Software tools for calculating materials properties in high-throughput (pymatgen, atomate, FireWorks)

Anubhav Jain
Energy Technologies Area
Lawrence Berkeley National Lab
Berkeley, CA

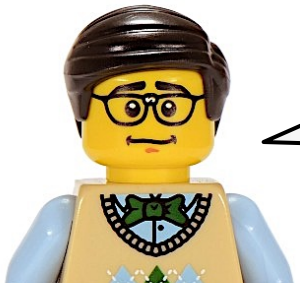
MP workshop 2017

Slides (already) posted to: http://www.slideshare.net/anubhavster

“Civilization advances by extending the number of important operations which we can perform without thinking about them.”

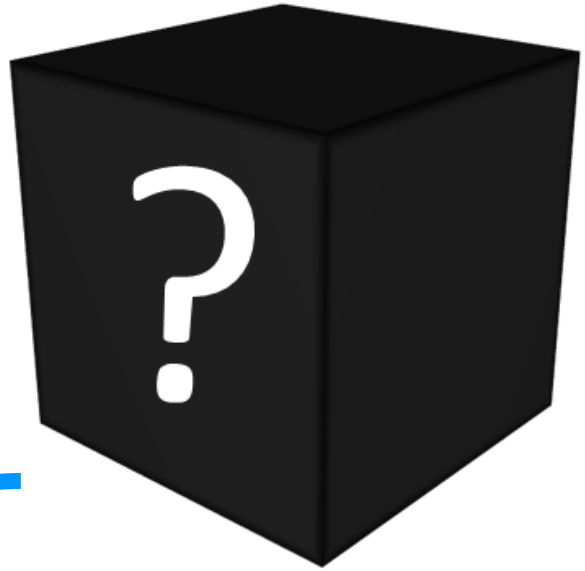
- Alfred North Whitehead

A "black-box" view of performing a calculation



What is the GGA-PBE elastic tensor of GaAs?

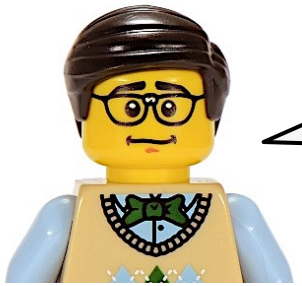
researcher



"something"

Results!

Unfortunately, the inside of the "black box" is usually tedious and "low-level"



researcher

What is the GGA-PBE elastic tensor of GaAs?

Input file flags
SLURM format
how to fix ZPOTRF?

lots of tedious, low-level work...

Results!

Calculations are labor intensive!

- Some of the things to do include:
  - set up the structure coordinates
  - write input files, double-check all the flags
  - copy to supercomputer
  - submit job to queue
  - deal with supercomputer headaches
  - monitor job
  - fix error jobs, resubmit to queue, wait again
  - repeat process for subsequent calculations in workflow
  - parse output files to obtain results
  - copy and organize results, e.g., into Excel
- This is often repeated for each and every run!

All the low-level steps can lead to errors!

Let's take a look at two alternate universes:

1

you

have coffee

copy files from previous simulation

edit 5 lines

run simulation, analyze data

2

you

forget coffee

copy files from previous simulation

edit 4 lines but forget LHFCALC=F

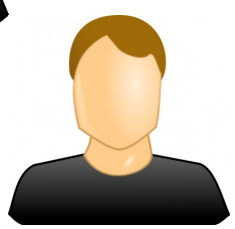
run simulation, looks fine at first, in a month you discover it was wrong

It is too easy to end up in the wrong timeline!

Today, it is difficult to learn and apply several computational procedures due to steep learning curve

Because of the multiple low-level steps that all need to be done correctly to apply computational methods, there is often a single group "expert" for each technique

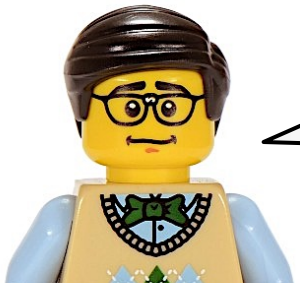
"Alice knows how to do charged defect calculations."



"Bob is the one who can properly converge GW runs."

