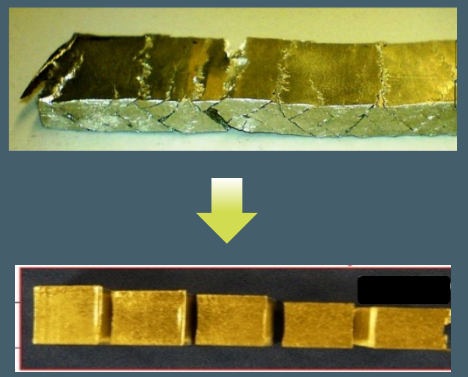
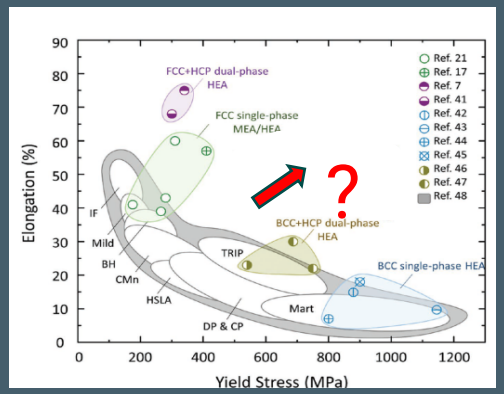
MAX-PLANCK-INSTITUT
FÜR EISENFORSCHUNG GmbH

Enabling next-generation materials science simulations by automated workflows

Jan Janssen, Tilmann Hickel, and Jörg Neugebauer

MPI für Eisenforschung
Department on Computational Materials Design

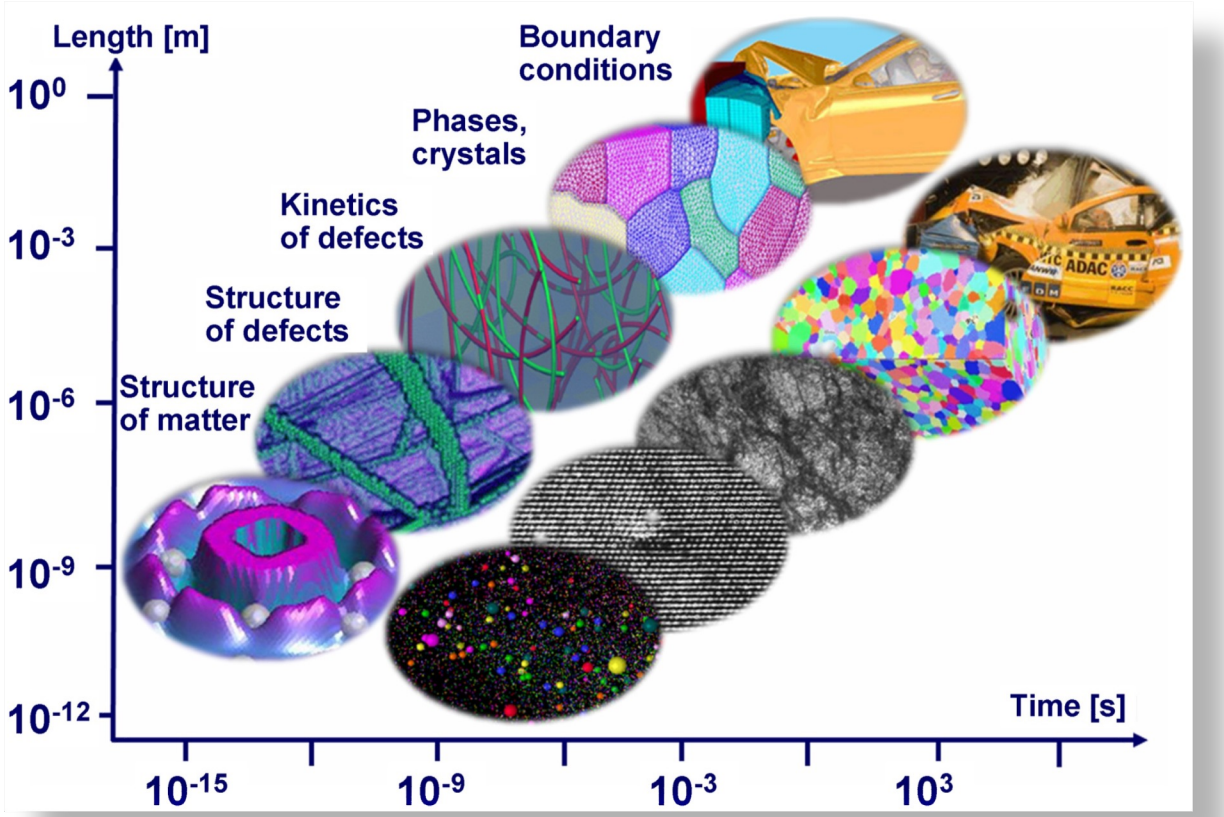




pyron

Multiscale challenge in materials modeling and experiment

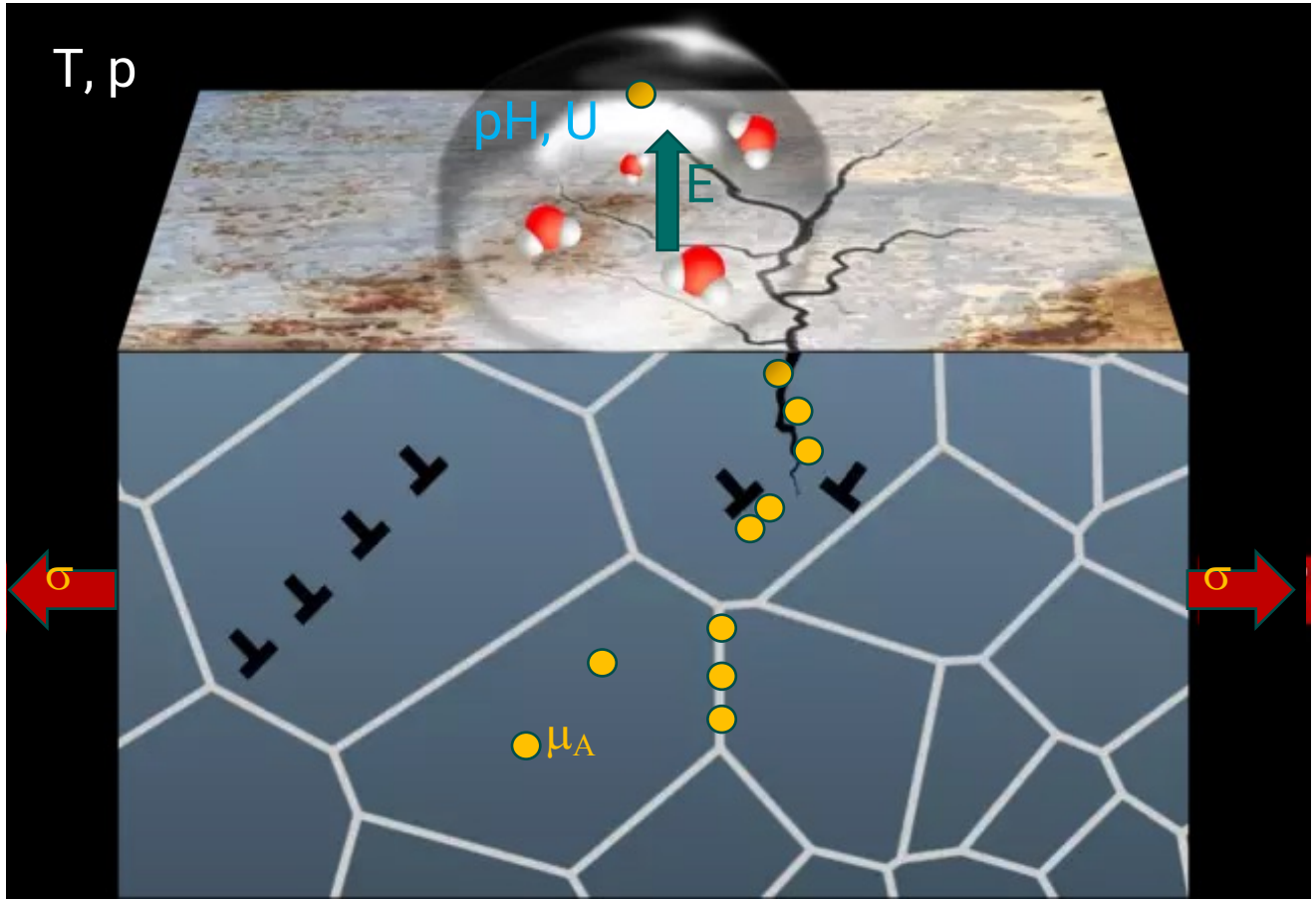
Inherent multiscale character of materials



Multiscale & multiscale simulations often require complex simulation workflows and large-scale simulations

- **But: Tools have been independently developed and are not interoperable**

Challenge: Multiscale and Multiphysics Character



Real Environments

Thermodynamics

- **T, p, μ**

Electrochemical

- **pH, U, E_{field}**

Mechanical

- **σ**

...

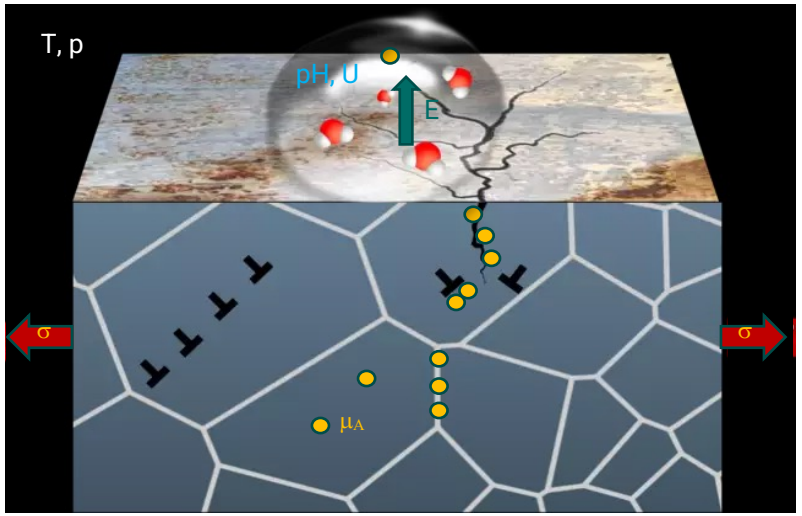
Virtual Materials Design

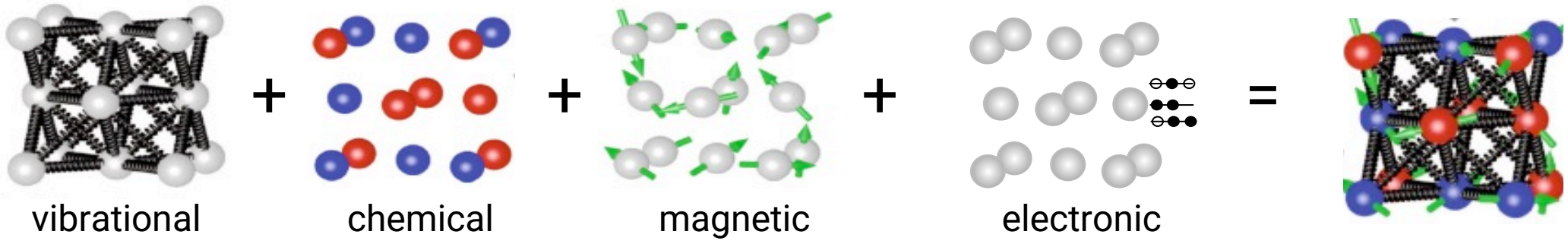
- Predict response of material to thermodynamic, electrochemical, mechanical environments

Methodological challenges

Exploring and navigating high-dimensional configuration spaces

$$Z(V,T,x,\alpha_{phase}) = \left\langle e^{-E^{QM}\left(\left\{\vec{R}_I, Z_I, \sigma_I, f_i, \ldots\right\}_V\right)/k_B T} \right\rangle_{V,T,x,\alpha_{phase}}$$

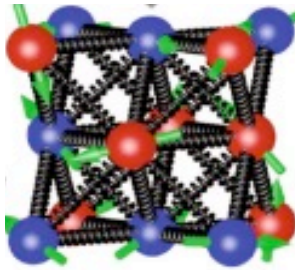




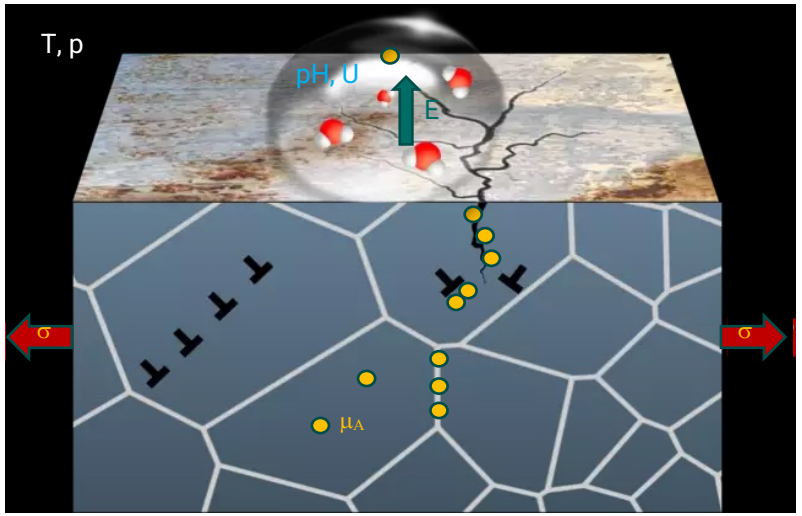
Methodological challenges

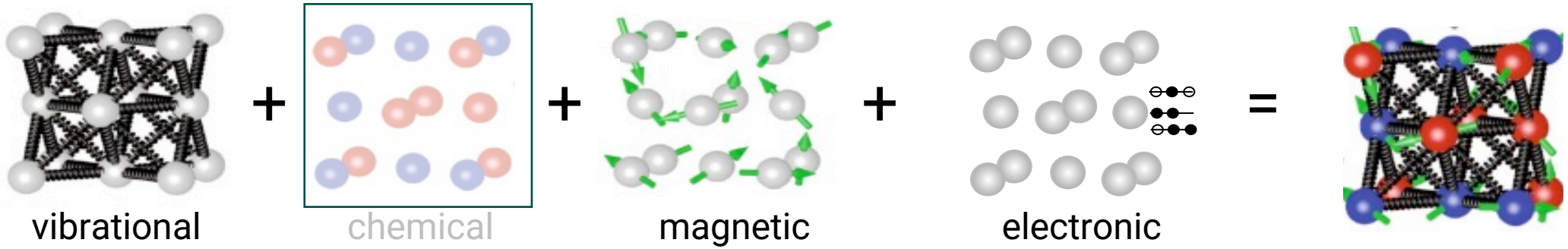
Exploring and navigating high-dimensional configuration spaces

- Thermal complexity



coupling



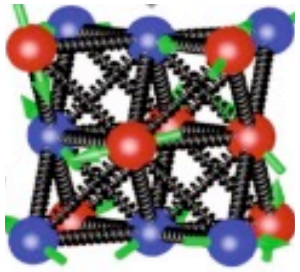


vibrational chemical magnetic electronic

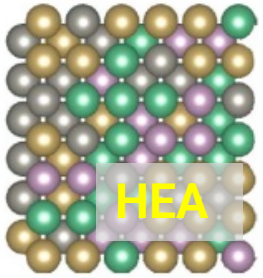
Methodological challenges

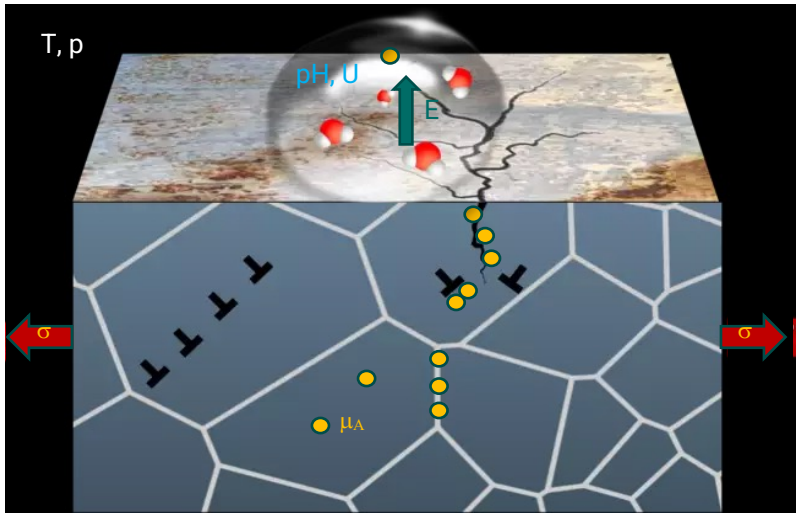
Exploring and navigating high-dimensional configuration spaces

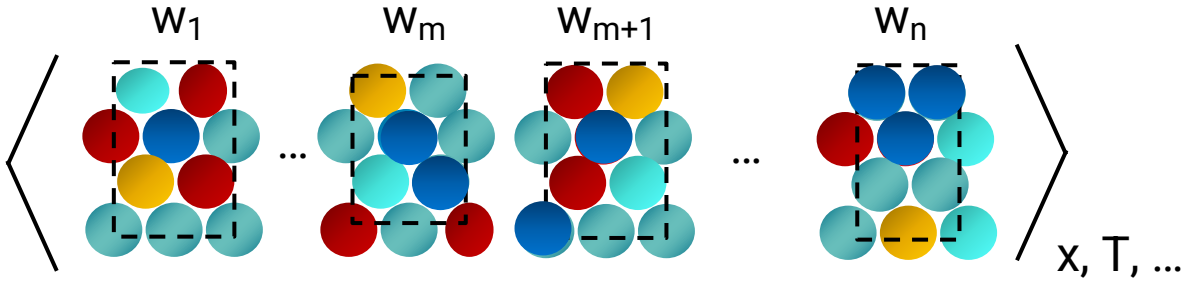
- Thermal complexity



- Chemical complexity



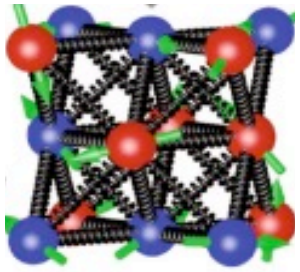




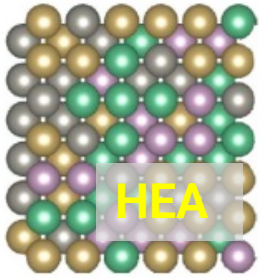
Methodological challenges

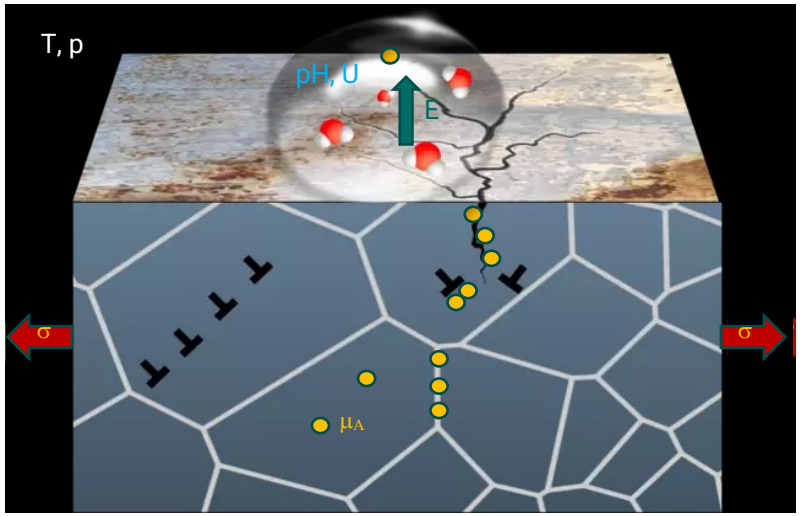
Exploring and navigating high-dimensional configuration spaces

- Thermal complexity

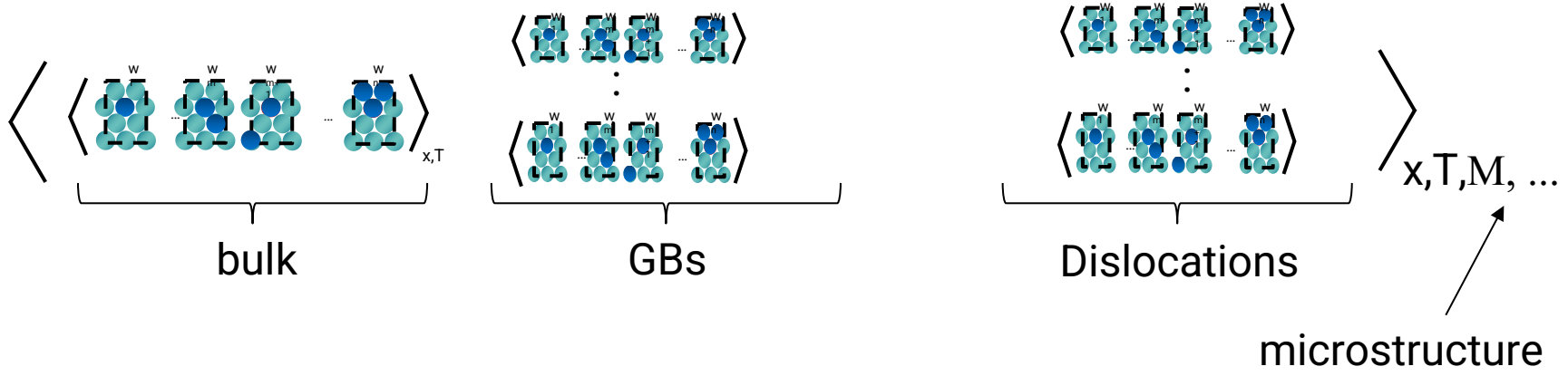


- Chemical complexity





- Structural complexity



Key Challenge – Sampling Huge Configuration Spaces

$$Z(V,T,x,\alpha_{phase}) = \left\langle e^{-E^{QM}\left(\left\{\vec{R}_I, Z_I, \sigma_I, f_i, \ldots\right\}_V\right)/k_B T} \right\rangle_{V,T,x,\alpha_{phase}}$$

