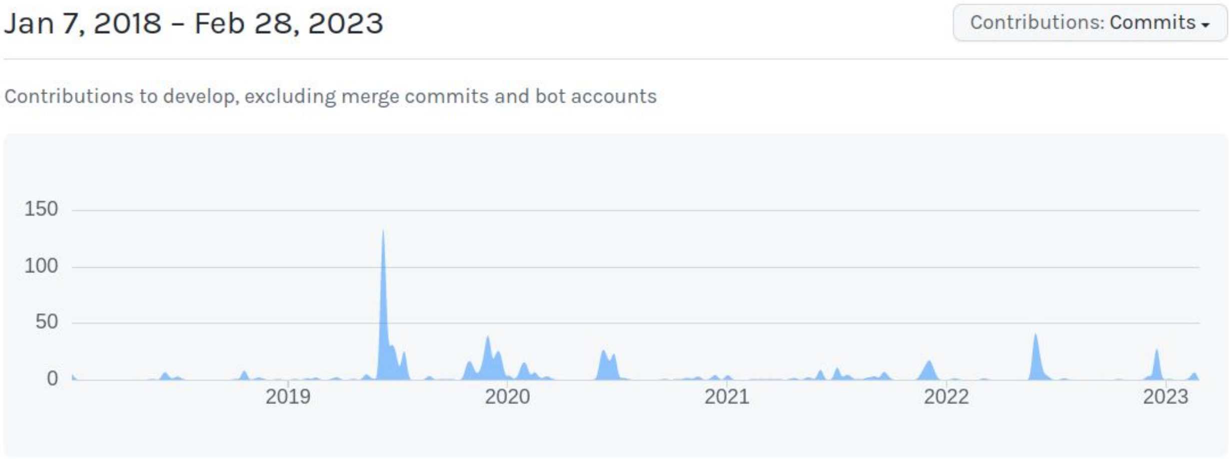
https://odbx.science

What's next for OPTIMADE?

What's next for OPTIMADE?

- Enhanced property standardization
- New fields driven by filtering use cases
  - e.g., domain (biomolecular) or topology (adatom on surface, clusters)
  - Symmetry (space groups, point groups)
- Working towards semantic interoperability

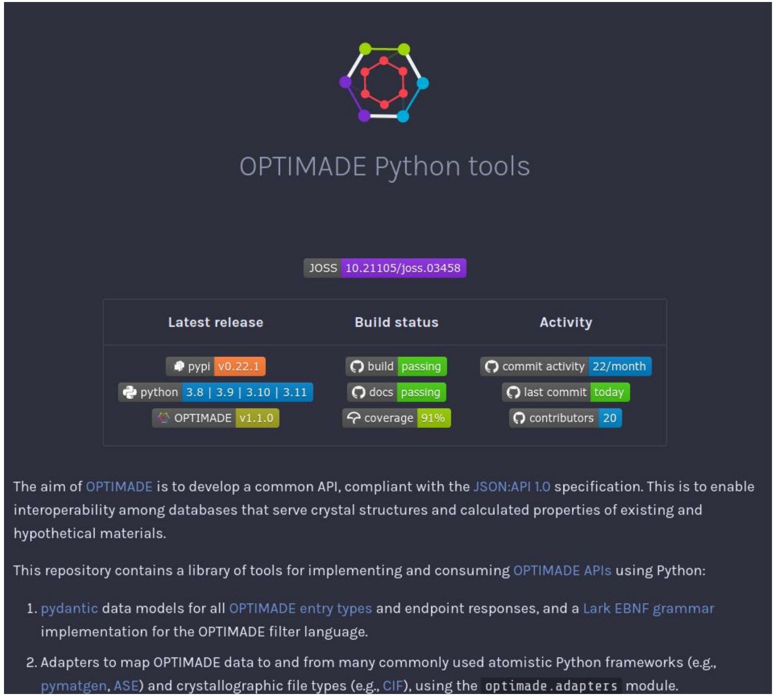
- Trajectories and collections
  - Molecular dynamics, phonons, Monte Carlo
- Federated property namespaces,
  - Sharing e.g. "`_dft_`" properties
- General use scientific API standard



Jan 7, 2018 – Feb 28, 2023

Contributions to develop, excluding merge commits and bot accounts

What's next for OPTIMADE?



OPTIMADE Python tools

JOSS 10.21105/joss.03458

Latest release	Build status	Activity
pypi v0.22.1	build passing	commit activity 22/month
python 3.8 | 3.9 | 3.10 | 3.11	docs passing	last commit today
OPTIMADE v1.1.0	coverage 91%	contributors 20

The aim of OPTIMADE is to develop a common API, compliant with the JSON:API 1.0 specification. This is to enable interoperability among databases that serve crystal structures and calculated properties of existing and hypothetical materials.