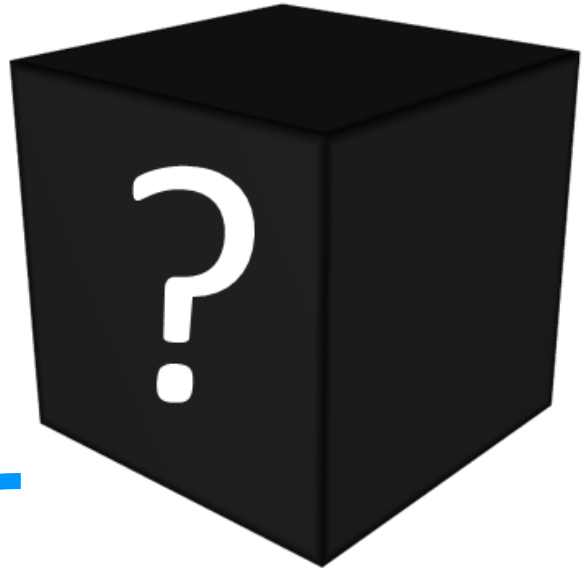
"Olga has all the scripts for phonon calculations."

What would be a better way?



What is the GGA-PBE elastic tensor of GaAs?

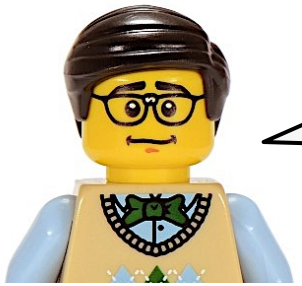
researcher



“something”

Results!

What would be a better way?



What is the GGA-PBE elastic tensor of GaAs?