3N N 3N N

N … Number of atoms

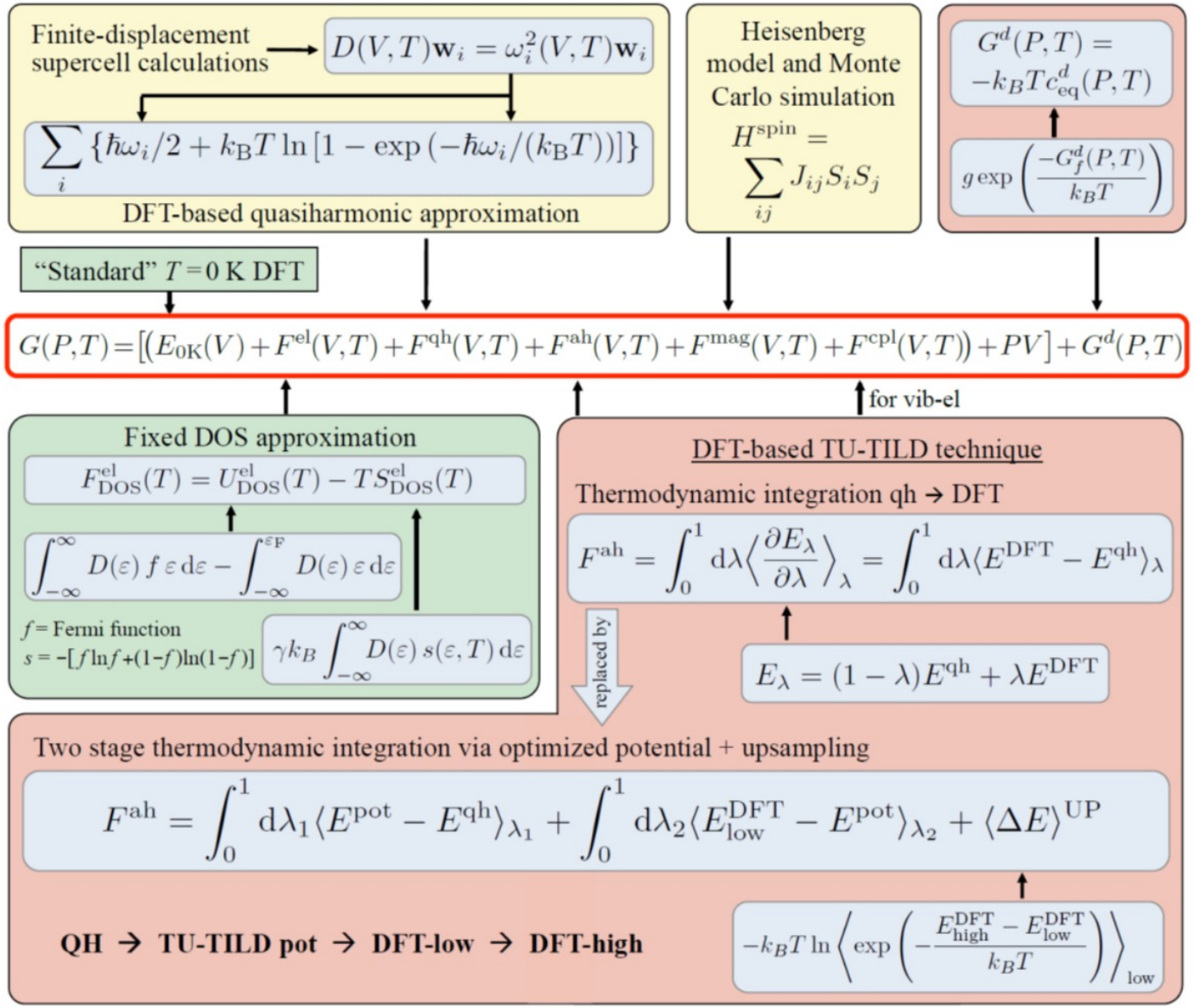
- Exploration of a huge (8N dimensional) configuration space!
  - In principle well-suited for exascale computing
  - But any brute force approach is unfeasible:
    - 10 data points on each coordinate
    - 100 atoms
    - 10^{800} configurations

Dimensionality reduction is mandatory!

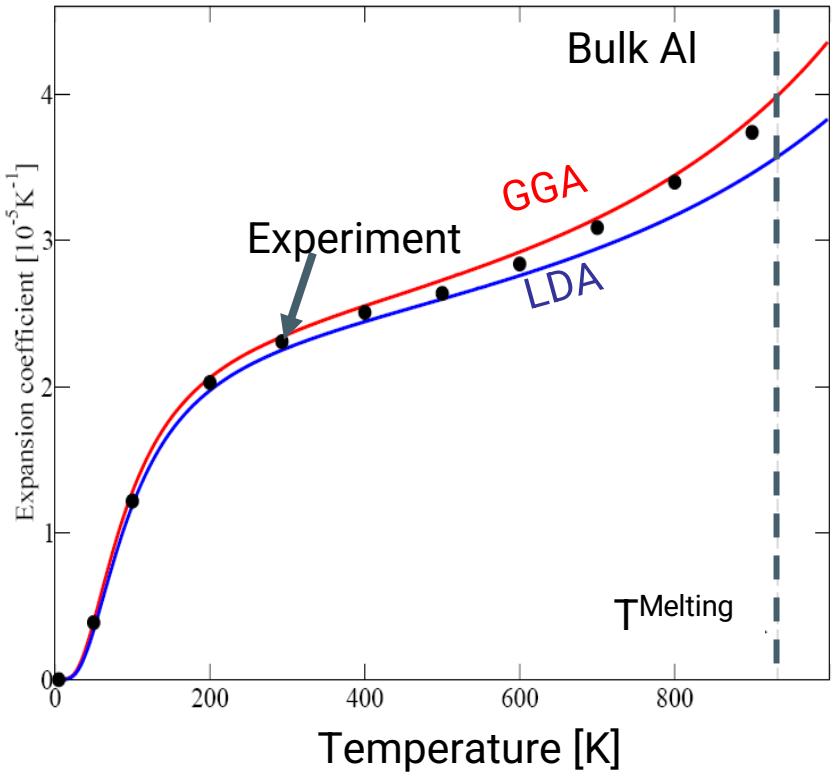
- Math + ML concepts

Practical examples of sampling/navigating in high-dimensional configuration spaces

Ab initio up to the melting temperature



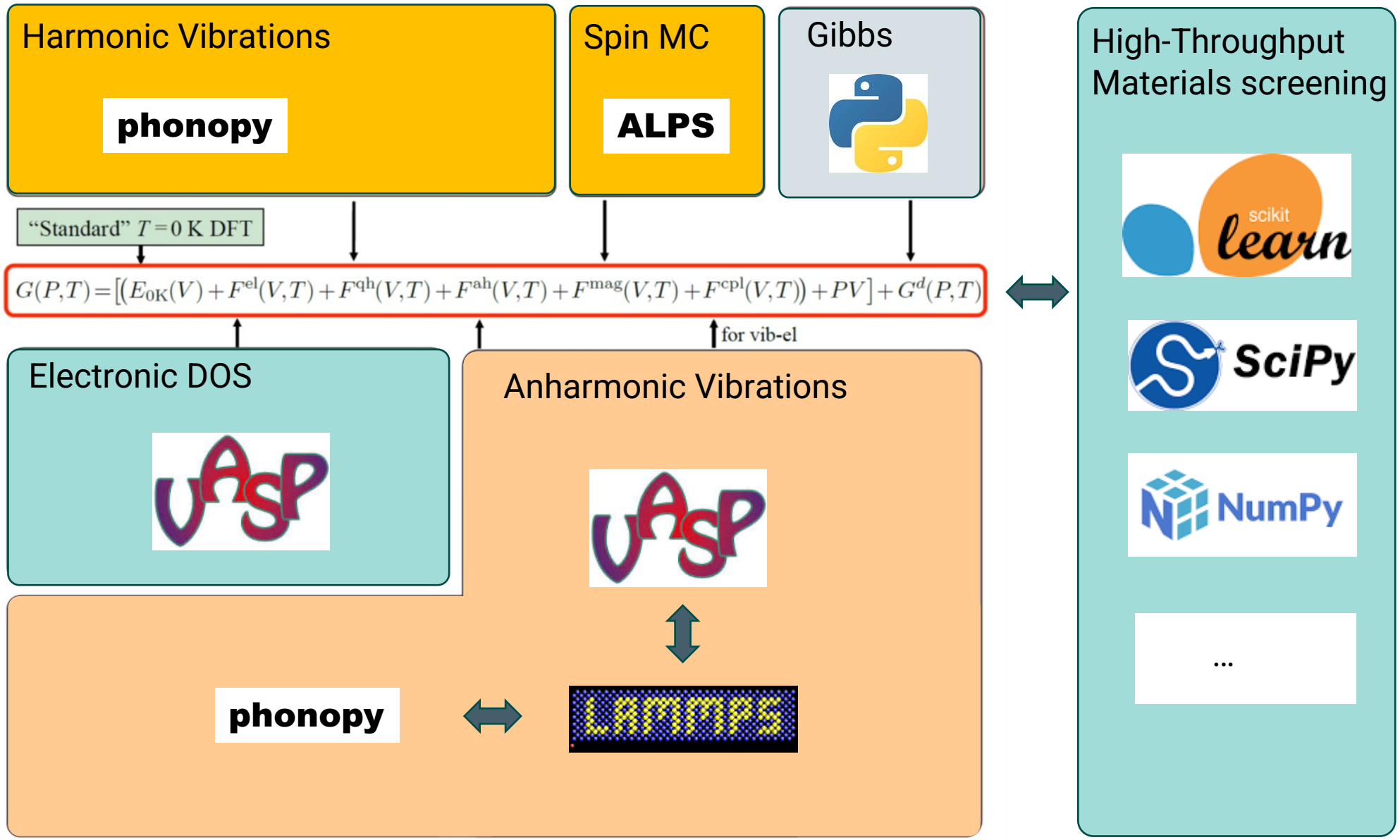
Thermal expansion coefficient vs T



Workflows reduce computational effort from 10^7 down to 10^2 high-quality DFT calculations needed!

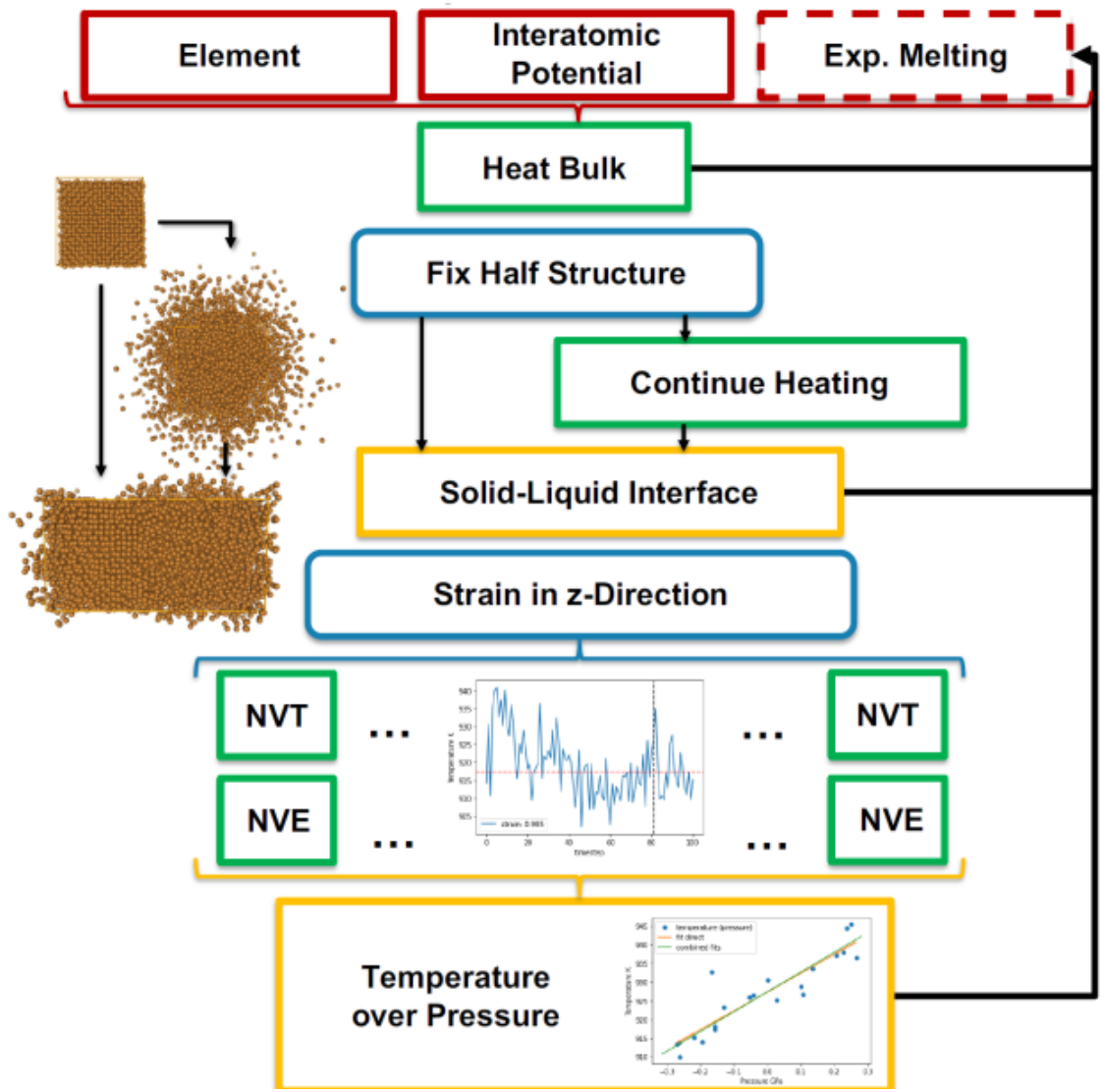
X. Zhang, T. Hickel, B. Grabowski, JN, Comp. Mat. Sci. (2018)

Ab initio up to the melting temperature

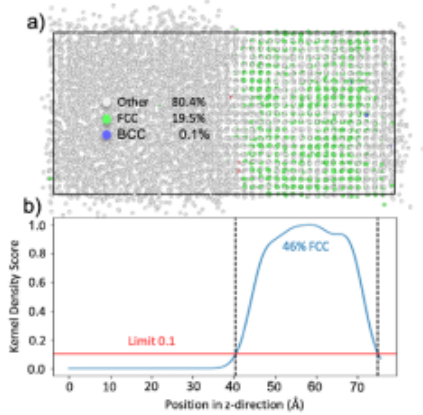


X. Zhang, T. Hickel, B. Grabowski, JN, Comp. Mat. Sci. (2018)

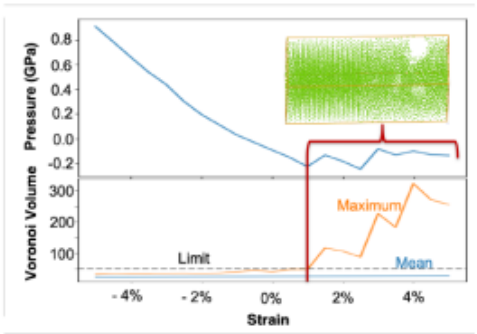
Fully autonomous algorithm to determine melting point



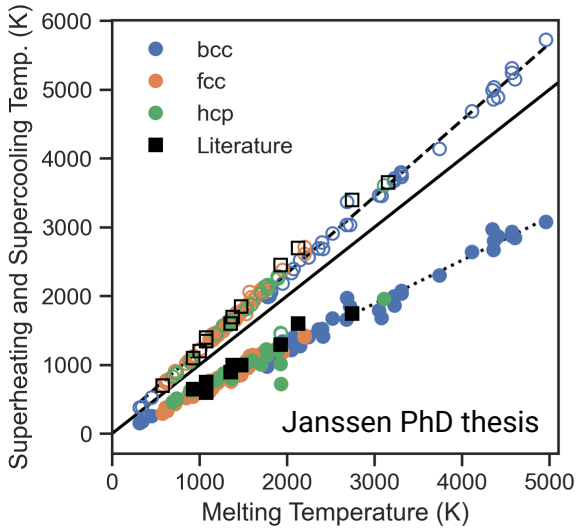
Solid Liquid Detector



Void Detector



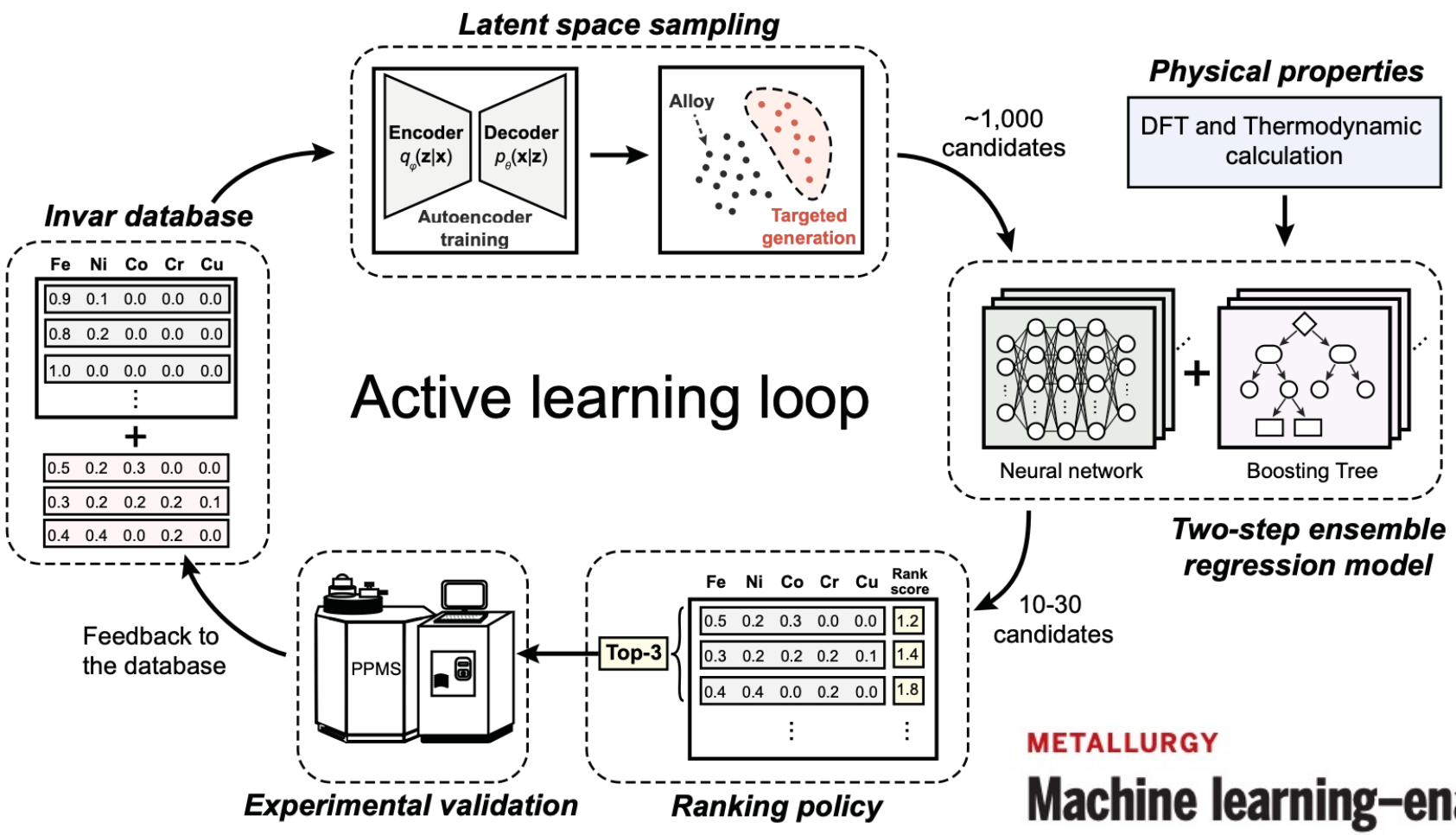
Chemical trends & descriptors



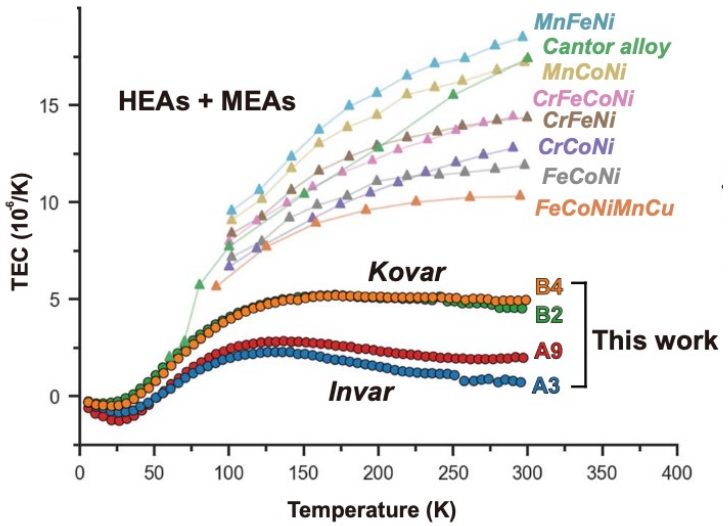
- Overheating/undercooling vs T_m
- > 300 emp. potentials

Zhu, Janssen, Ishibashi, Kormann, Grabowski, JN, Comp. Mat. Sci. 187, 12 (2021)

High-entropy alloy discovery



Experimental validation



Discovery of a new generation of invar alloys

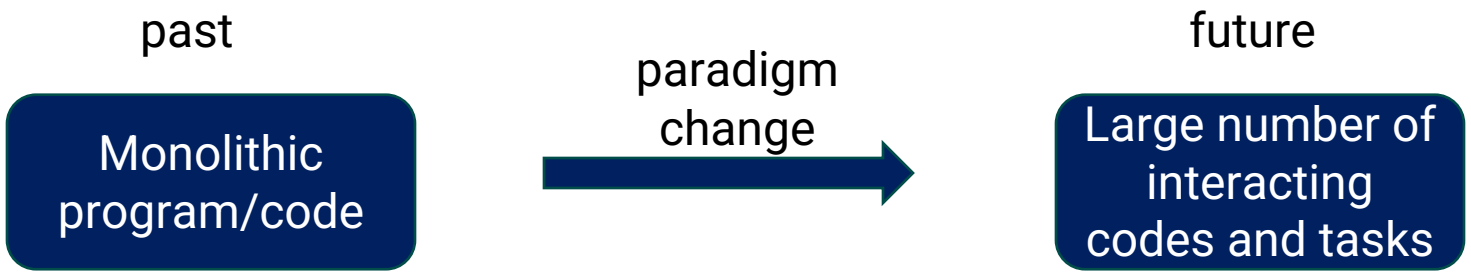
METALLURGY

Rao *et al.*, *Science* **378**, 78–85 (2022)