This repository contains a library of tools for implementing and consuming OPTIMADE APIs using Python:

1. pydantic data models for all OPTIMADE entry types and endpoint responses, and a Lark EBNF grammar implementation for the OPTIMADE filter language.
2. Adapters to map OPTIMADE data to and from many commonly used atomistic Python frameworks (e.g., pymatgen, ASE) and crystallographic file types (e.g., CIF), using the `optimade.adapters` module.

- **Improved tooling and applications:**
  - ➢ Continued development of `optimade-python-tools`
  - ➢ Support more backends, e.g., SQL
  - ➢ Enhance library functionality for creating apps
  - ➢ "**Baked data**": Serving APIs from static/archived data
- **More data (your data?)**
  - ➢ Big ML-derived datasets (30M structures) in the pipeline
  - ➢ Constantly badgering people
  - ➢ Enhanced integrations with user-driven platforms like Materials Cloud

Links

- https://optimade.org: **monthly virtual meetings**
- https://optimade.org/optimade-python-tools: **always in need of maintainers**
- https://github.com/Materials-Consortia
- Forum: https://matsci.org/c/optimade
- Papers:
  - “OPTIMADE, an API for exchanging materials data”, Andersen *et al.*, Scientific Data (2021) arXiv:2103.02068
  - “optimade-python-tools: a Python library for serving and consuming materials data via OPTIMADE APIs” Evans *et al.*, Journal of Open Source Software (2021) 10.21105/joss.03458
- Validation:
  - https://www.optimade.org/providers-dashboard/
  - https://github.com/Materials-Consortia/optimade-validator-action
- Web clients with rich interfaces:
  - https://optimadeclient.materialscloud.io/
  - https://optimade.science

Acknowledgements

CENTRE EUROPÉEN DE CALCUL ATOMIQUE ET MOLÉCULAIRE
cecam
SINCE 1969

COD

OQMD
The Open Quantum Materials Database

NOMAD
NOVEL MATERIALS DISCOVERY

MATERIALSCLOUD

The Materials Project

AFLOW
Automatic - FLOW for Materials Discovery

httk

Casper W. Andersen[1,†], Rickard Armiento[2,†], Evgeny Blokhin[3,4,†], Gareth J. Conduit[5,†], Shyam Dwaraknath[6,†], Matthew L. Evans[5,7,†], Ádám Fekete[8,7,†], Abhijith Gopakumar[9,†], Saulius Gražulis[10,11,†], Andrius Merkys[10,†], Fawzi Mohamed[12,†], Corey Oses[13,14,†], Giovanni Pizzi[1,†], Gian-Marco Rignanese[7,†], Markus Scheidgen[12,15,†], Leopold Talirz[1,16,†], Cormac Toher[13,14,†], Donald Winston[6,†], Rossella Aversa[17,18], Kamal Choudhary[19], Pauline Colinet[13,14], Stefano Curtarolo[13,14], Davide Di Stefano[20], Claudia Draxl[15], Suleyman Er[21], Marco Esters[13,14], Marco Fornari[22,13], Matteo Giantomassi[7], Marco Govoni[23], Geoffroy Hautier[7], Vinay Hegde[9], Matthew K. Horton[6], Patrick Huck[6], Georg Huhs[15], Jens Hummelshøj[24], Ankit Kariryaa[25], Boris Kozinsky[26,27], Snehal Kumbhar[1], Mohan Liu[9], Nicola Marzari[1], Andrew J. Morris[28], Arash A. Mostofi[29], Kristin A. Persson[6,30], Guido Petretto[7], Thomas Purcell[12], Francesco Ricci[7], Frisco Rose[13,14], Matthias Scheffler[12], Daniel Speckhard[12,15], Martin Uhrin[1], Antanas Vaitkus[10], Pierre Villars[4], David Waroquiers[7], Chris Wolverton[9], Michael Wu[6], and Xiaoyu Yang[31]

Johan Bergsma

Exercises

- 4 general exercises
  - Familiarise yourself with the format with any tool you like
- 2 tool-specific exercises
  - `pymatgen`
  - `optimade-python-tools`
- 2 database-specific exercises
  - AFLOW: Crystallographic prototypes and properties
  - OQMD: Simple ML example
- Optional extensions:
  - Constructing secondary databases with `optimade-python-tools`
  - Hosting an OPTIMADE API with optimade-python-tools

Exercises on GitHub

Materials-Consortia/optimade-tutorial-exercises

- Can ask questions:
  - Here, over lunch, over drinks
  - on the forums (https://matsci.org/c/optimade)
  - over email
- The best practice will be trying to integrate OPTIMADE into your own research!
- Please give us feedback on what you think works/doesn’t work!

Thank you for listening!