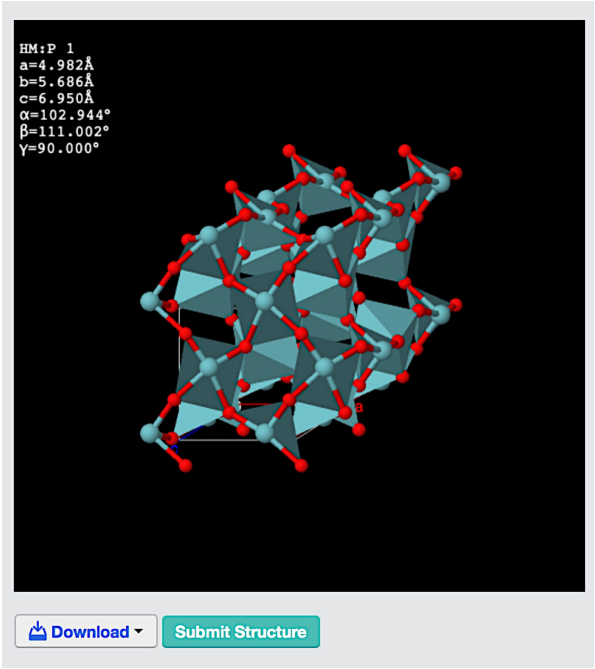
researcher

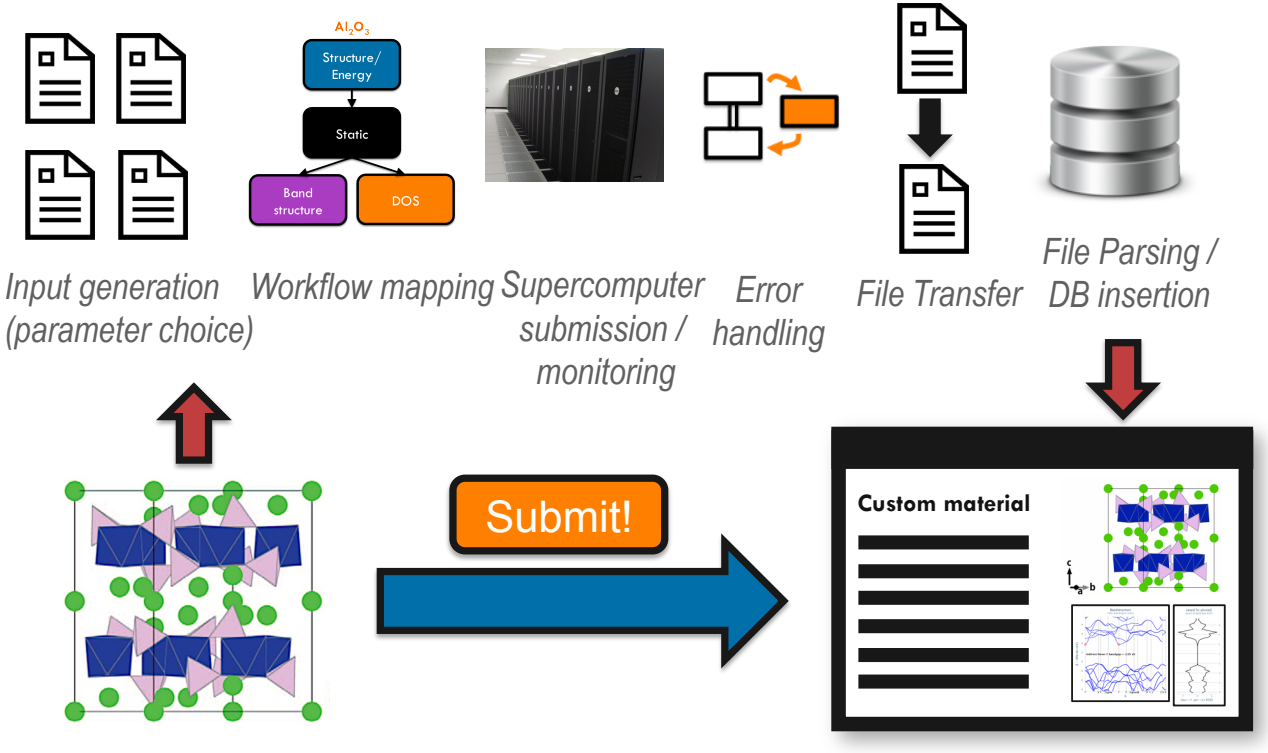
Submit!

a button

Results!

MPComplete on Materials Project works this way





www.materialsproject.org
"Crystal Toolkit"
Anyone can find, edit, and submit (suggest) structures

Currently, this feature is available for:

- structure optimization
- band structures
- elastic tensors