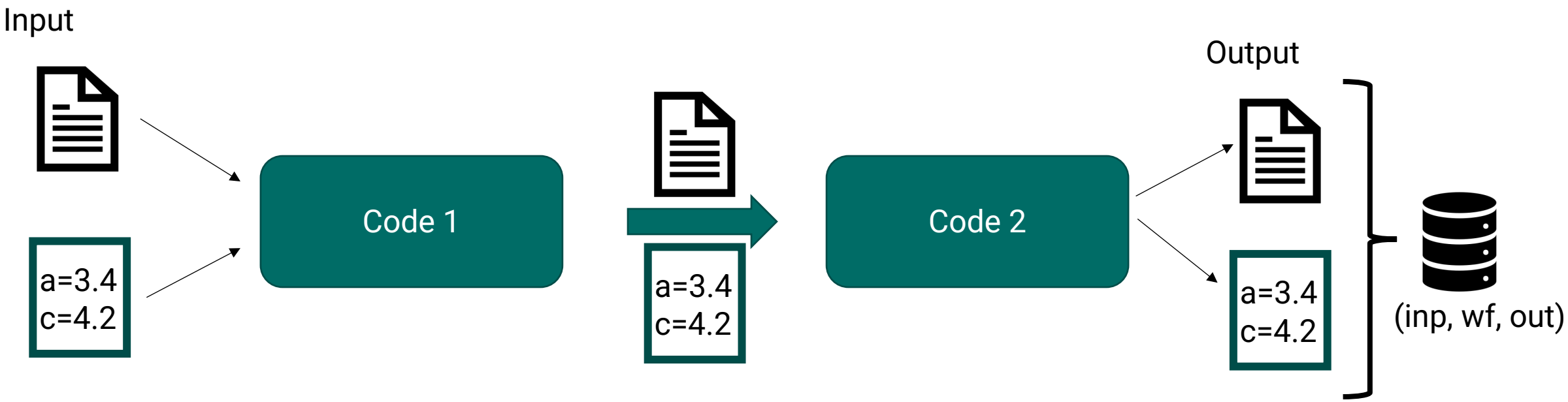
Machine learning–enabled high-entropy alloy discovery

Ziyuan Rao[1], Po-Yen Tung[1,2], Ruiwen Xie[3], Ye Wei[1]*, Hongbin Zhang[3], Alberto Ferrari[4], T.P.C. Klaver[4], Fritz Körmann[1,4], Prithiv Thoudden Sukumar[1], Alisson Kwiatkowski da Silva[1], Yao Chen[1,5], Zhiming Li[1,6], Dirk Ponge[1], Jörg Neugebauer[1], Oliver Gutfleisch[1,3], Stefan Bauer[7], Dierk Raabe[1]*

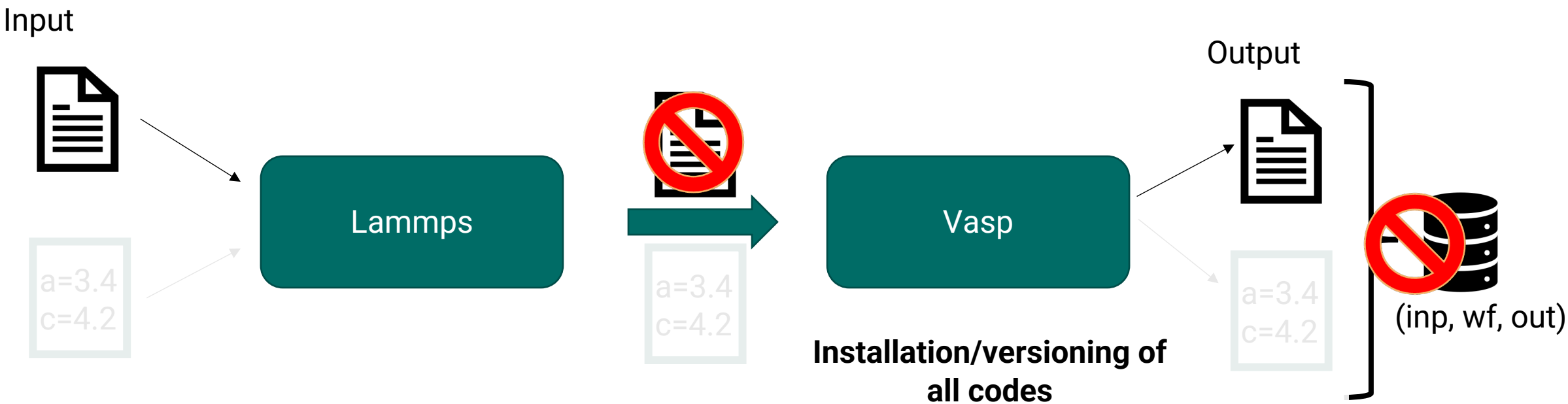
Many of the recent breakthroughs in materials science simulations rely on complex simulation protocols (workflows)



Workflows: Fundamentals



Workflows: Challenges



Code specific input

- Technical expertise for all codes in workflow is mandatory

Lack of interoperability/ interactivity

Code specific output

Manual upload to repository

Libraries that successfully address some of these issues


```
>>> import numpy as np
>>> from scipy.optimize import minimize


>>> def rosen(x):
...     """The Rosenbrock function"""
...     return sum(100.0*(x[1:]-x[:-1]**2.0)**2.0 + (1-x[:-1])**2.0)


>>> x0 = np.array([1.3, 0.7, 0.8, 1.9, 1.2])
>>> res = minimize(rosen, x0, method='nelder-mead',
...                options={'xatol': 1e-8, 'disp': True})
Optimization terminated successfully.
         Current function value: 0.000000
         Iterations: 339
         Function evaluations: 571


>>> print(res.x)
[1. 1. 1. 1. 1.]
```

Libraries that successfully address some of these issues


```
>>> import numpy as np
>>> from scipy.optimize import minimize


>>> def rosen(x):
...     """The Rosenbrock function"""
...     return sum(100.0*(x[1:]-x[:-1]**2.0)**2.0 + (1-x[:-1])**2.0)

>>> x0 = np.array([1.3, 0.7, 0.8, 1.9, 1.2])
>>> res = minimize(rosen, x0, method='nelder-mead',
...                options={'xatol': 1e-8, 'disp': True})
Optimization terminated successfully.
         Current function value: 0.000000
         Iterations: 339
         Function evaluations: 571


>>> print(res.x)
[1. 1. 1. 1. 1.]
```

standard python data types

method selection via single parameter

generic input

generic output

Fully embedded in python!

Translation to materials science simulations

```
>>> import numpy as np
>>> from scipy.optimize import minimize


>>> def rosen(x):
...     """The Rosenbrock function"""
...     return sum(100.0*(x[1:]-x[:-1]**2.0)**2.0 + (1-x[:-1])**2.0)
```

standard python data types

method selection via single parameter

```
>>> x0 = np.array([1.3, 0.7, 0.8, 1.9, 1.2])
>>> res = minimize(structure, method='vasp',
...                energy_conv=1e-6, force_conv=1e-4)
```

generic input

```
>>> res.output.energy_tot
```

generic output

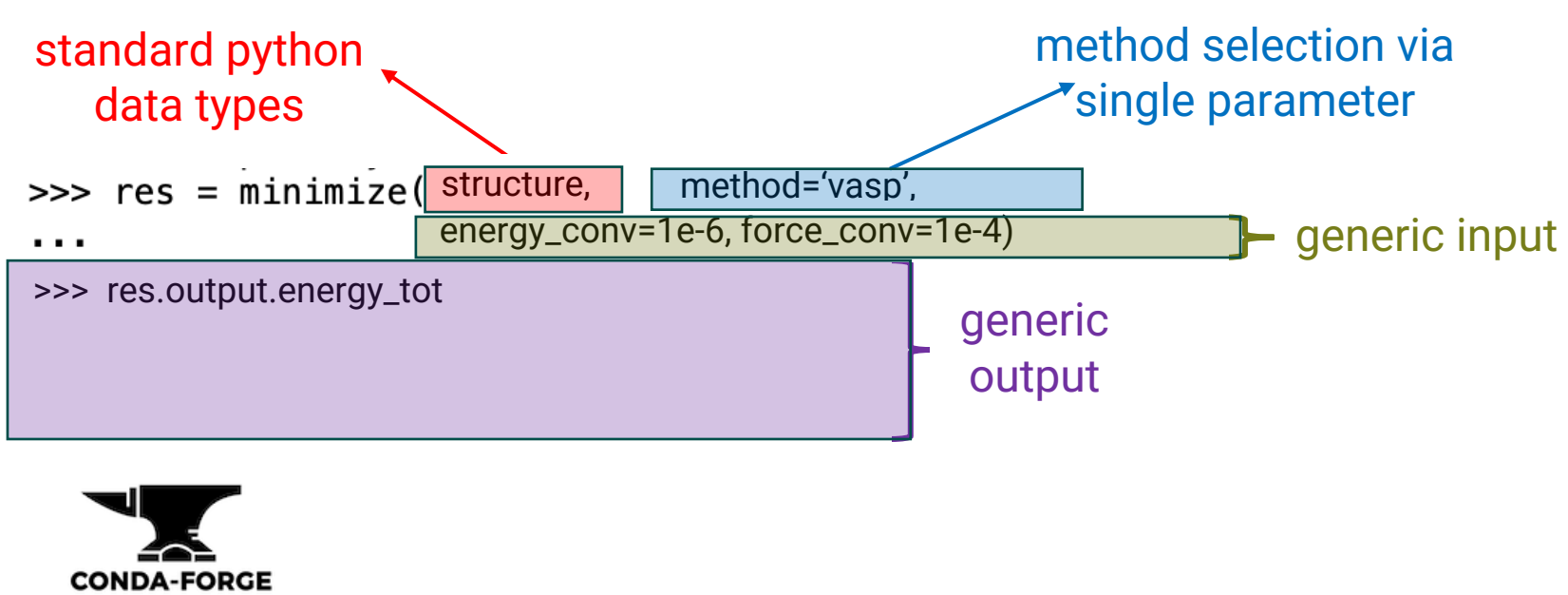
```
>>> print(res.x)
[1. 1. 1. 1. 1.]
```

Fully embedded in python!

Translation to materials science simulations

Features beyond NumPy/SciPy

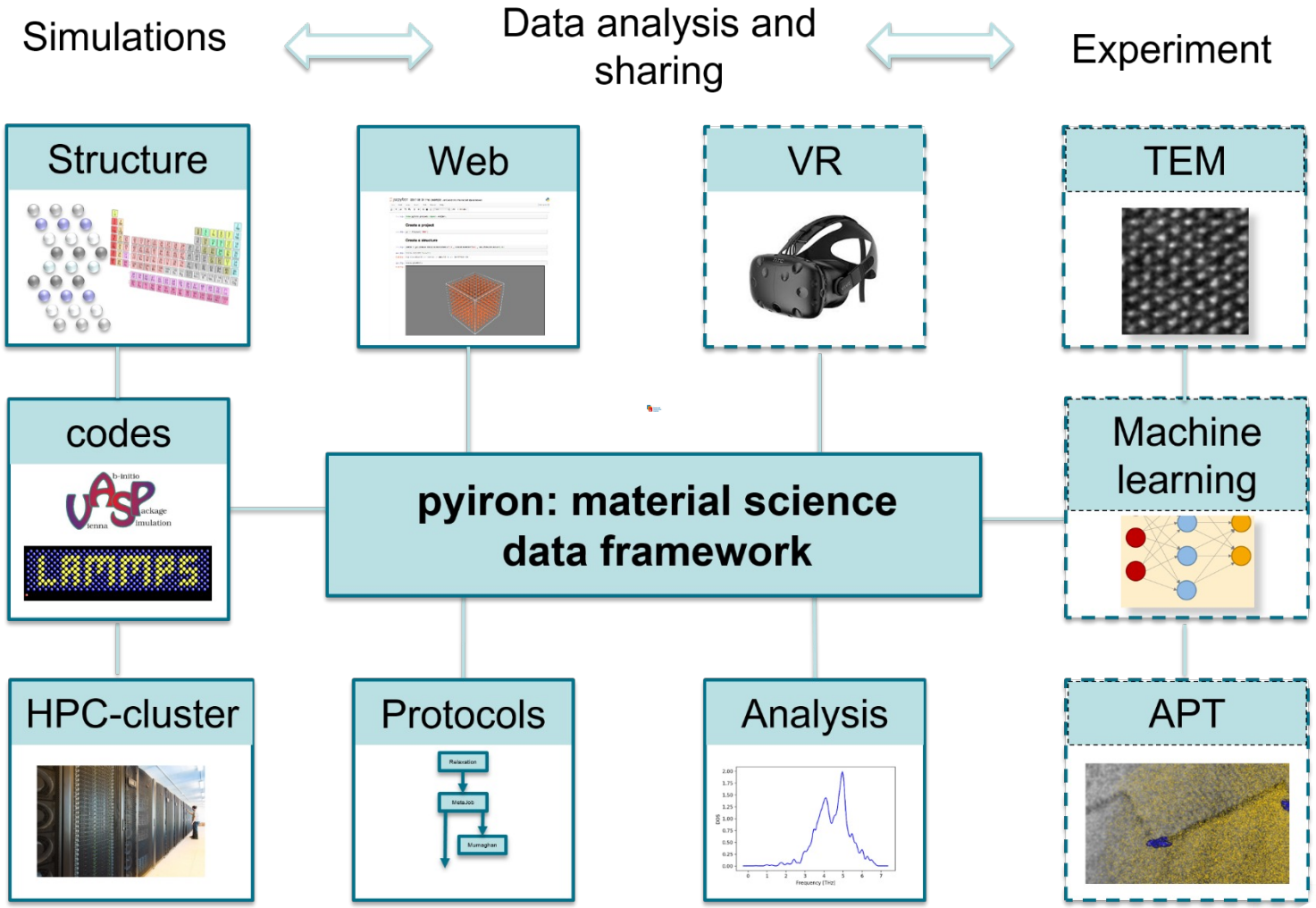
- Automatic upload of input + selected output to database/storage
  - Serialization
- Upscaling to HPC (exascale)
  - Job managment
- Full integration into Jupyter
  - e.g. visualization of complex data structures such as atomic trajectories



Fully embedded in python!

Integrated development environment (IDE) for workflows

Pyiron: Python + Max Planck Institute for Iron research

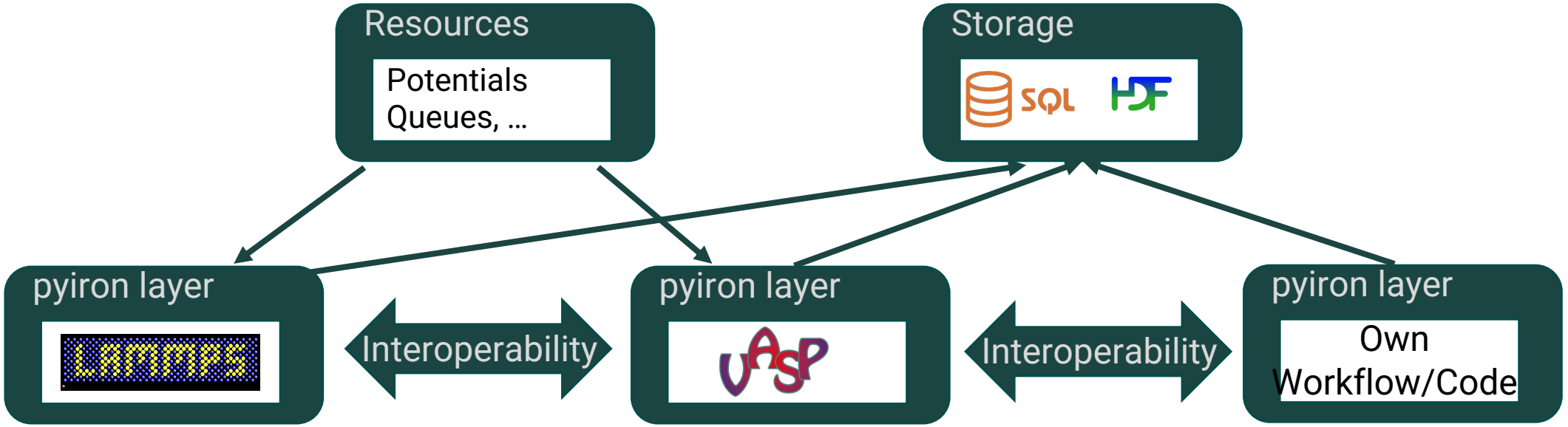


Janssen, Surendralal, Hickel, Todorova, Lysogorskiy, Drautz, JN, Comp.Mat. Sci. **163**, 24 (2019)

www.pyiron.org

Key concept: pyiron objects

Identical generic look and feel

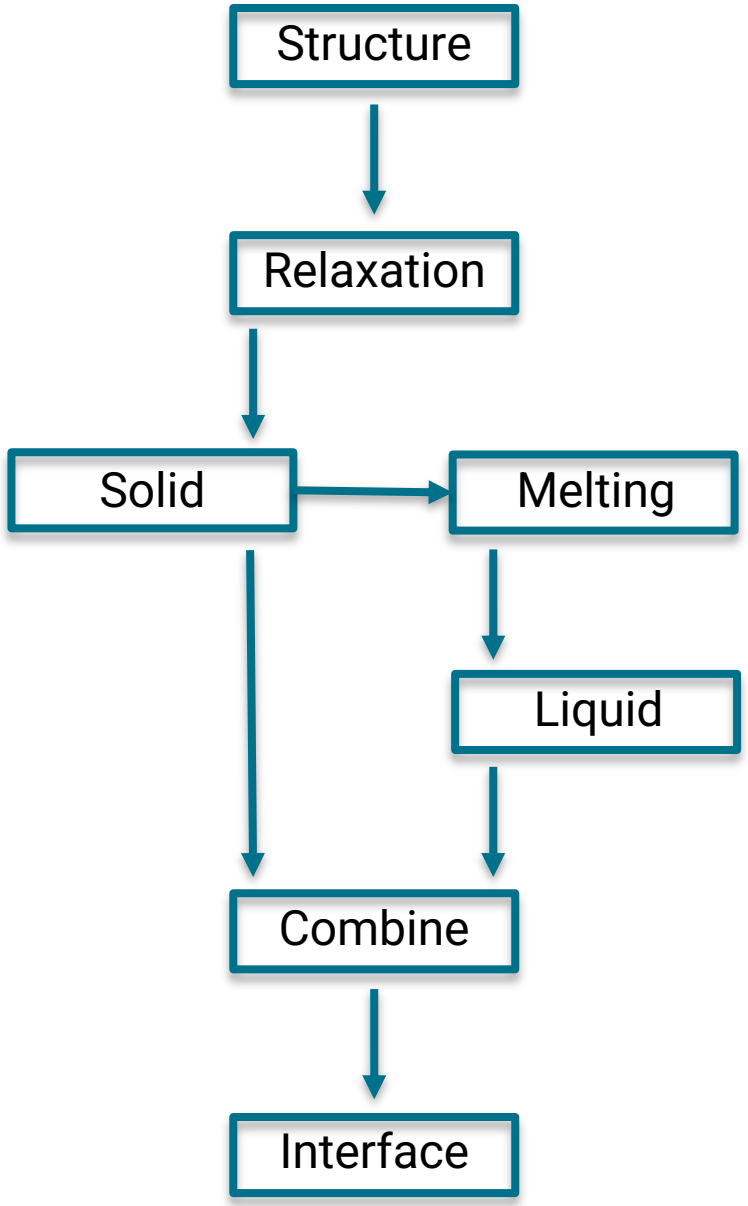


job.structure
job.input
job.output.energy_tot
job.run()

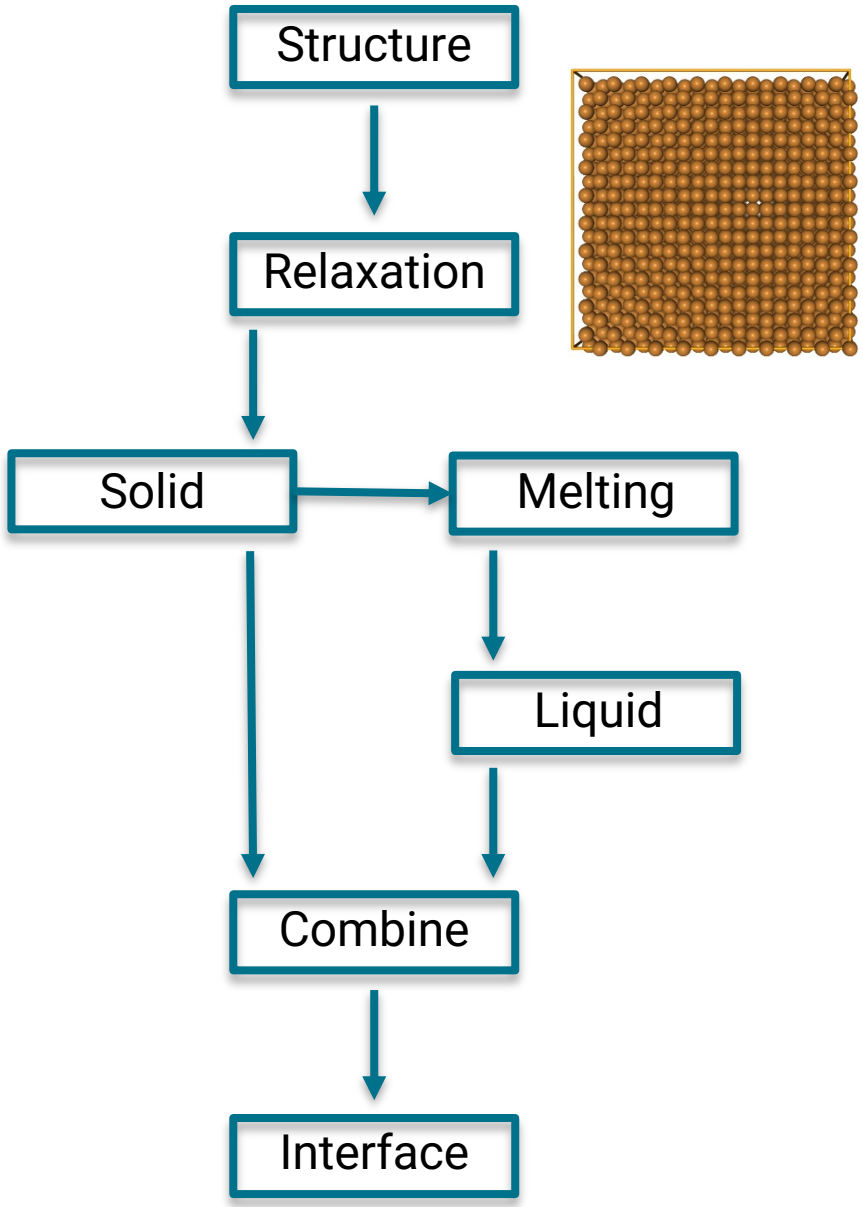
job.structure
job.input
job.output.energy_tot
job.run()

job.structure
job.input
job.output.energy_tot
job.run()