What if you want to do something more complicated?

- Start with all the known binary oxides
- Substitute oxygen with sulfur
- Re-optimize the structure and compute the band structure
- Plot the band gap of oxide vs the sulfide

- Rerun compounds with heavy elements with spin-orbit coupling enabled

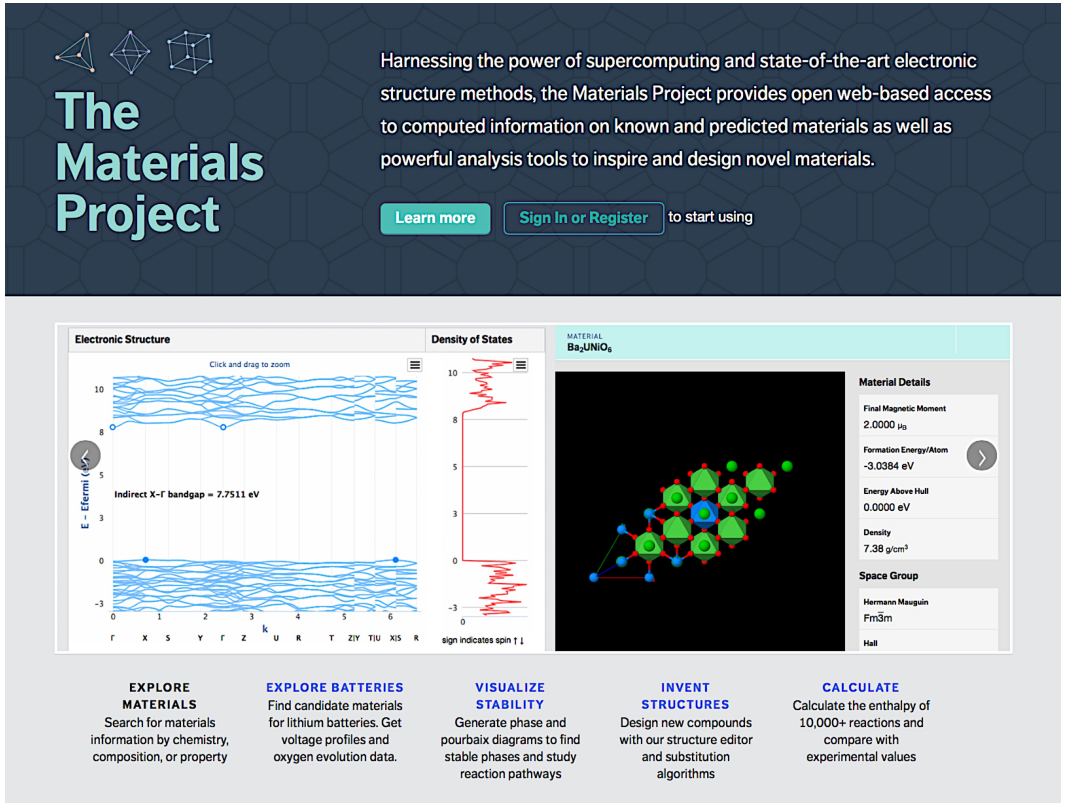
Now, a simple button won't do it anymore.