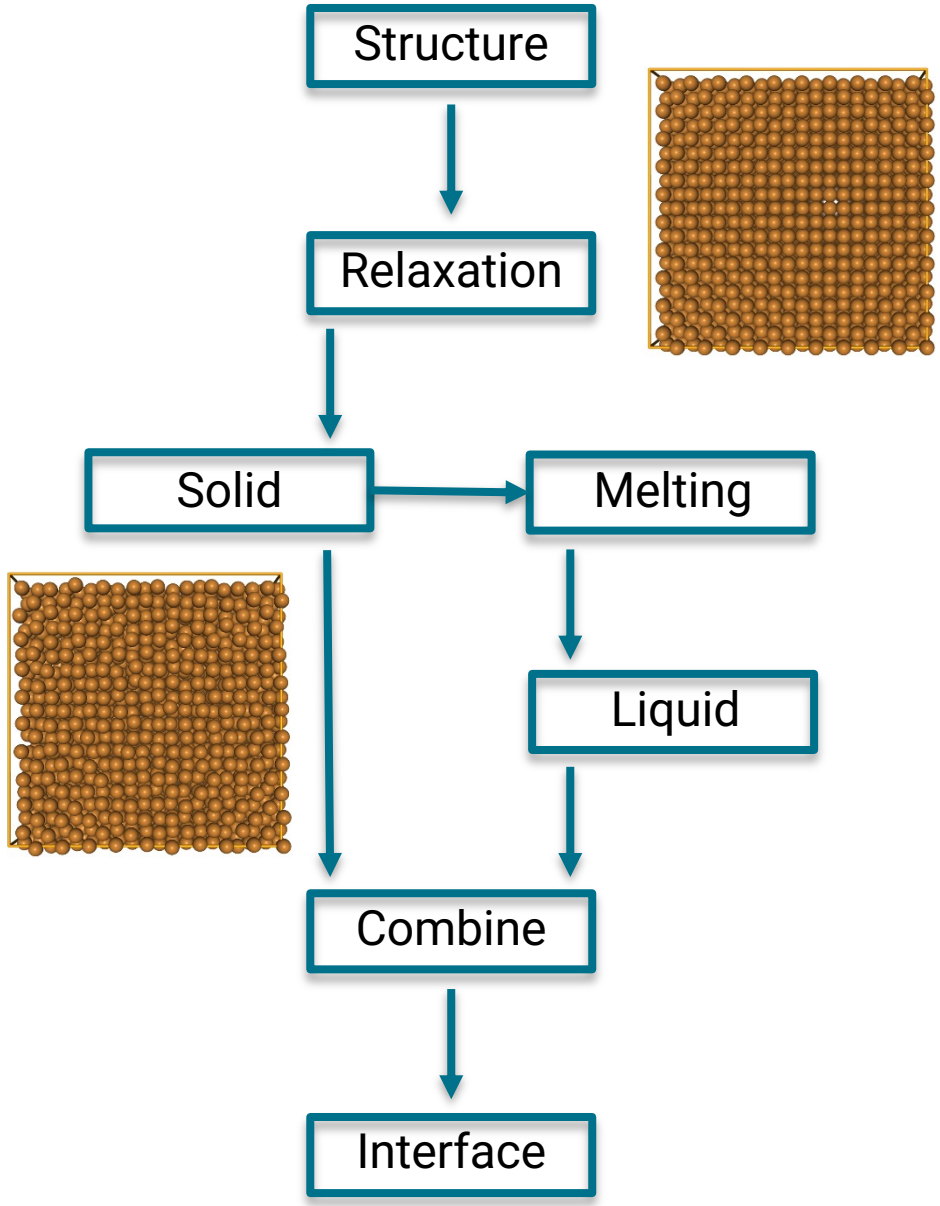
Melting Point calculation



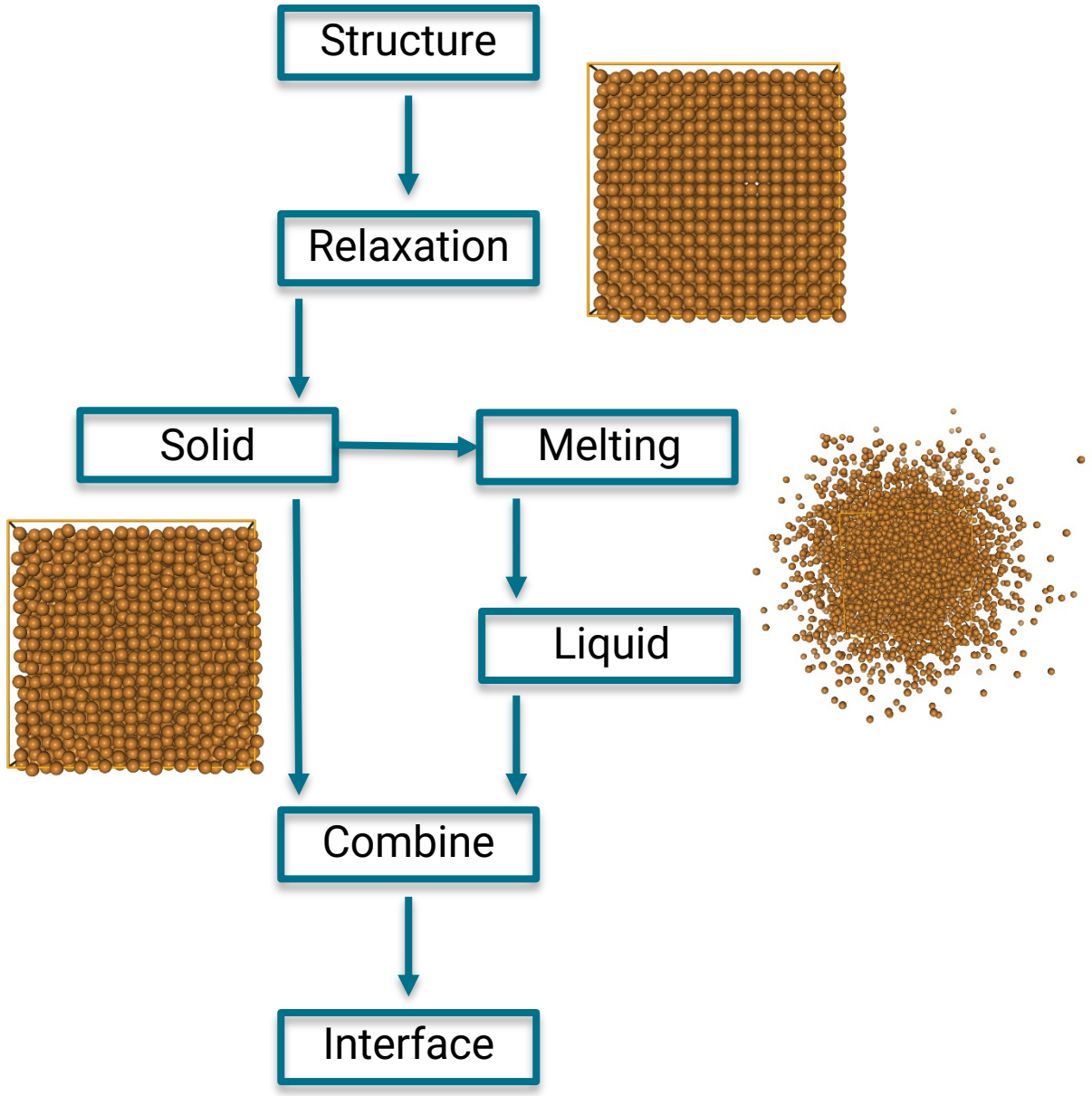
Melting Point calculation



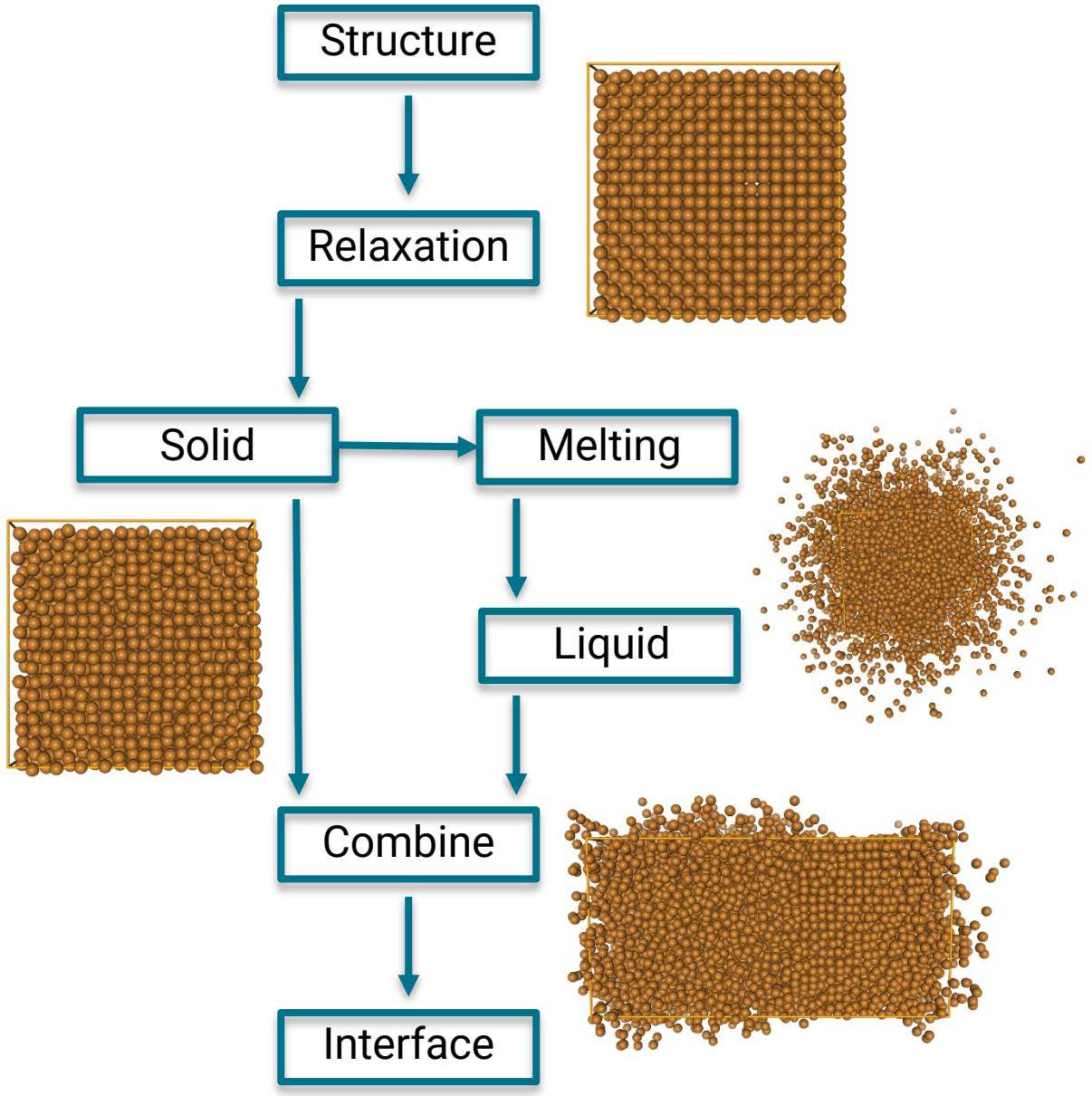
Melting Point calculation



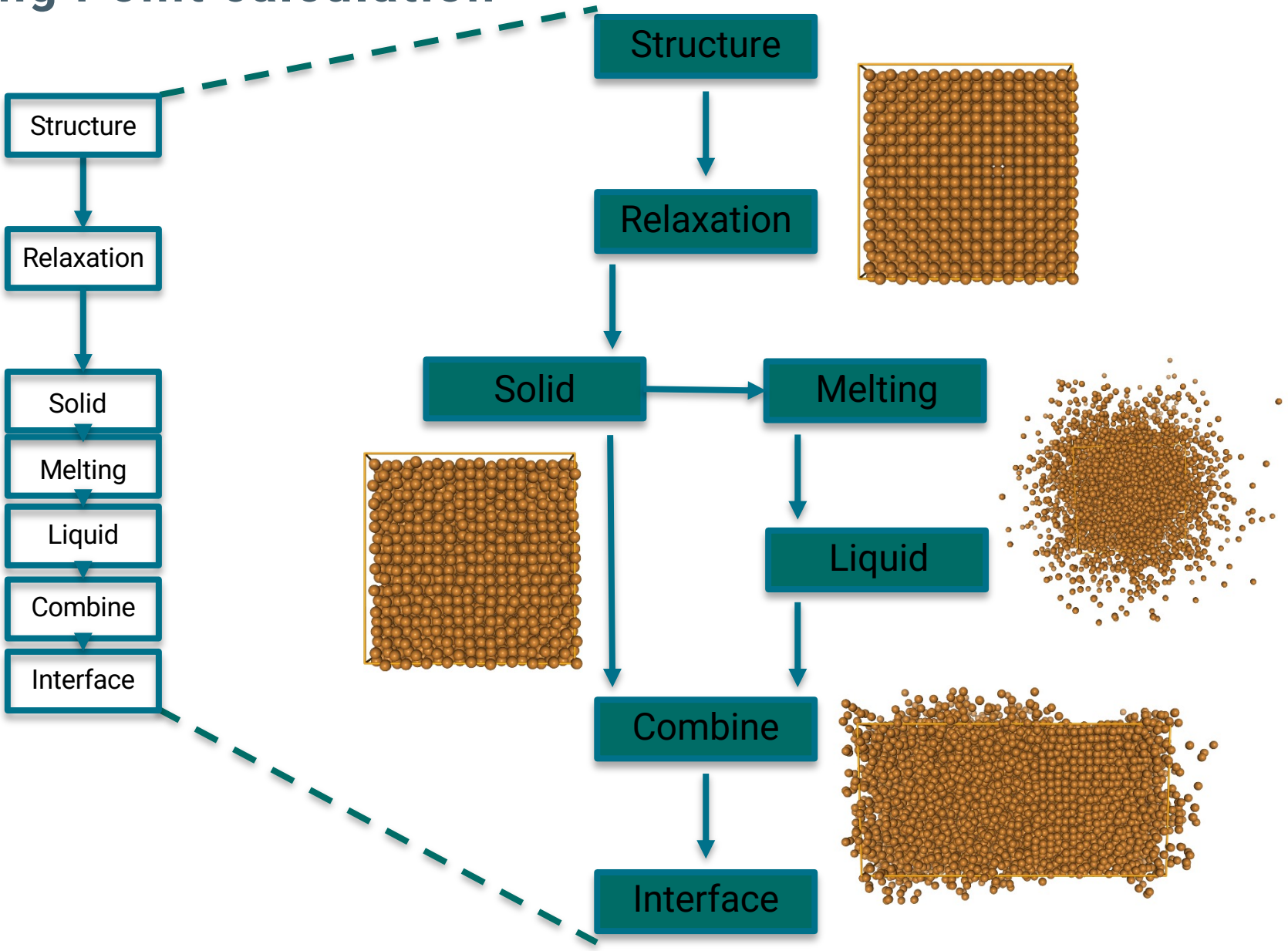
Melting Point calculation



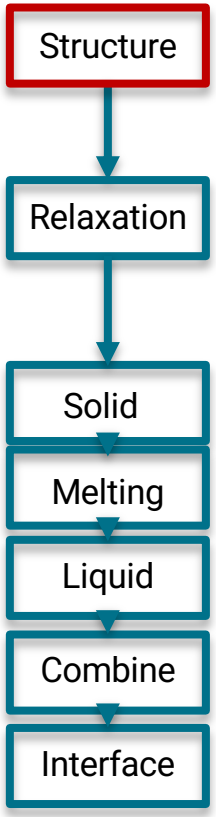
Melting Point calculation



Melting Point calculation



Melting Point calculation



Implement workflow for computing melting temperatures

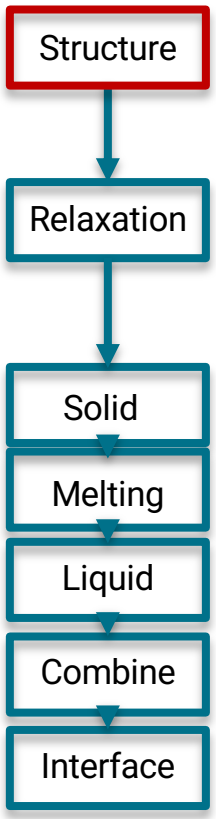
Import libraries

```
from pyiron import Project
import matplotlib.pylab as plt
```

Create project

```
pr = Project('demo')
```

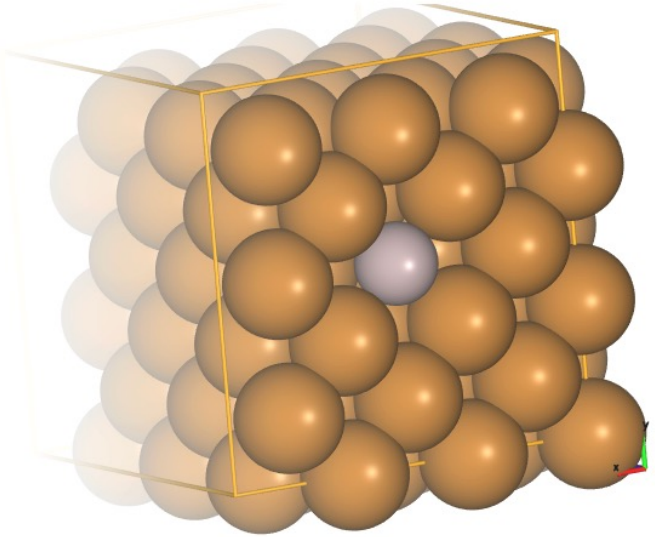
Melting Point calculation



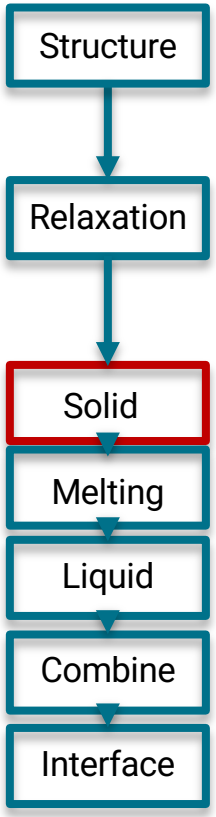
Create structure

fcc Al with vacancy

```
Cu = pr.create.structure.bulk('Cu', cubic=True).repeat(3)
Cu[19] = 'Al'
Cu.plot3d(particle_size=2)
```



Solid at finite temperature

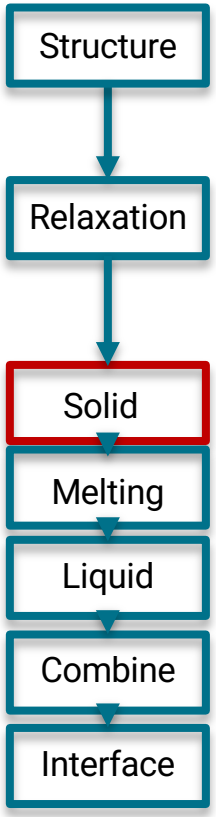


Perform MD simulations

```
job = pr.create.job.Lammps(job_name='MD_700K')
job.structure = Cu
job.list_potentials()[:10]
```

```
['1999--Liu-X-Y--Al-Cu--LAMMPS--ipr1',
 '2011--Apostol-F--Al-Cu--LAMMPS--ipr1',
 '2012--Jelinek-B--Al-Si-Mg-Cu-Fe--LAMMPS--ipr2',
 '2016--Zhou-X-W--Al-Cu--LAMMPS--ipr2',
 '2018--Zhou-X-W--Al-Cu-H--LAMMPS--ipr1',
 '2022--Mahata-A--Al-Cu--LAMMPS--ipr1',
 'EAM_Dynamo_CaiYe_1996_AlCu__MO_942551040047_005',
 'EAM_Dynamo_LiuLiuBorucki_1999_AlCu__MO_020851069572_000',
 'EMT_Asap_Standard_JacobsenStoltzeNorskov_1996_AlAgAuCuNiPdPt__MO_115316750986_001',
 'MEAM_LAMMPS_JelinekGrohHorstemeyer_2012_AlSiMgCuFe__MO_262519520678_000']
```

Solid at finite temperature



Perform MD simulations

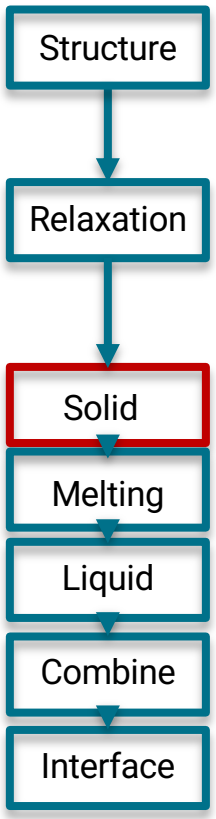
```
job = pr.create.job.Lammps(job_name='MD_700K')
job.structure = Cu
job.list_potentials()[:10]
```

```
['1999--Liu-X-Y--Al-Cu--LAMMPS--ipr1',
 '2011--Apostol-F--Al-Cu--LAMMPS--ipr1',
 '2012--Jelinek-B--Al-Si-Mg-Cu-Fe--LAMMPS--ipr2',
 '2016--Zhou-X-W--Al-Cu--LAMMPS--ipr2',
 '2018--Zhou-X-W--Al-Cu-H--LAMMPS--ipr1',
 '2022--Mahata-A--Al-Cu--LAMMPS--ipr1',
 'EAM_Dynamo_CaiYe_1996_AlCu__MO_942551040047_005',
 'EAM_Dynamo_LiuLiuBorucki_1999_AlCu__MO_020851069572_000',
 'EMT_Asap_Standard_JacobsenStoltzeNorskov_1996_AlAgAuCuNiPdPt__MO_115316750986_001',
 'MEAM_LAMMPS_JelinekGrohHorstemeyer_2012_AlSiMgCuFe__MO_262519520678_000']
```

```
job.potential = '2022--Mahata-A--Al-Cu--LAMMPS--ipr1'
job.calc_md(temperature=700, n_ionic_steps=10000)

job.run(run_again=True)
```

Solid at finite temperature



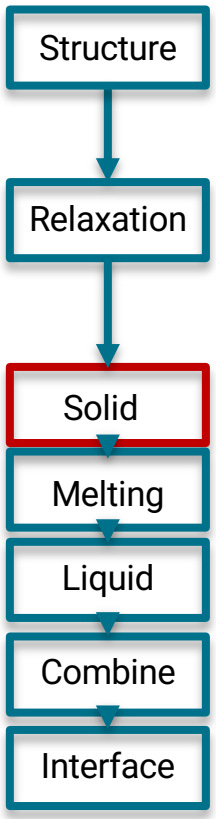
Direct access of all simulation data from database

Load job from database and analyse it

```
job_MD = pr.load('MD_700K')
job_MD.input.control
```

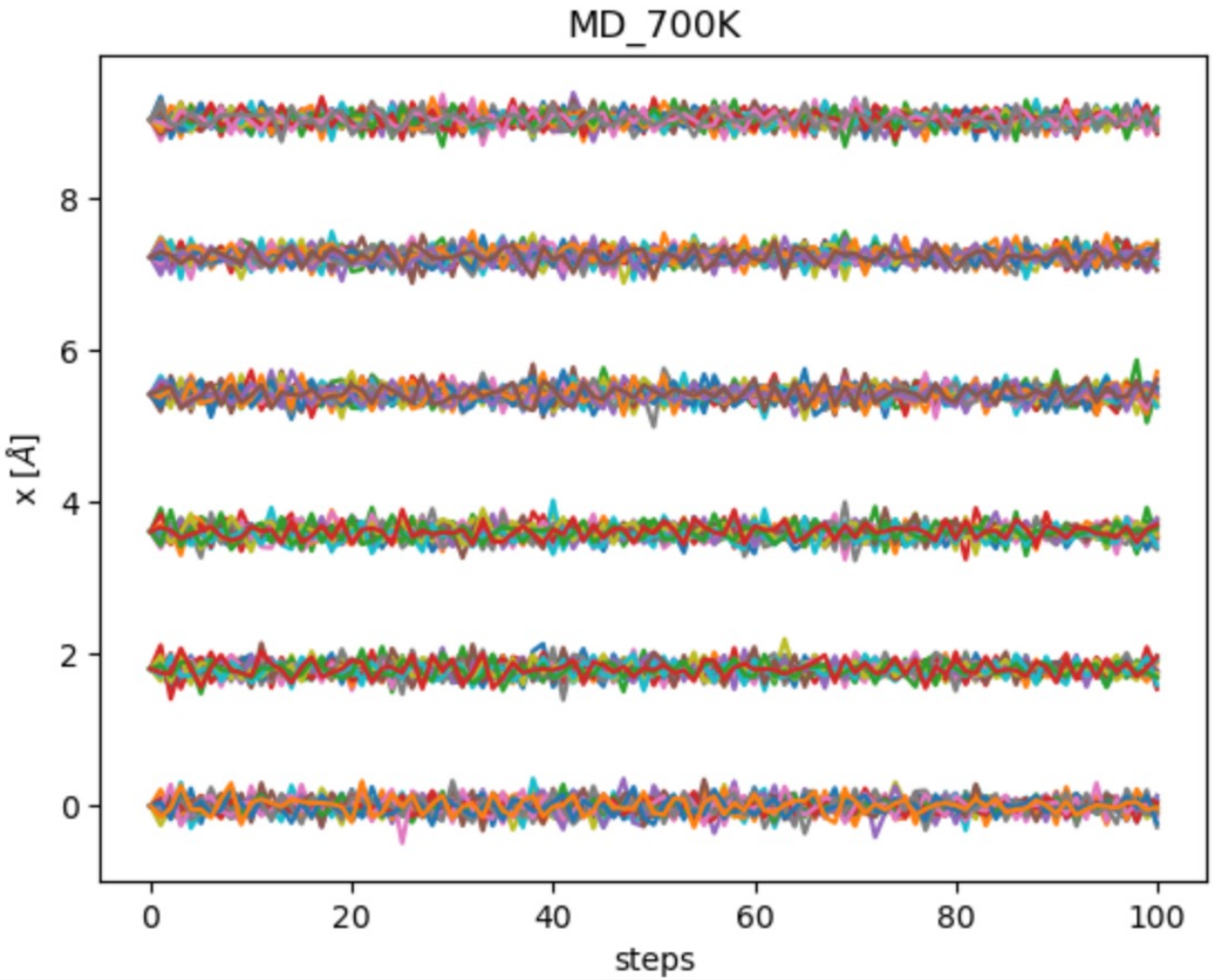
	Parameter	Value
0	units	metal
1	dimension	3
2	boundary	p p p
3	atom_style	atomic
4	read_data	structure.inp
5	include	potential.inp
6	fix___ensemble	all nvt temp 700.0 700.0 0.1
7	variable___dumptime	equal 100
8	variable___thermotime	equal 100
9	timestep	0.001
10	velocity	all create 1400.0 53471 dist gaussian
11	dump___1	all custom ${dumptime} dump.out id type xsu ysu zsu fx fy fz vx vy vz
12	dump_modify___1	sort id format line "%d %d %20.15g %20.15g %20.15g %20.15g %20.15g %20.15g %20.15g %20.15g %20.15g"
13	thermo_style	custom step temp pe etotal pxx pxy pxz pyy pyz pzz vol
14	thermo_modify	format float %20.15g

Solid at finite temperature

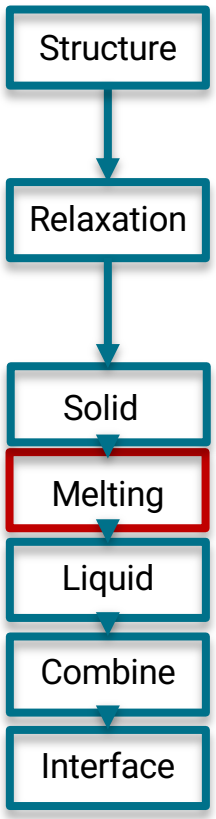


Direct access of all simulation data from database

```
plt.plot(job_MD.output.unwrapped_positions[:,:,0]);
plt.title(job_MD.job_name)
plt.xlabel('steps')
plt.ylabel('x [$\AA$]');
```



Solid at finite temperature



Jobs that are already in the database are automatically loaded rather than rerun

Run over series of temperatures

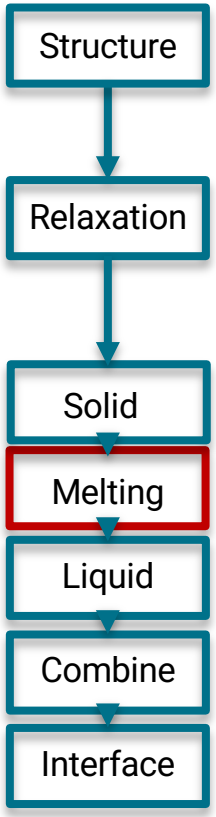
```
for T in np.arange(700, 1800, 100):
    job = pr.create.job.Lammps(job_name=f'MD_{T}K')
    job.structure = Cu

    job.potential = '2022--Mahata-A--Al-Cu--LAMMPS--ipr1'
    job.calc_md(temperature=T, n_ionic_steps=10000, pressure=0)

    job.run()
```

```
2023-04-30 19:28:51,123 - pyiron_log - WARNING - The job MD_700K is being loaded instead of running.
The job MD_800K was saved and received the ID: 282
The job MD_900K was saved and received the ID: 283
The job MD_1000K was saved and received the ID: 284
The job MD_1100K was saved and received the ID: 285
The job MD_1200K was saved and received the ID: 286
The job MD_1300K was saved and received the ID: 287
```

Solid at finite temperature



pyiron tables provide a powerful tool to map/reduce complex data into pandas Dataframes

Analyse data using pyiron tables

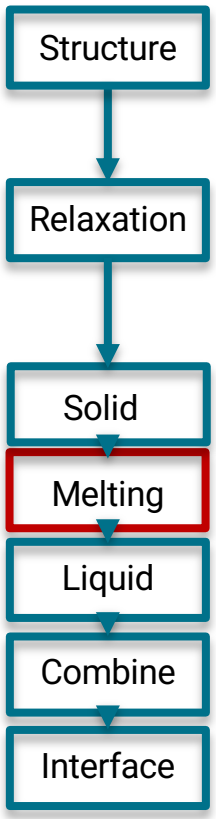
```
def get_temperature(job):
    return int(job.job_name.split('_')[1][:-1])

def get_displacement(job):
    pos = job['output/generic/unwrapped_positions']
    return np.mean(np.linalg.norm(pos[-1] - pos[0], axis=-1))
```

```
table = pr.create.table(delete_existing_job=True)
table.add['temperature'] = get_temperature
table.add['displacement'] = get_displacement
table.add.get_volume
table.add.get_energy_tot_per_atom
table.add.get_job_name
table.run()
```

```
The job table was saved and received the ID: 290
Loading and filtering jobs: 100% 13/13 [00:00<00:00, 1733.79it/s]
Processing jobs: 100% 12/12 [00:00<00:00, 261.72it/s]
```

Solid at finite temperature

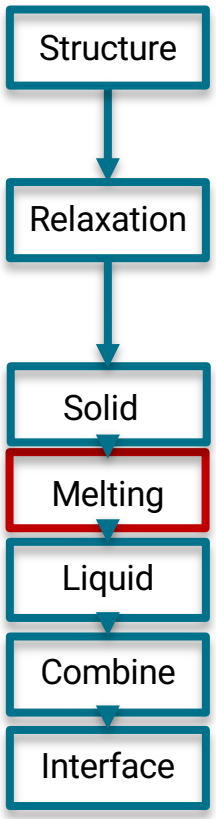


pyiron tables provide a powerful tool to map/reduce complex data into pandas Dataframes

```
df = table.get_dataframe().sort_values(by='temperature')
df
```

	job_id	job_name	energy_tot	volume	temperature	displacement
0	278	MD_700K	-3.353856	1270.238787	700	0.181369
4	282	MD_800K	-3.328802	1328.153374	800	0.247824
5	283	MD_900K	-3.256737	1352.336401	900	0.321491
6	284	MD_1000K	-3.286276	1334.194147	1000	0.264272
7	285	MD_1100K	-3.233808	1353.052931	1100	0.325095
8	286	MD_1200K	-3.241711	1346.373954	1200	0.310809
9	287	MD_1300K	-3.218381	1358.605471	1300	0.338850
1	279	MD_1400K	-3.203465	1374.474437	1400	0.362491
10	288	MD_1500K	-3.110352	1374.445628	1500	0.406013
3	281	MD_1600K	-3.075405	1382.937737	1600	0.646145

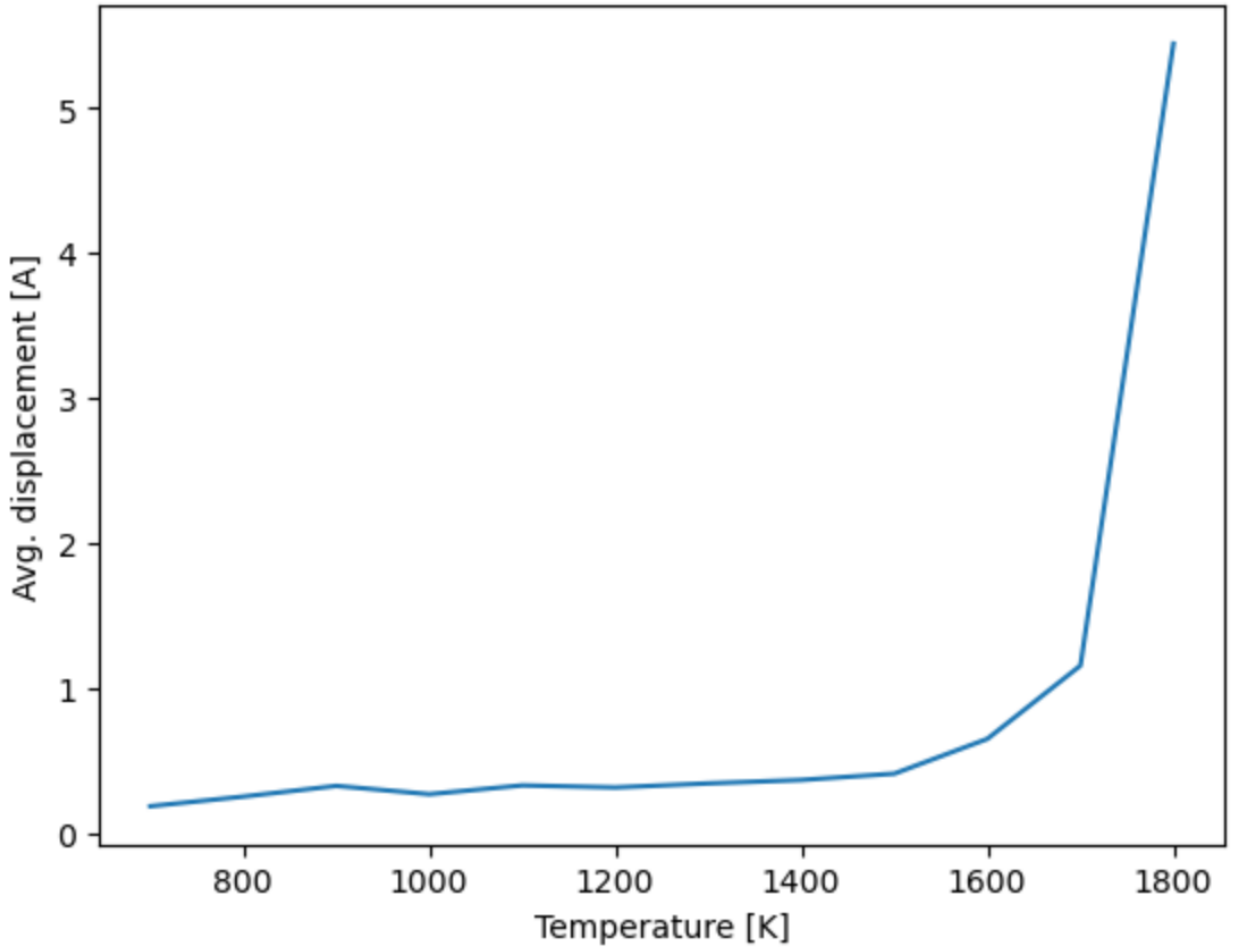
Solid at finite temperature



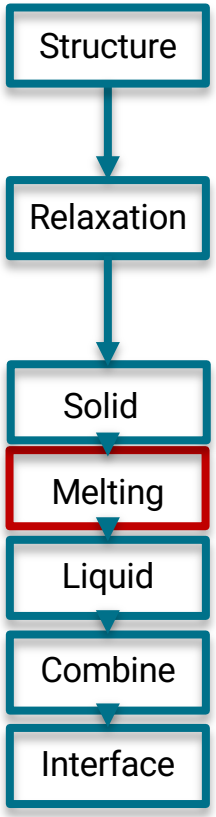
Even large datasets such as trajectories are accessible via hierarchical database structure (sql + hdf5)

Analyse average displacement

```
plt.plot(df.temperature, df.displacement)
plt.xlabel('Temperature [K]')
plt.ylabel('Avg. displacement [A]');
```

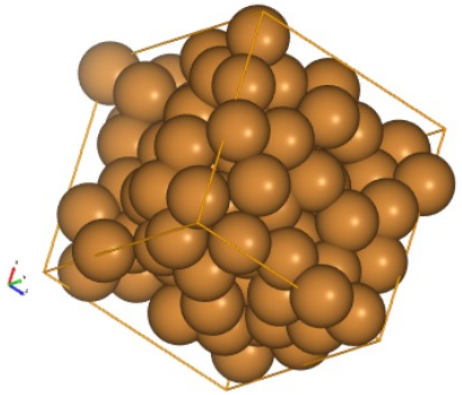


Solid at finite temperature

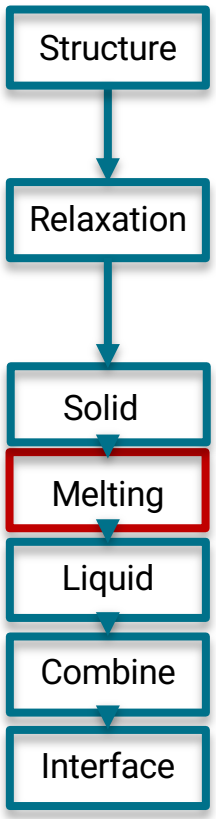


Analyse structure

```
job_MD.get_structure(-1).plot3d(particle_size=2)
```



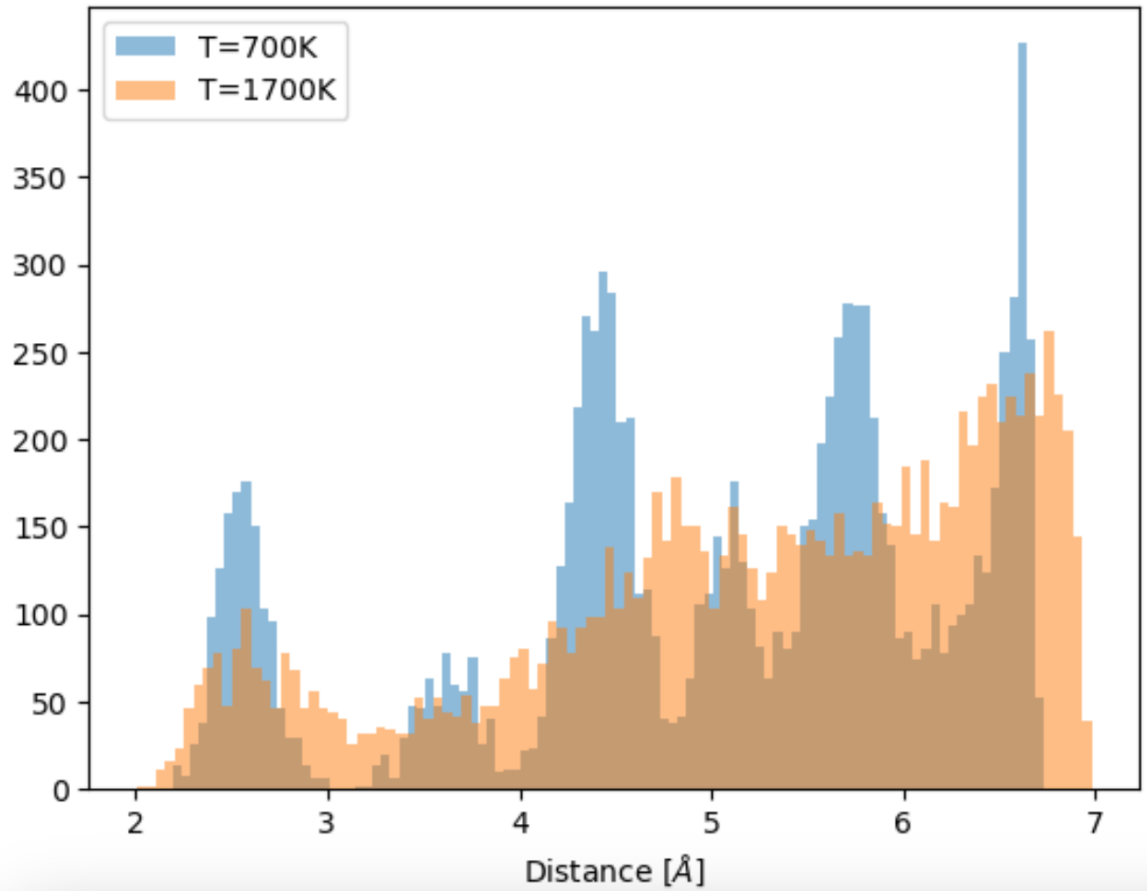
Solid at finite temperature



Full integration into numpy, matplotlib etc.