What you want is a high-level language for defining, running, and executing simulations.

With such a capability, one could do many applications.

Example: The Materials Project database

The Materials Project (http://www.materialsproject.org)



Jain*, Ong*, Hautier, Chen, Richards, Dacek, Cholia, Gunter, Skinner, Ceder, and Persson, APL Mater., 2013, 1, 011002. *equal contributions

free and open

>30,000 registered users around the world

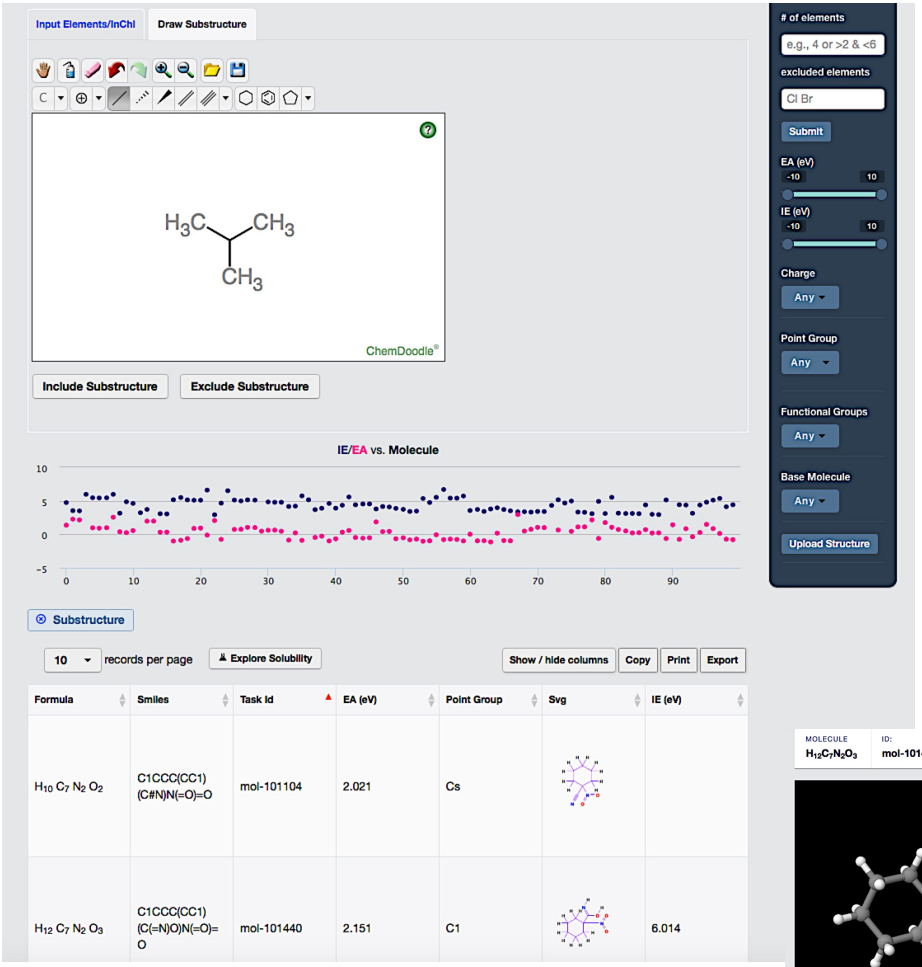
>65,000 compounds calculated

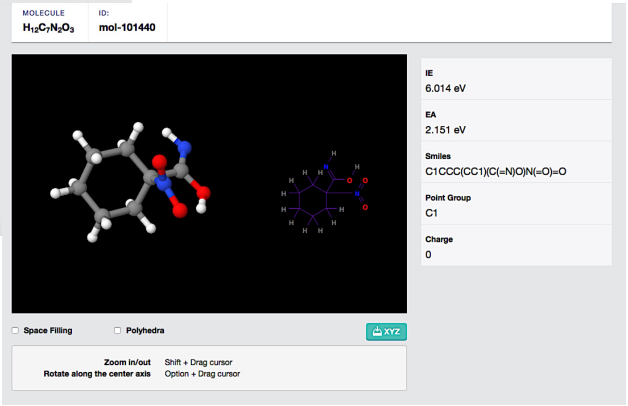
Data includes

- thermodynamic props.
- electronic band structure
- aqueous stability (E-pH)
- elasticity tensors
- piezoelectric tensors

>75 million CPU-hours invested = massive scale!

Example: The Electrolyte Genome





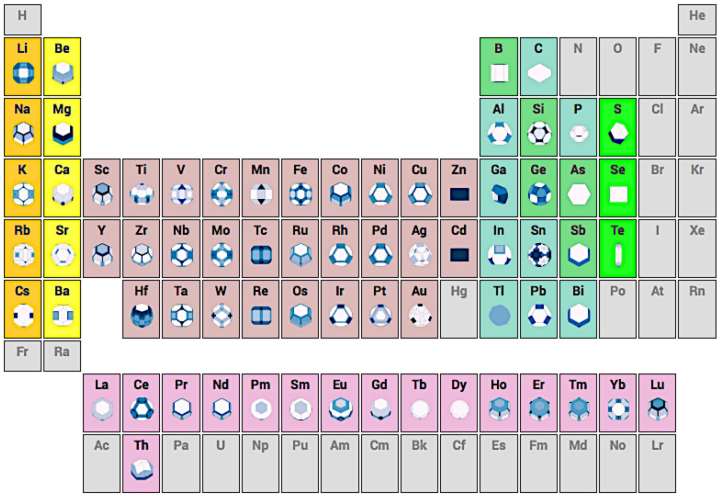
data on ~22,000 molecules (mainly geometry + IP/EA via full adiabatic calcs)

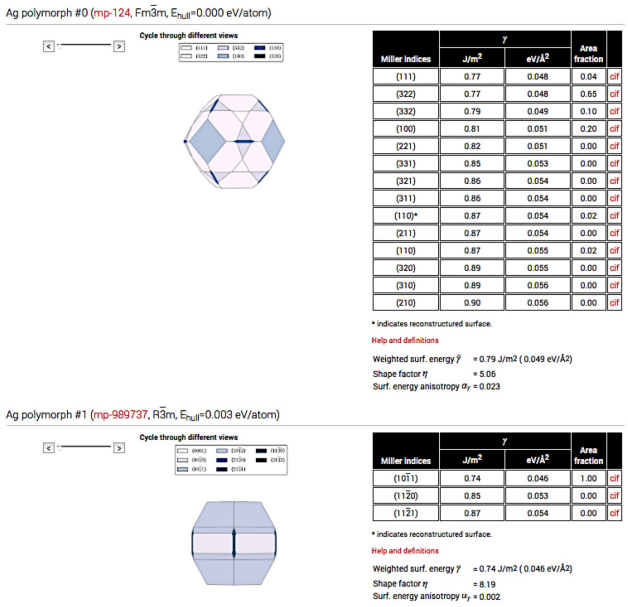
Also deployed on the Materials Project web site

L. Cheng, R.S. Assary, X. Qu, A. Jain, S.P. Ong, N.N. Rajput, et al., J. Phys. Chem. Lett. 6 (2015) 283–291.

X. Qu, A. Jain, N.N. Rajput, L. Cheng, Y. Zhang, S.P. Ong, et al., Comput. Mater. Sci. 103 (2015) 56–67.

Example: Crystalium (Ong / Persson)