Pair-correlation function ¶

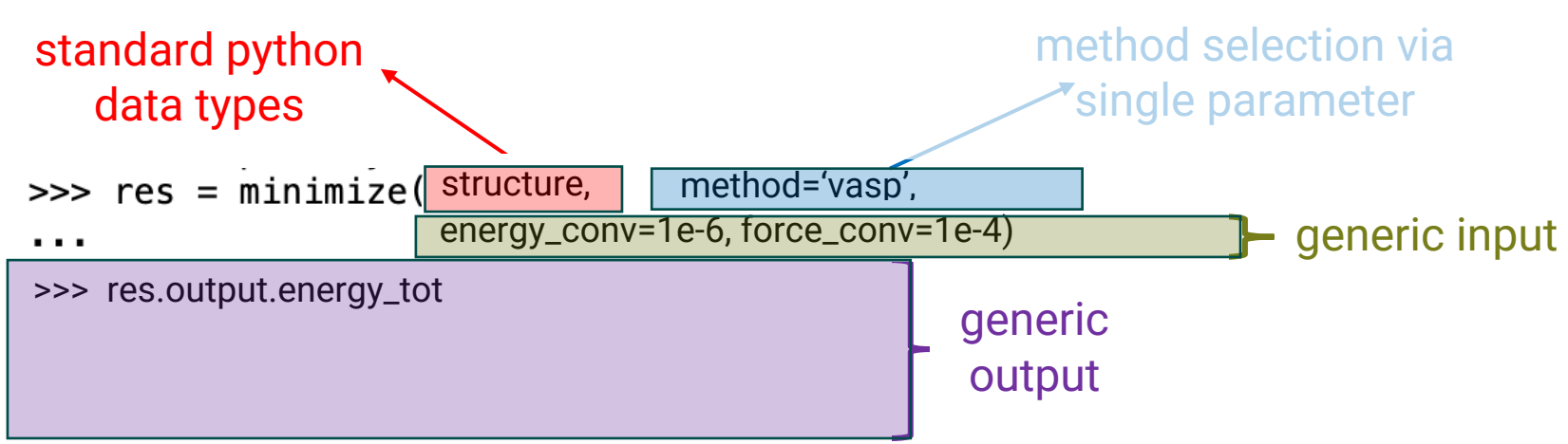
```
for T in [700, 1700]:
    job_MD = pr.load(f'MD_{T}K')
    nbrs = job_MD.get_structure(-1).get_neighbors(num_neighbors=100)
    plt.hist(nbrs.distances.flatten(), bins=100, alpha=0.5, label=f'T={T}K')
    plt.xlabel('Distance [$\AA$]')
    plt.legend();
```



Translation to materials science simulations

Features beyond NumPy/SciPy

- Automatic upload of input + selected output to database/storage
  - Serialization
- Upscaling to HPC (exascale)
  - Job managment
- Full integration into Jupyter
  - e.g. visualization of complex data structures such as atomic trajectories

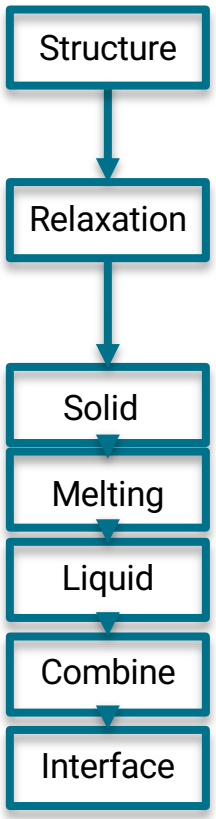


CONDA-FORGE

pyiron

Pyiron provides all these features!

Upscale to HPC




```
job.potential = 'CuAl_lammps_eam'
job.calc_md(temperature=600, pressure=0, n_ionic_steps=1000)
job.server.queue_view
# job.run()
```

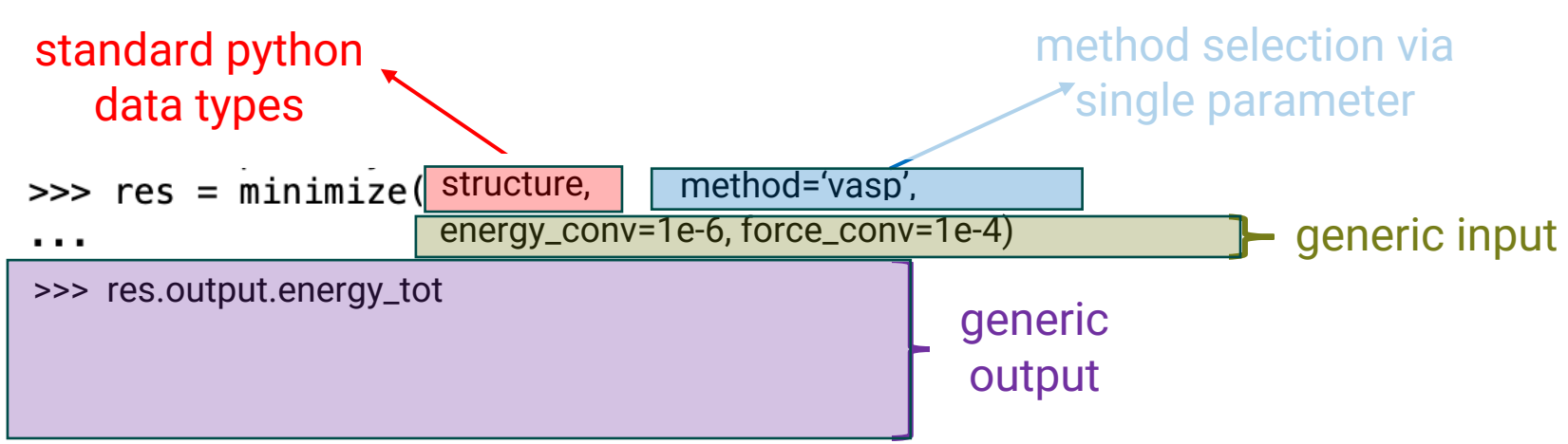
	maximum cores	minimum cores	run time limit
impi_hy*	1280	40	259200
impi_hydra	4240	20	259200
impi_hydra_cmfe.*	1280	40	259200
impi_hydra_small	40	1	604800

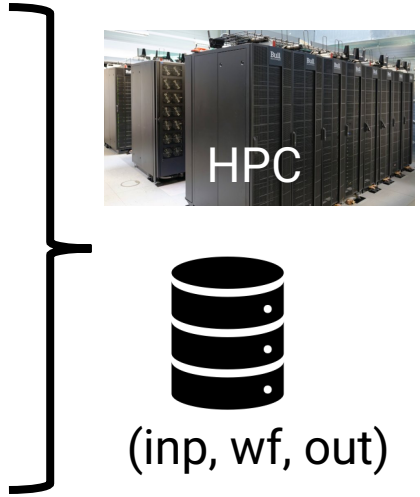
```
job.potential = 'CuAl_lammps_eam'
job.calc_md(temperature=600, pressure=0, n_ionic_steps=1000000)
job.server.core = 40
job.server.queue = 'impi_hydra_small'
job.run()
```

Translation to materials science simulations

Features beyond NumPy/SciPy

- Automatic upload of input + selected output to database/storage
  - Serialization
- Upscaling to HPC (exascale)
  - Job managment
- Full integration into Jupyter
  - e.g. visualization of complex data structures such as atomic trajectories



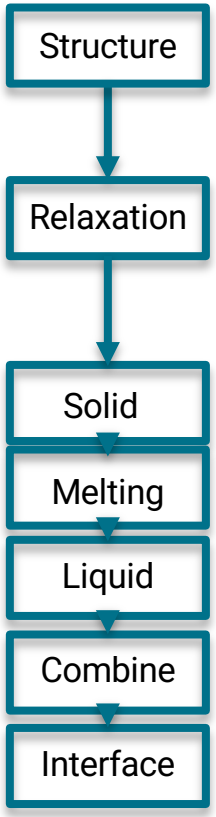


CONDA-FORGE

pyiron

Pyiron provides all these features!

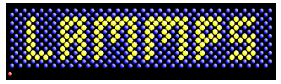
Upscale to HPC



Run over several atomistic simulations

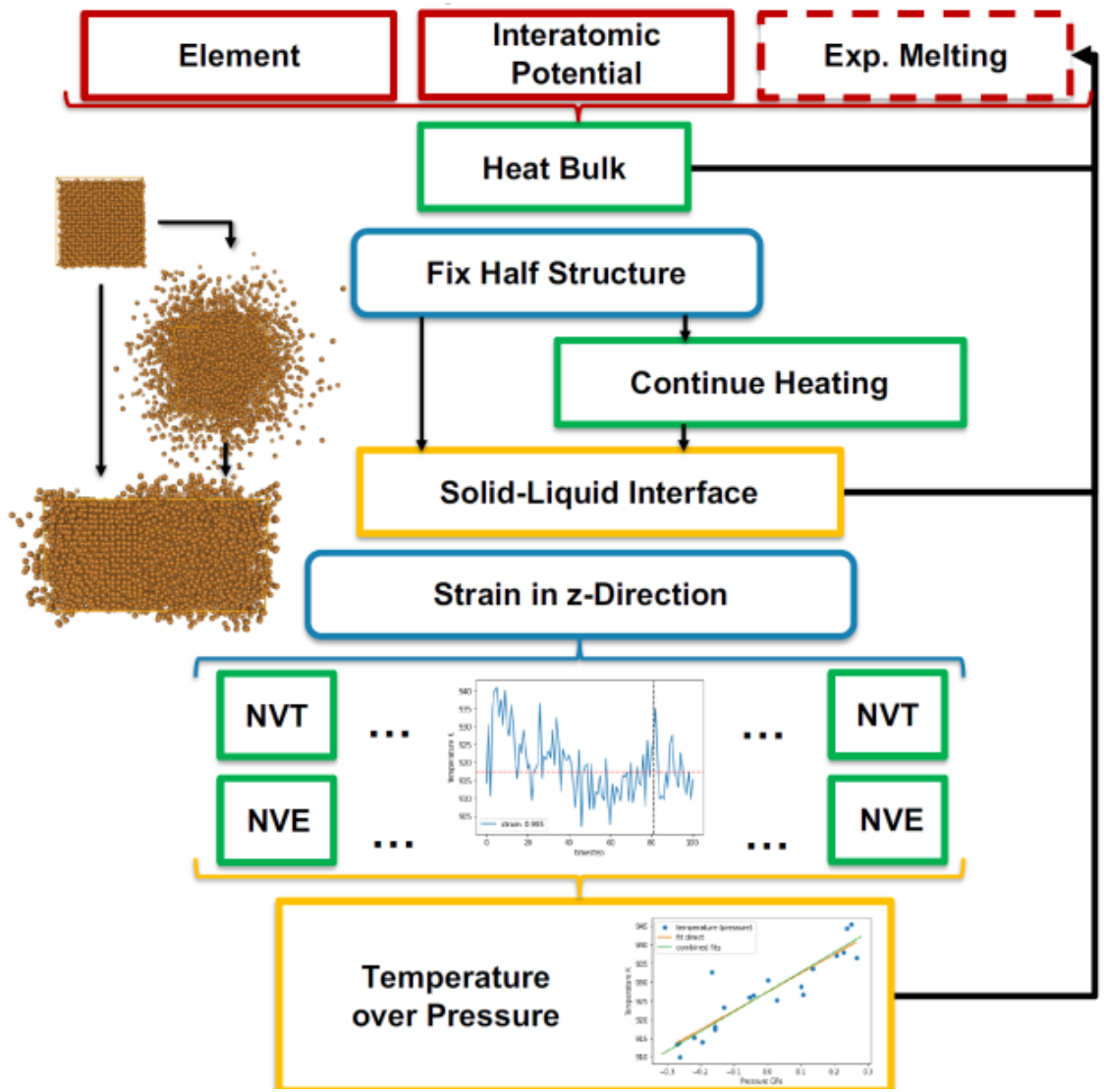
```
pr = Project('benchmark')
Al = pr.create.structure.bulk('Al', cubic=True).repeat(3)
for job_type in ['Lammps', 'Vasp', 'Sphinx']:
    job = pr.create_job(job_type=job_type, job_name=job_type)
    job.structure = Al
    job.run()

    Al = job.get_structure(-1)  # get last structure
```

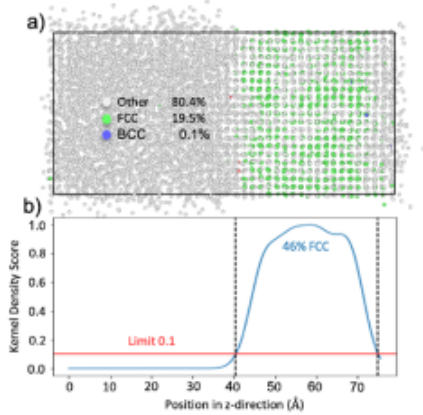


Fully interoperable via generic input/output

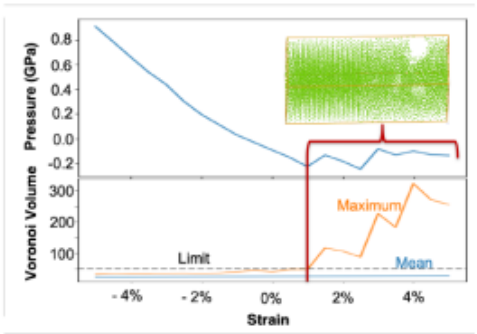
Fully autonomous algorithm to determine melting point



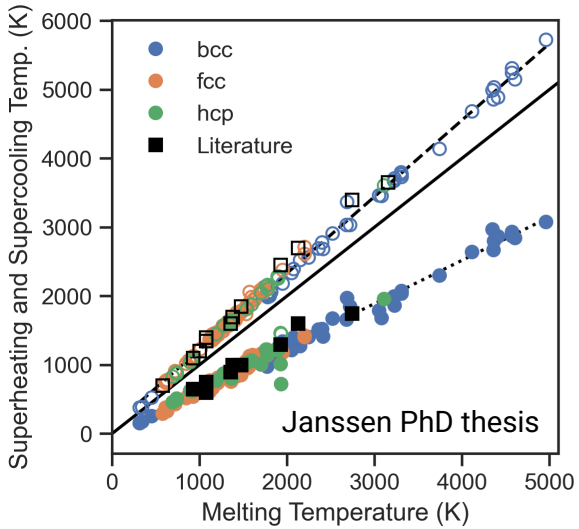
Solid Liquid Detector



Void Detector



Chemical trends & descriptors



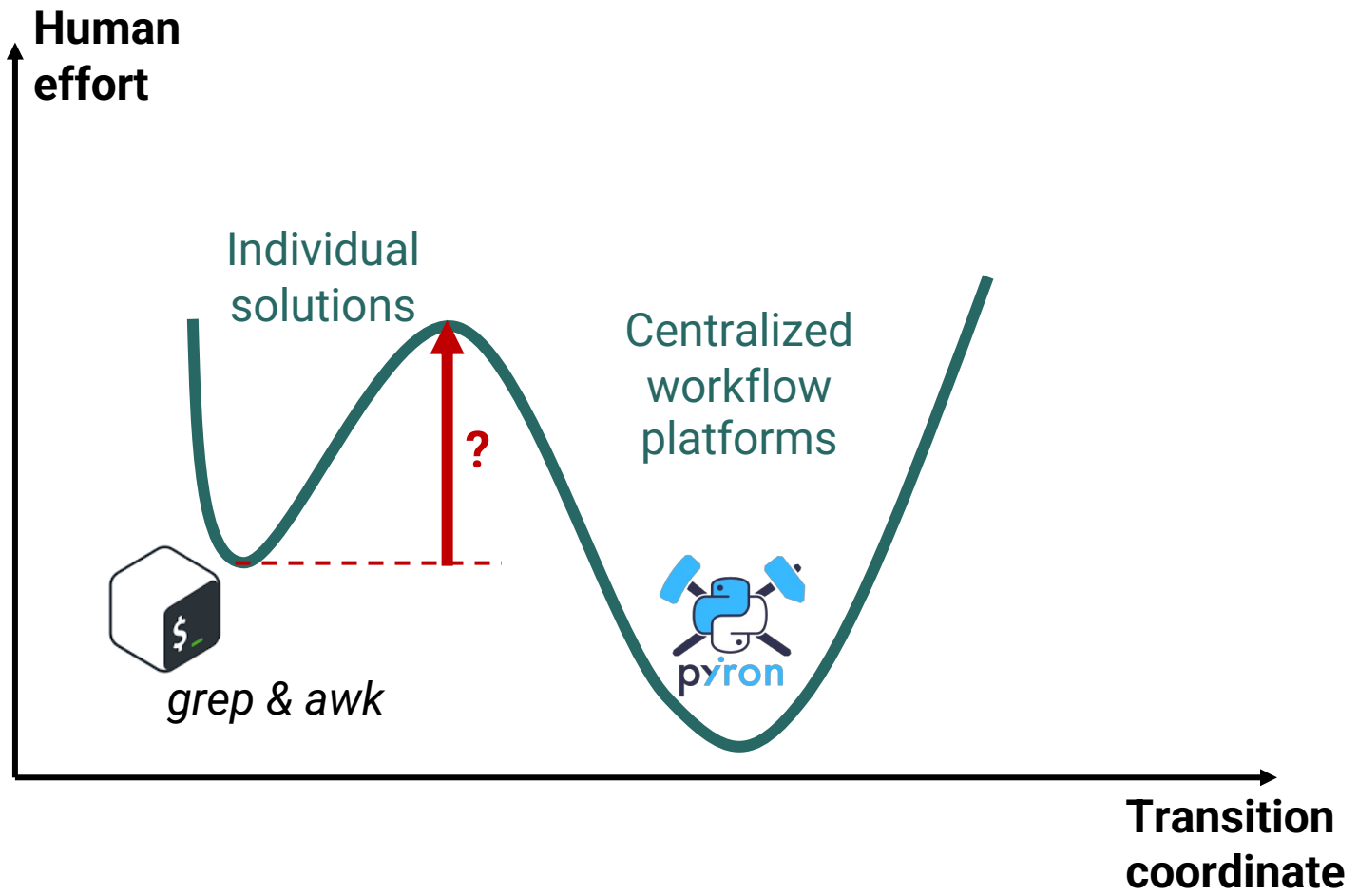
- Overheating/undercooling vs T_m
- > 300 emp. potentials

Zhu, Janssen, Ishibashi, Kormann, Grabowski, JN, Comp. Mat. Sci. 187, 12 (2021)

Sociological challenges

How to successfully drive transfer?

- user-friendliness
- bottom-up development
- developers must be active users
- internal & external workshops
- hackathons
- documentation & examples
- ‘killer’ applications
- scientific breakthroughs



pyiron as platform for internal & external collaborations

Hands-on interactive workshops (examples)

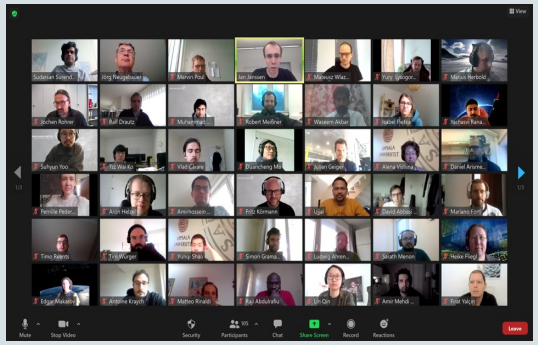
From Atomistics to Phase Diagrams

Constructing ML Potentials

- **Talk by R. Drautz**

ADIS 2021

> 200 participants



Hackathons & Journal Club

- Weekly 2-3 hour events
- 5-10 people
- Interactive prototyping & development of new concepts and tools

Interactive seminars

Teaching platform (example)

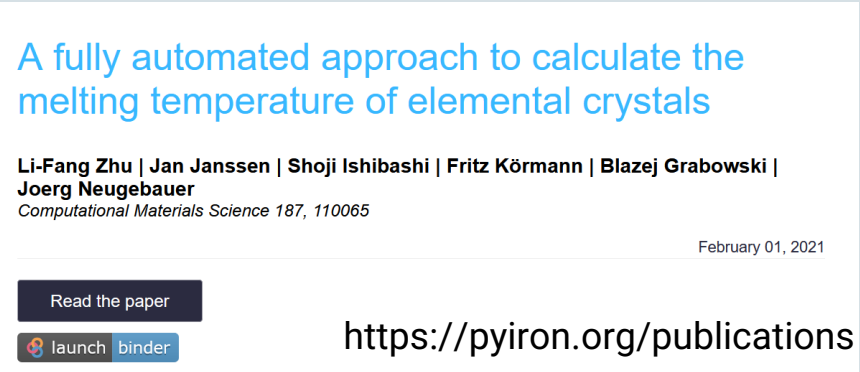
Lecture: Modelling and Engineering of Nanoscale Materials (Univ. Gent)

Quote:

"pyiron changed our exercise lessons drastically, by allowing more time to tackle advanced and complex problems instead of dealing with code technicalities."
– Sander Borgmans (Tutor)

Publications

Read papers and run published notebooks using pyiron and mybinder



https://pyiron.org/publications

Nucleus for large-scale networks: MaterialDigital

MATERIALDIGITAL

Initiative to digitize materials: particular strategic importance for Germany as a business location

The project goals: What an industrially relevant material data space must fulfill

Sovereignty: Security, ownership and access rights to the data, taking into account the interests of the data creators, are at the forefront of all activities.

Reproducibility: The standardized description of data generation. This must be transparent, traceable and repeatable.

Accessibility: The uniformly and comprehensively described data must be retrievable in order to avoid redundancies in research.

Adaptability: Flexible data logic that is always aligned with new research and development findings.

Curation: Clearly defined specifications for the data enable its quality assurance.

Fraunhofer IWM · KIT Karlsruher Institut für Technologie · IWT Leibniz-Institut für Werkstofforientierte Technologien · MAX-PLANCK-INSTITUT FÜR EISENFORSCHUNG GmbH · BAM Bundesanstalt für Materialforschung und -prüfung · Fraunhofer AISEC · FIZ Karlsruhe Leibniz Institute for Information Infrastructure

www.materialdigital.de

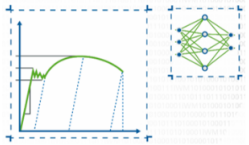
Nucleus for large-scale networks: MaterialDigital

Initiative to digitize materials: particular strategic importance for Germany as a business location

The vision: A decentralized data room as a unity of data and data processing

Generation of material data: When data is generated, it is stored according to unified schemas.

Decentralized storage: The large amounts of data remain at the place of their creation. No data provider cedes its control.



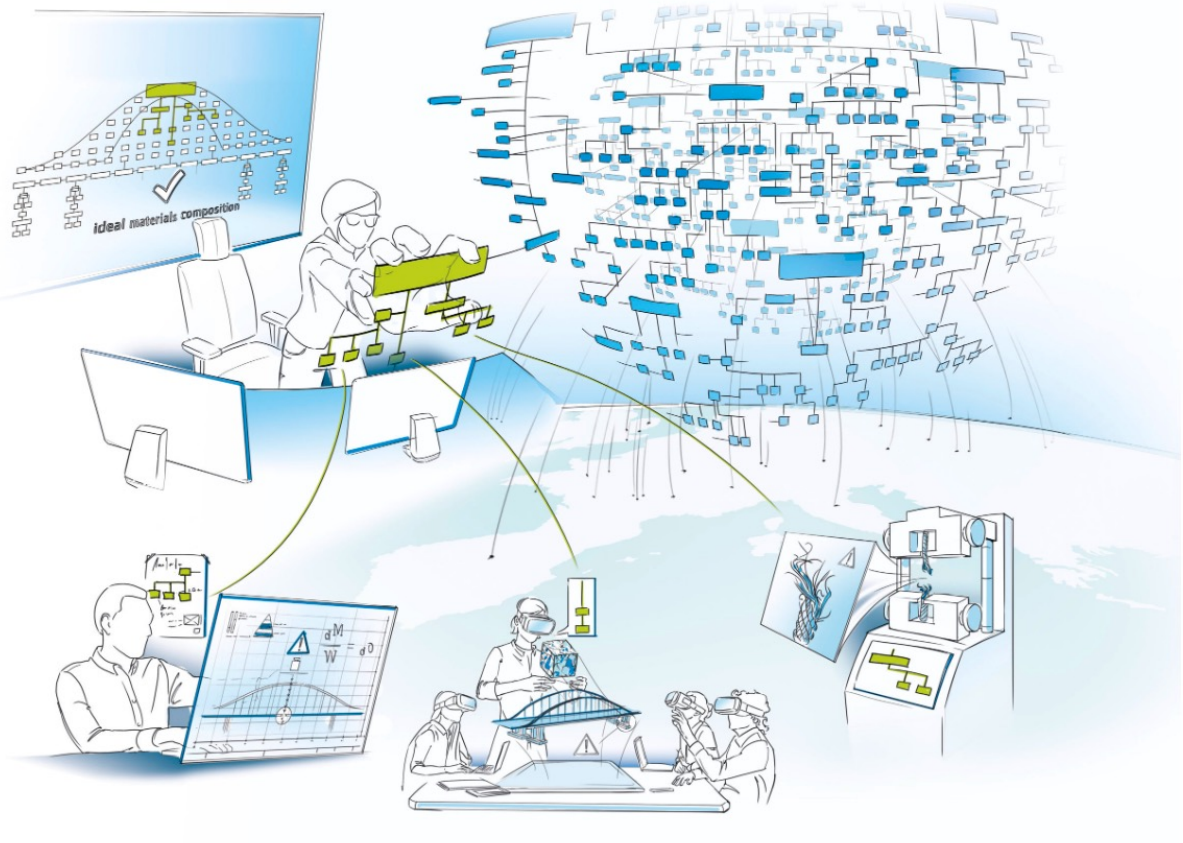
Integrated analysis: Linked software environments allow standardized data access and processing.

Remote access: The network architecture allows external partners to authorize access to the local environment.

Automated workflows

pyiron

Nucleus for large-scale networks: NFDI-MatWerk



NFDI-MatWerk

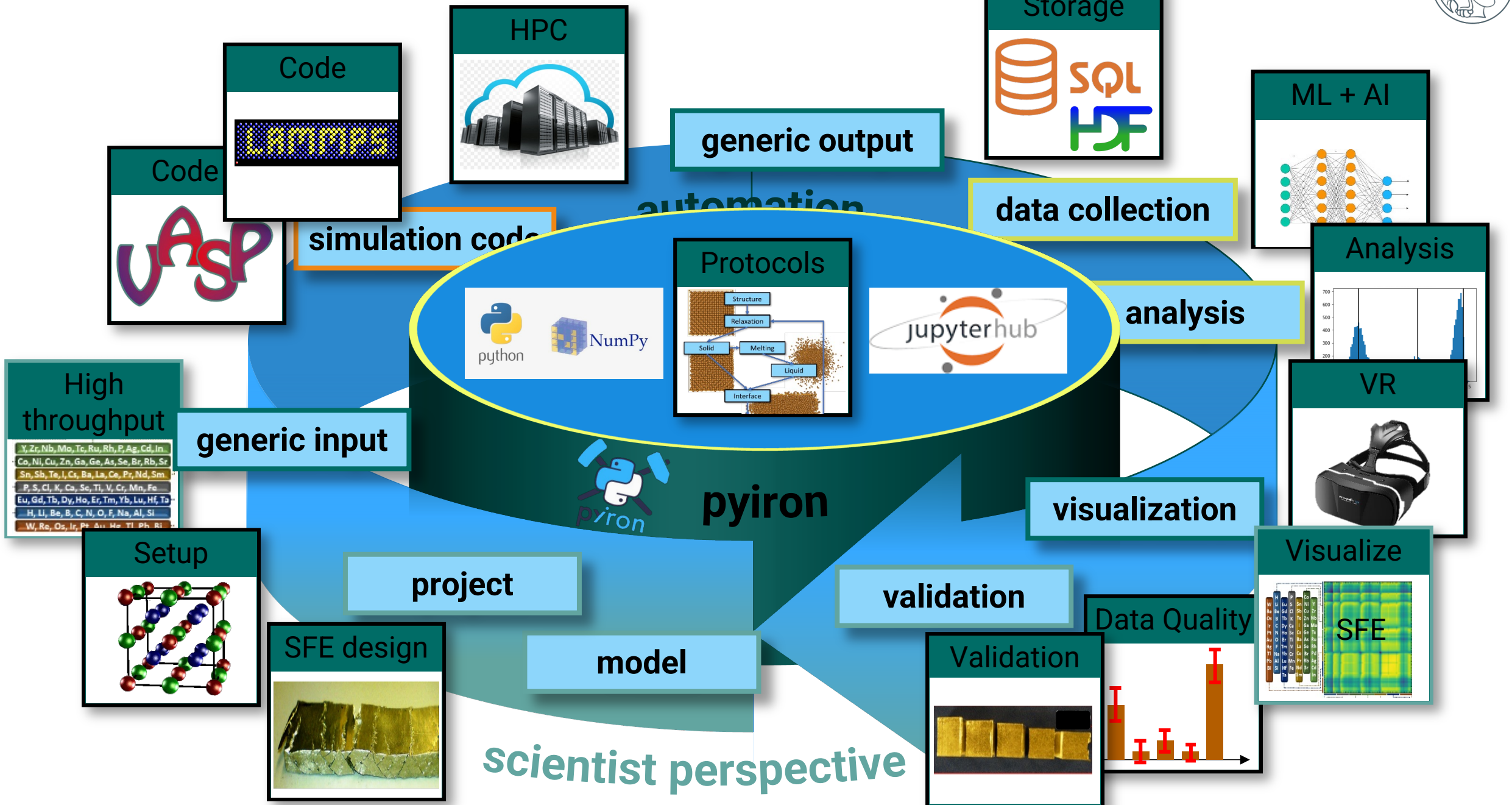
On the route towards a *National Research Data Infrastructure* for Materials Science & Engineering.

Workflows as central part in materials science & engineering!

pyiron

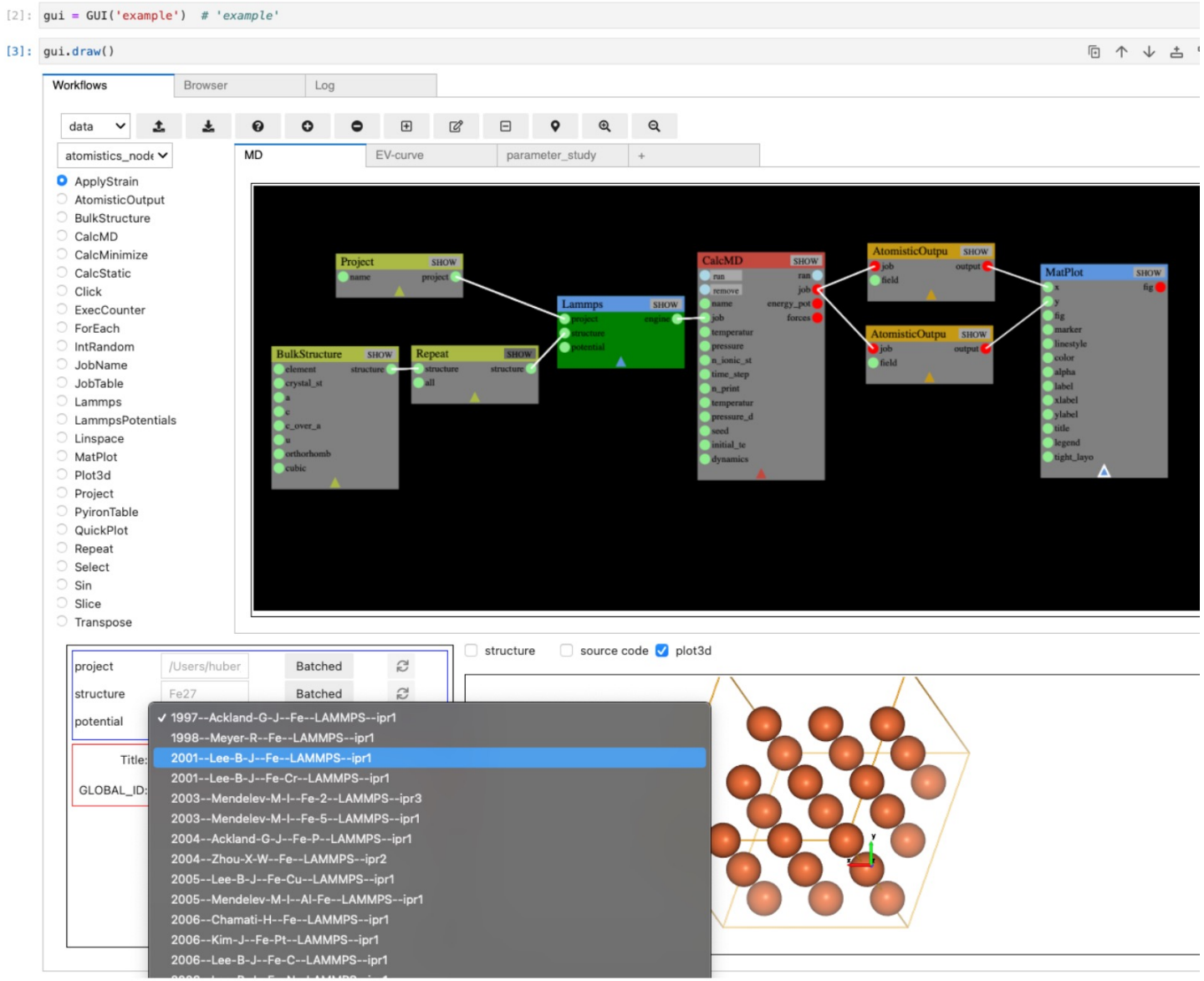
www.nfdi-matwerk.de

Automating simulation life cycle by pyiron



Next steps

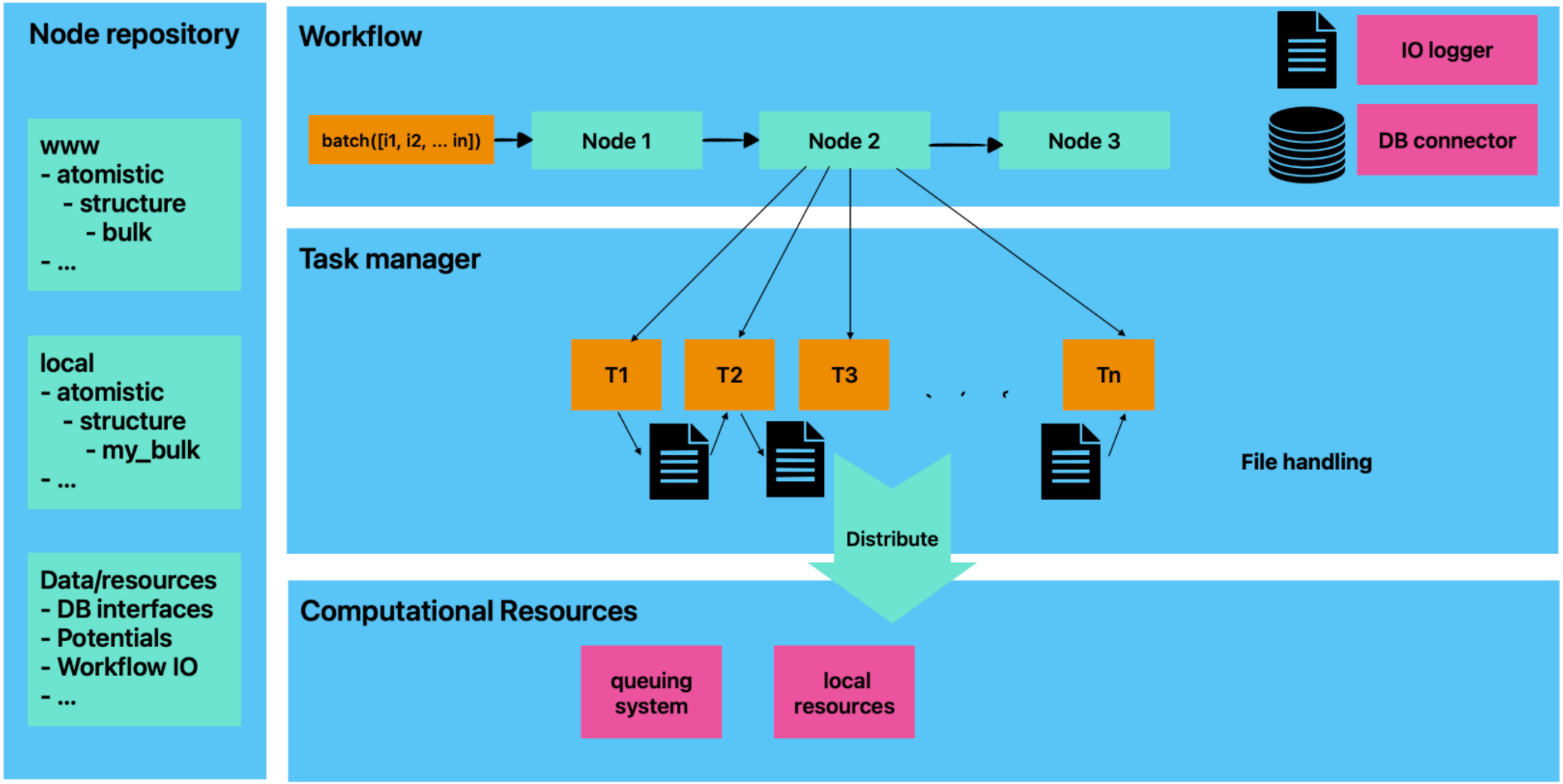
IronFlow - Jupyter-based visual scripting gui for running pyiron workflow graphs



- Note: under active development
  - not yet production ready
- Features
  - Ontologic typing
  - Fully integrated in Jupyter
  - Batch jobs (e.g.for ML)
  - New nodes can be easily written and added
- Workflow programming and application for everybody

https://github.com/pyiron/ironflow

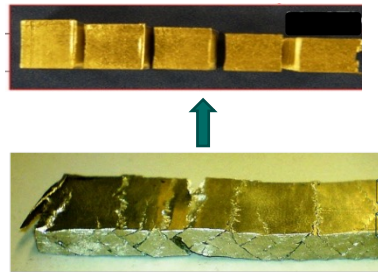
Node-based pyiron architecture



https://github.com/pyiron/ironflow/issues/175

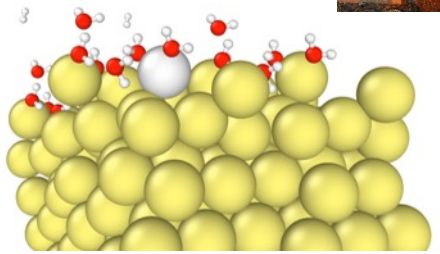
Application: Materials design

Alloy design



Sandlobes et al. Sci. Reports 2017
Rao et al., Science 2022

Corrosion

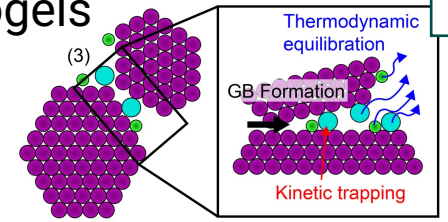


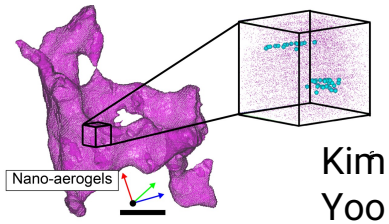
Surendralal et al. PRL 2018 & 2021
Yoo et al. PRL 2018

Explore configuration space by automated workflows

pyiron

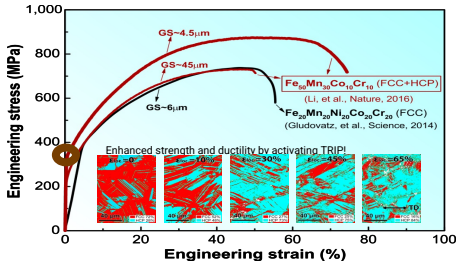
Nano-aerogels





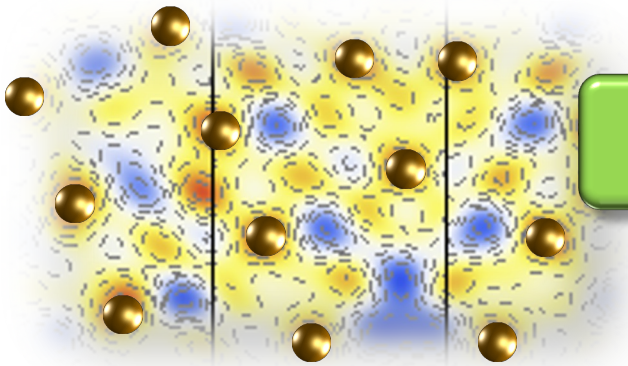
Kim et al., JACS 2022
Yoo et al. NJP 2022

Magnetic materials (HEA, …)



Rao et al. Adv. Funct. Mater. 2021
Ikeda et al. NPJ Comp. Mat 2021

Example: Paramagnetic description of defects



AimsGB
(GB structure)

SQS generator
(Disorder)

Chemical positions

SxExtOpt
(Freysoldt scheme)

pyiron

Averaged forces

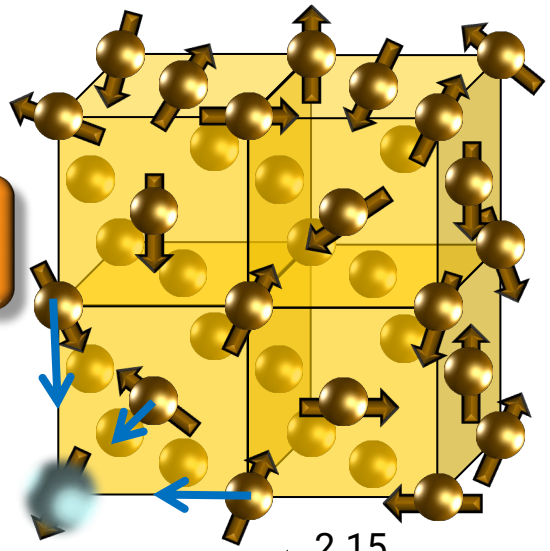
Atomic forces

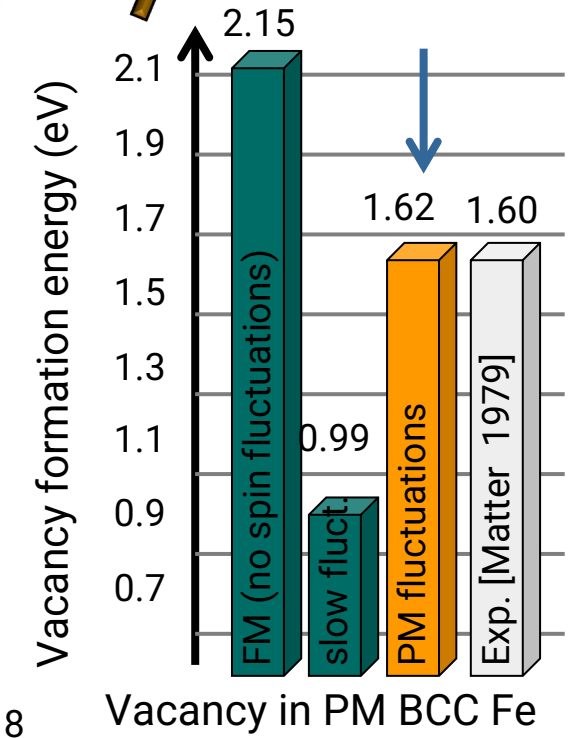
Atomic forces

Magnetic configuration

Magnetic configuration

DFT: SPHInX
(spin constraints)





I. Stockem et al. PRL 2018
O. Hegde et al., PRB 2020

Example: Alloy Discovery

METALLURGY

Rao *et al.*, *Science* **378**, 78–85 (2022)

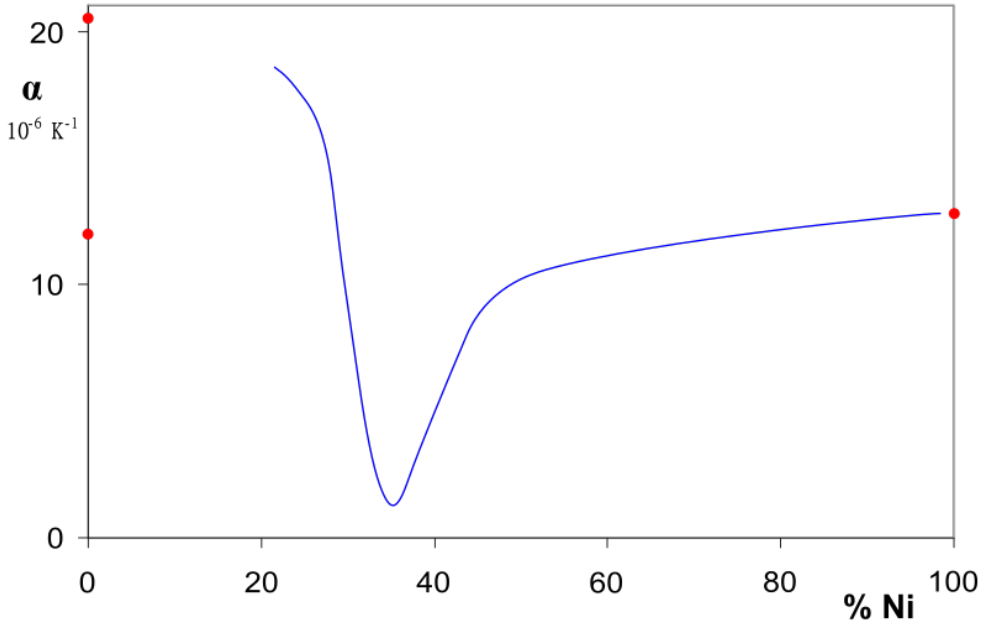
Machine learning–enabled high-entropy alloy discovery