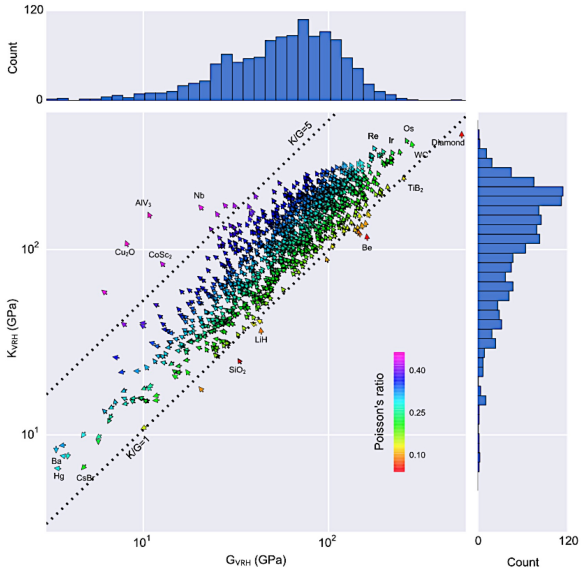
http://crystalium.materialsvirtuallab.org

surface energies for 142 polymorphs of 72 elements + rotatable Wulff shapes

computed & maintained by the Ong group (UC San Diego) with support from Persson Group (UC Berkeley)

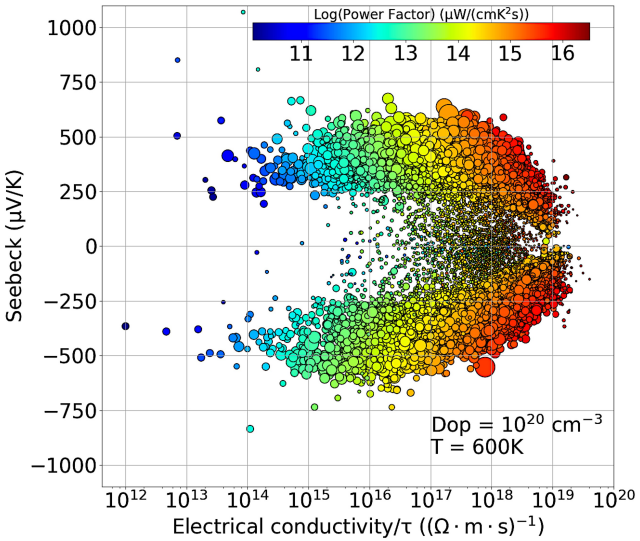
R. Tran, Z. Xu, B. Radhakrishnan, D. Winston, W. Sun, K. A. Persson, and S. P. Ong, Sci. Data, 2016, 3, 160080.

Example: Rapid data generation



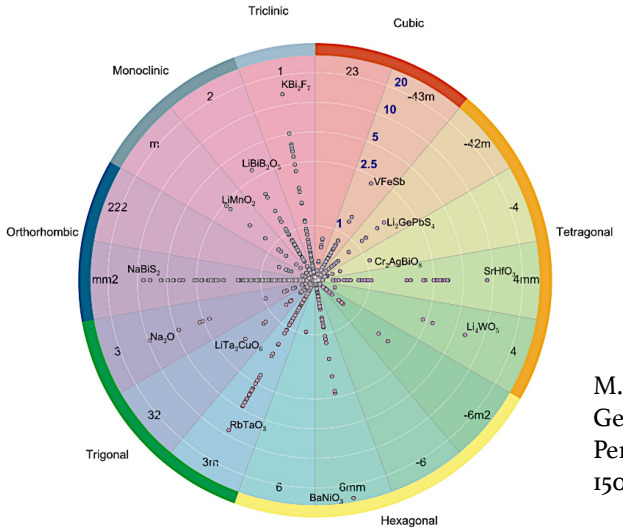
>4500 elastic tensors

M. De Jong, W. Chen, T. Angsten, A. Jain, R. Notestine, A. Gamst, M. Sluiter, C. K. Ande, S. Van Der Zwaag, J. J. Plata, C. Toher, S. Curtarolo, G. Ceder, K. a Persson, and M. Asta, Sci. Data, 2015, 2, 150009.



>48000 Seebeck coefficients + cRTA transport

Ricci, Chen, Aydemir, Snyder, Rignanese, Jain, & Hautier, Sci Data 2017, 4, 170085.



>900 piezoelectric tensors

M. de Jong, W. Chen, H. Geerlings, M. Asta, and K. A. Persson, Sci. Data, 2015, 2, 150053.

How can we design a simple yet powerful tool for performing simulations?