Ziyuan Rao[1], Po-Yen Tung[1,2], Ruiwen Xie[3], Ye Wei[1]*, Hongbin Zhang[3], Alberto Ferrari[4], T.P.C. Klaver[4], Fritz Körmann[1,4], Prithiv Thoudden Sukumar[1], Alisson Kwiatkowski da Silva[1], Yao Chen[1,5], Zhiming Li[1,6], Dirk Ponge[1], Jörg Neugebauer[1], Oliver Gutfleisch[1,3], Stefan Bauer[7], Dierk Raabe[1]*

Aim:

Identify from millions of possible compositionally complex alloy compositions those with the lowest/vanishing thermal expansion coefficient (Invar alloys)

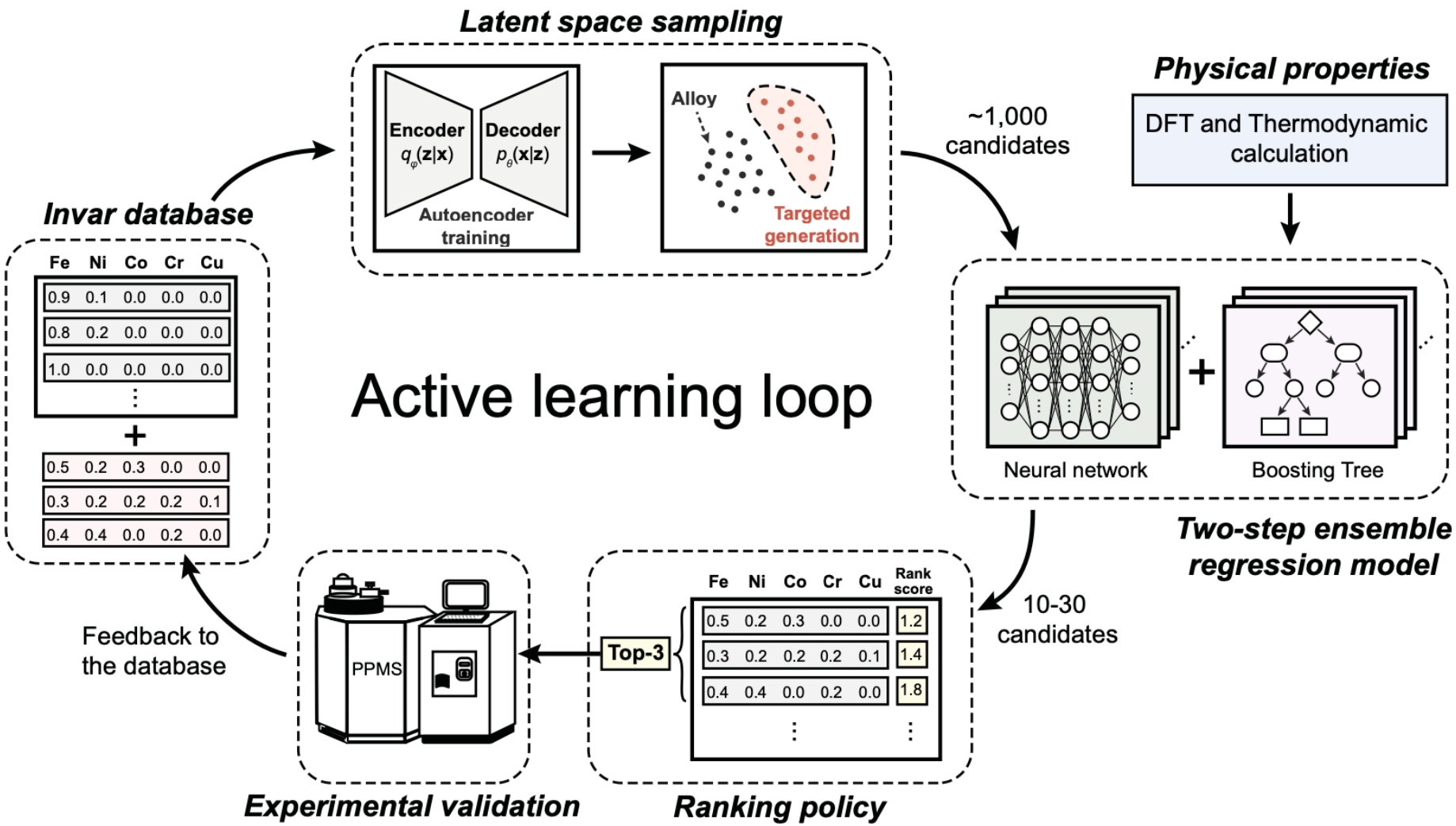
Original Invar Alloy (FeNi)



Challenges:

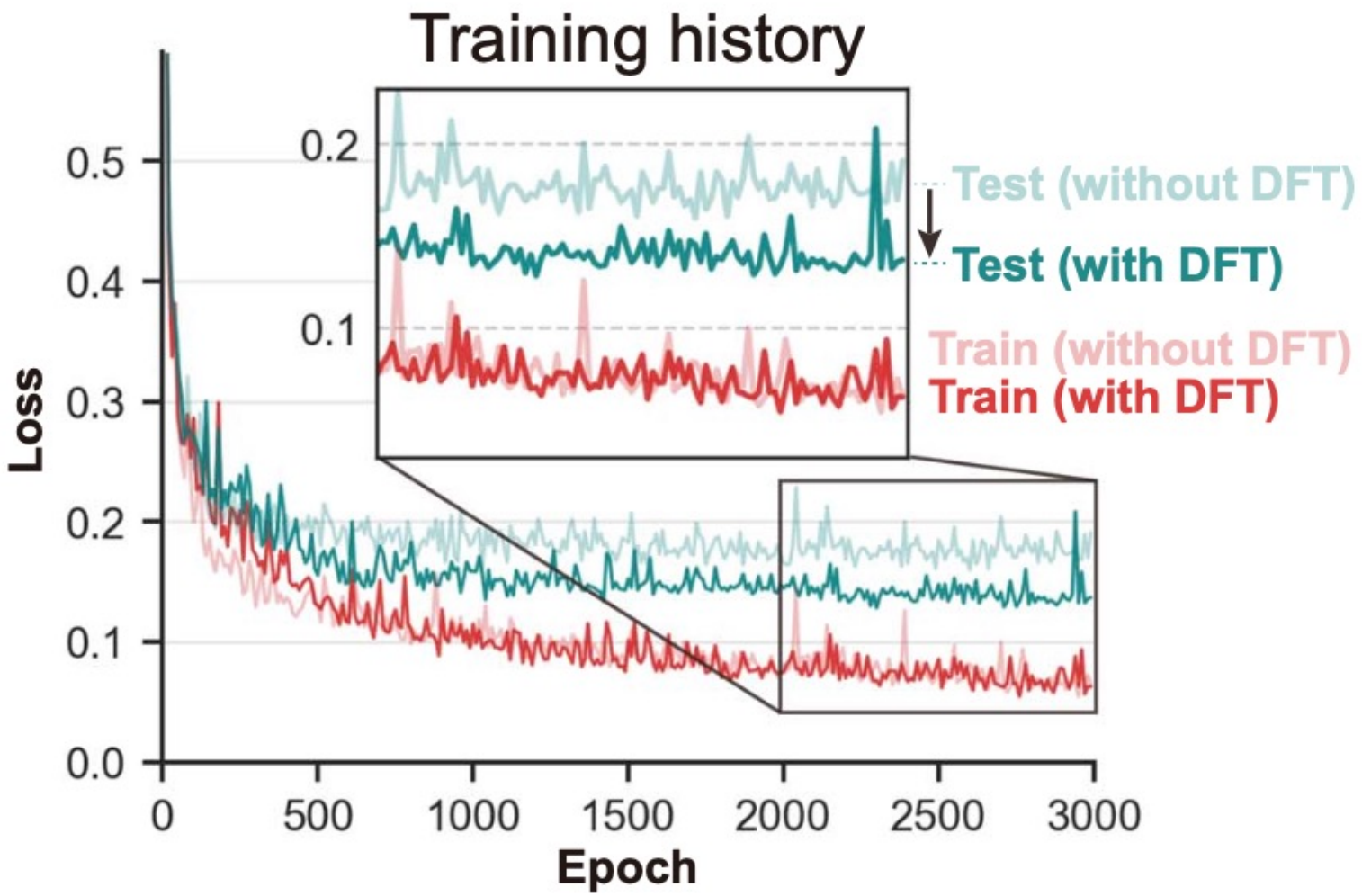
- Sharp local minima
- Interplay between magnetism & phonons
- > 5 elements

ML Architecture



Rao et al., Science 378, 78–85 (2022)

Importance of Physics-Informed Descriptors

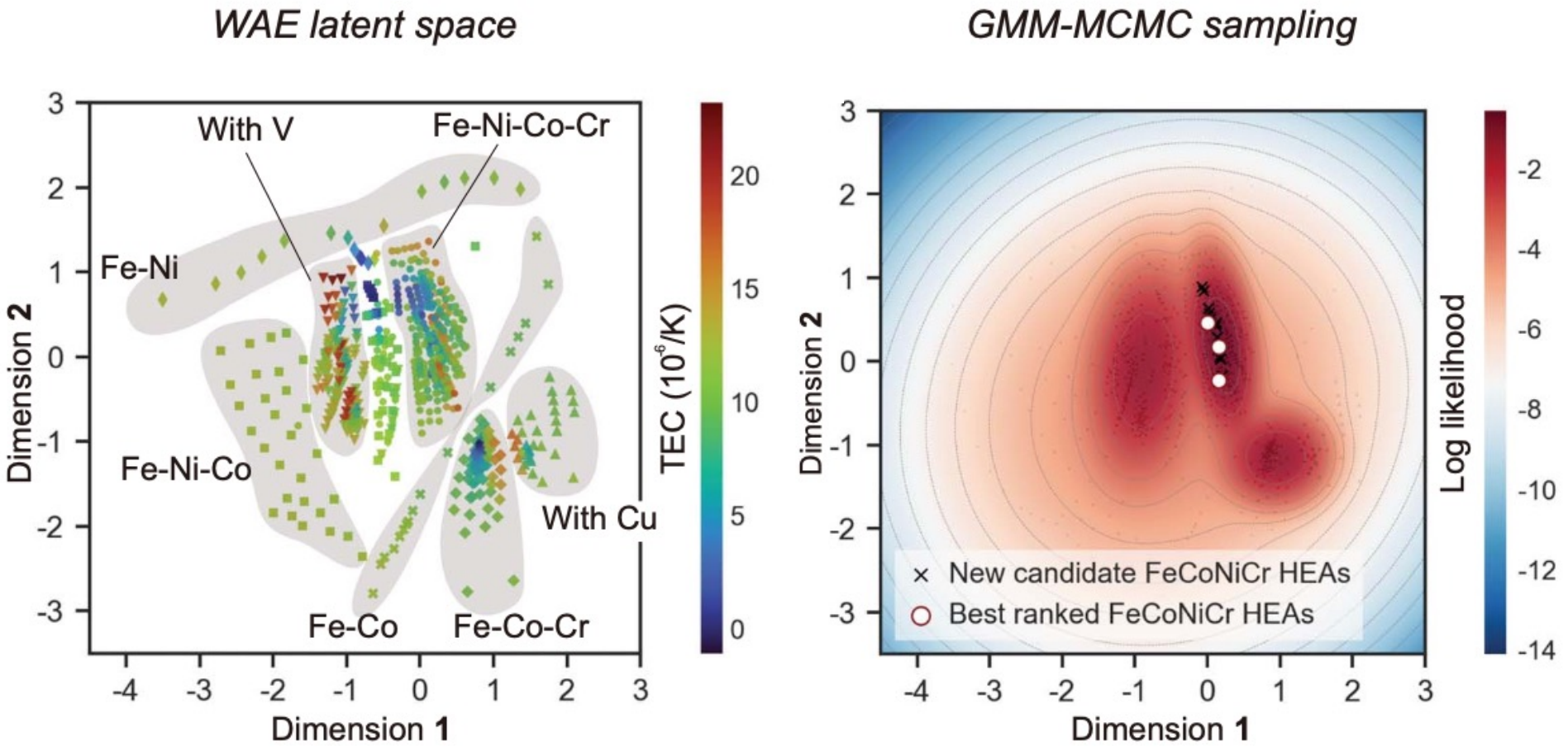


Physics descriptors:
ω_s - magnetostriction
T_c - Curie temperature):

Inclusion of DFT computed descriptors significantly improves ML model!

Rao et al., Science 378, 78–85 (2022)

Dimensionality Reduction



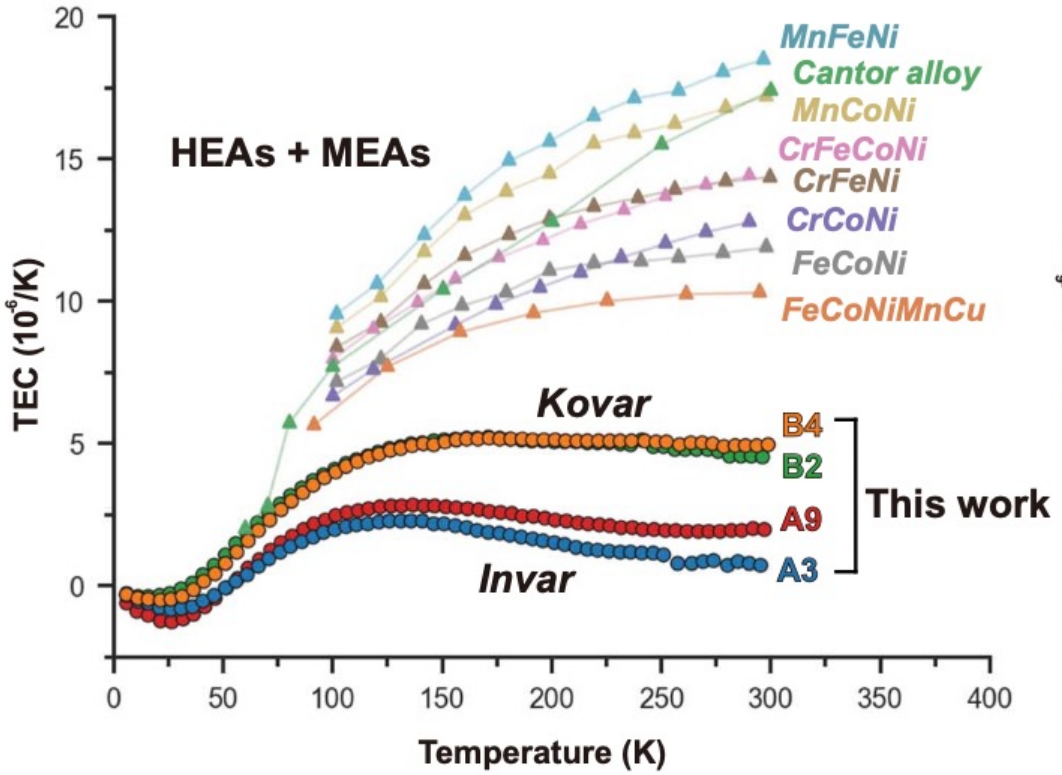
Islands in latent space reflect compositional differences and provide treasure map for HEA-based Invar alloys!

GMM - Gaussian mixture model
MCMC - Markov chain Monte Carlo sampling
WAE - Wasserstein autoencoder architecture

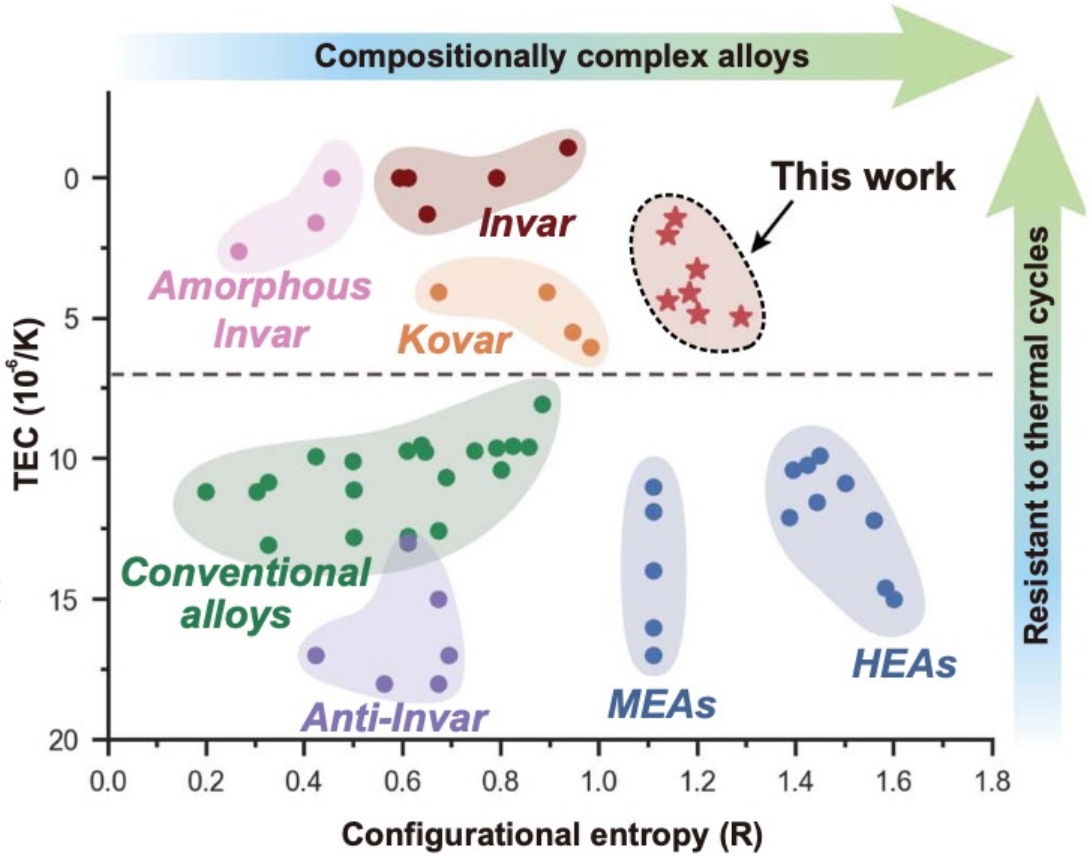
Rao et al., Science 378, 78–85 (2022)

Discovery of Invar Alloys

Experiment



Comparison



Design of high-entropy Invar alloys using very sparse experimental data by employing DFT descriptors based on physical models!

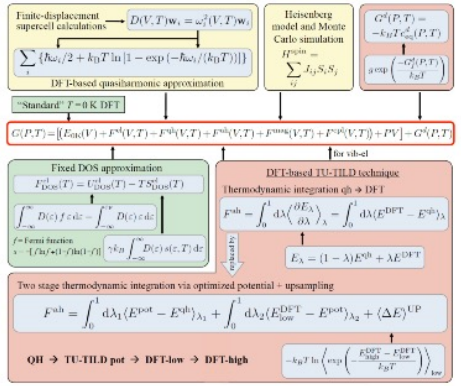
Rao et al., Science 378, 78–85 (2022)

Conclusions

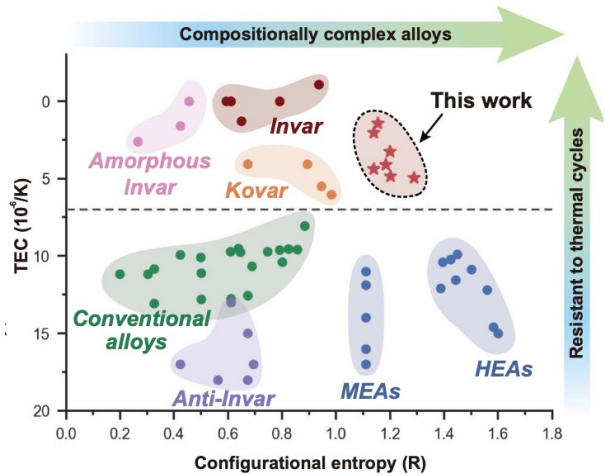
Advanced thermodynamic and ML approaches together with physics-based descriptors provide powerful tools to **explore and utilize high-dimensional configuration spaces**

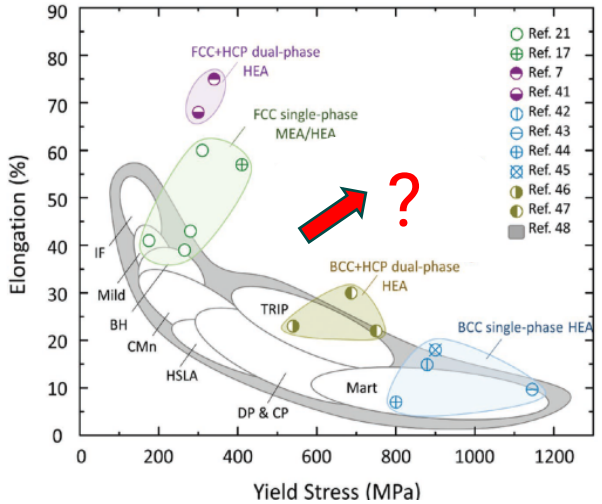
Workflows/simulation protocols to run these approaches become exceedingly complex

→ Our approach: **pyiron** as materials IDE



Enable **computational materials design in high-dimensional configuration spaces**





Thank you for your attention!

MAX-PLANCK-INSTITUT
FÜR EISENFORSCHUNG GmbH

ICAMS
INTERDISCIPLINARY CENTRE FOR ADVANCED MATERIALS SIMULATION

SFB 1394 Structural and Chemical Atomic Complexity

DFG Deutsche Forschungsgemeinschaft

erc European Research Council
ERC Advanced Grant SMARTMET

IMPRS SURMAT

MATERIALD1G1TAL

VDI

Federal Ministry of Education and Research

CLUSTER OF EXCELLENCE - EXC 1069
RESOLV
RUHR EXPLORES SOLVATION

Alexander von Humboldt
Stiftung/Foundation

Jan Janssen

Fritz Körmann

Tilmann Hickel

Developer Team

pyiron