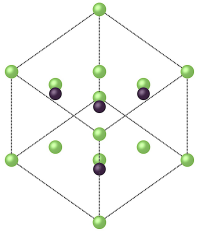
- First, we'll go through the four high-level steps in doing a theory study

- In each step, we'll talk about how software tools can help make you more productive and help you concentrate on things that require your brainpower, not mindless labor (e.g., manually resubmitting failed jobs)

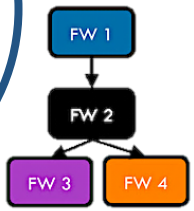
Steps for theory & simulation

material generation

(crystal, surface/interface, molecule, etc...)



simulation procedure



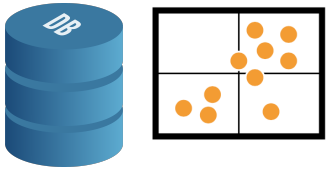
simulation parameters and sequence (workflow)

calculation & execution

job management, execution, error recovery

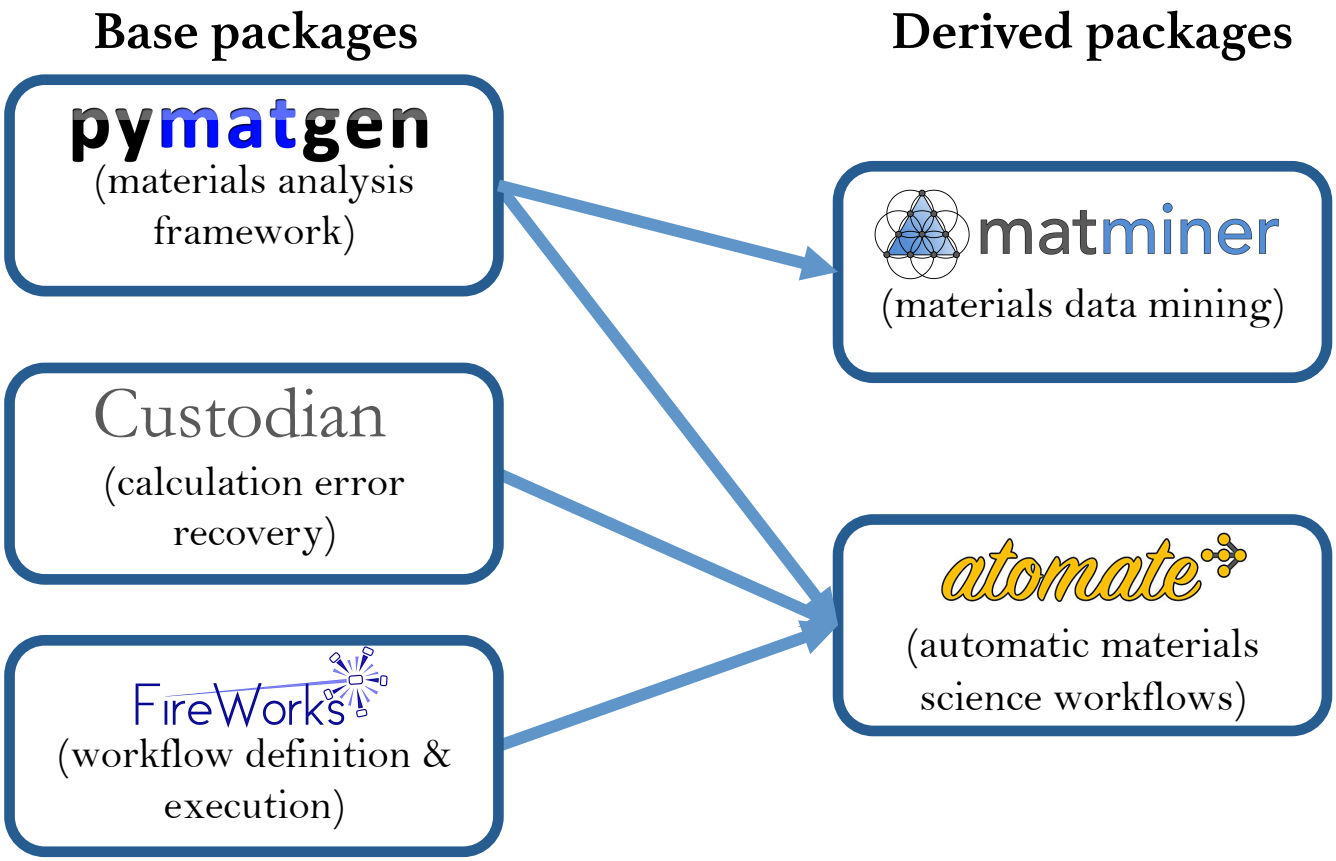
data analysis

databases, plotting, analysis, insight

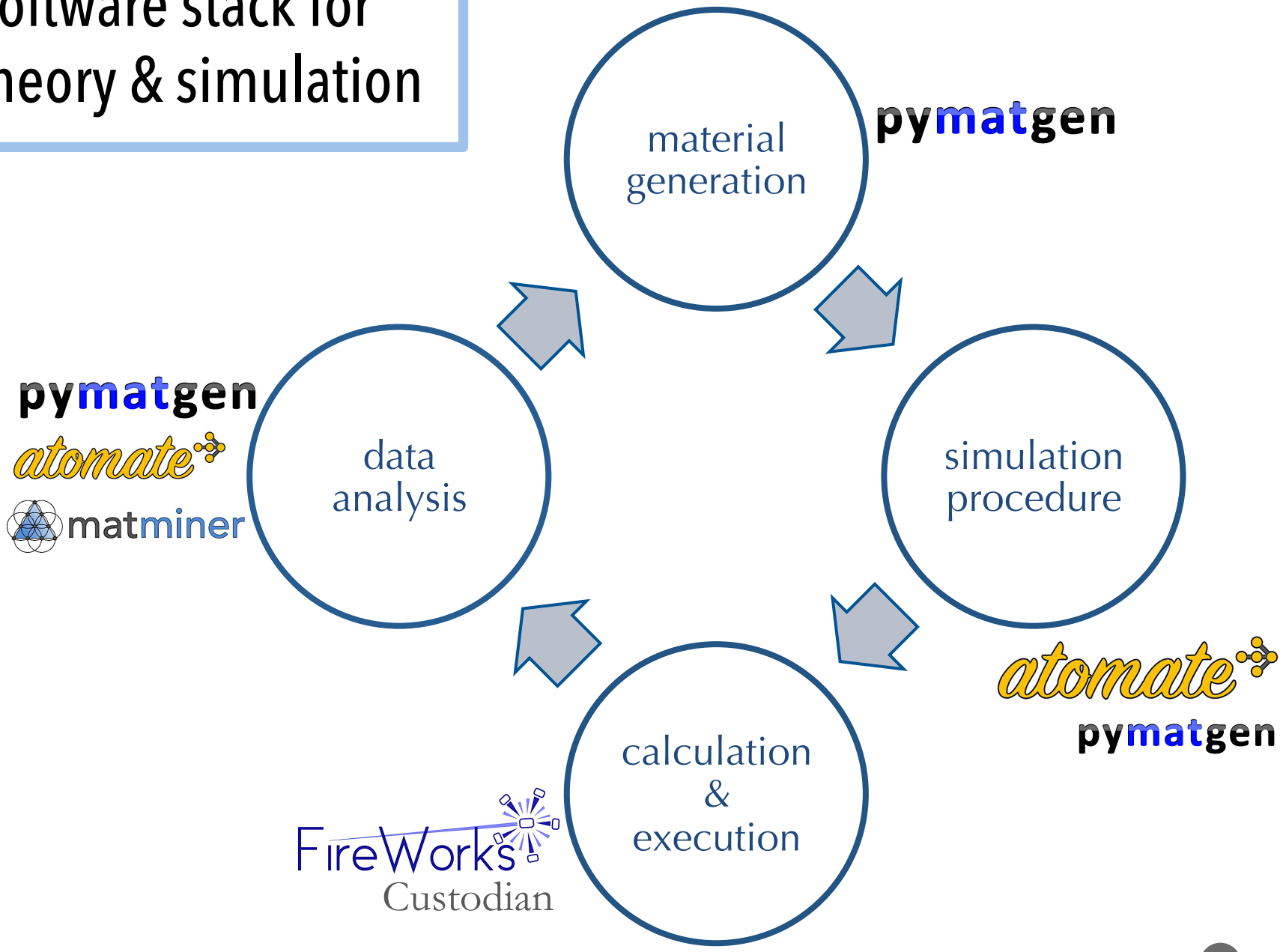


Software technologies for theory / calculation

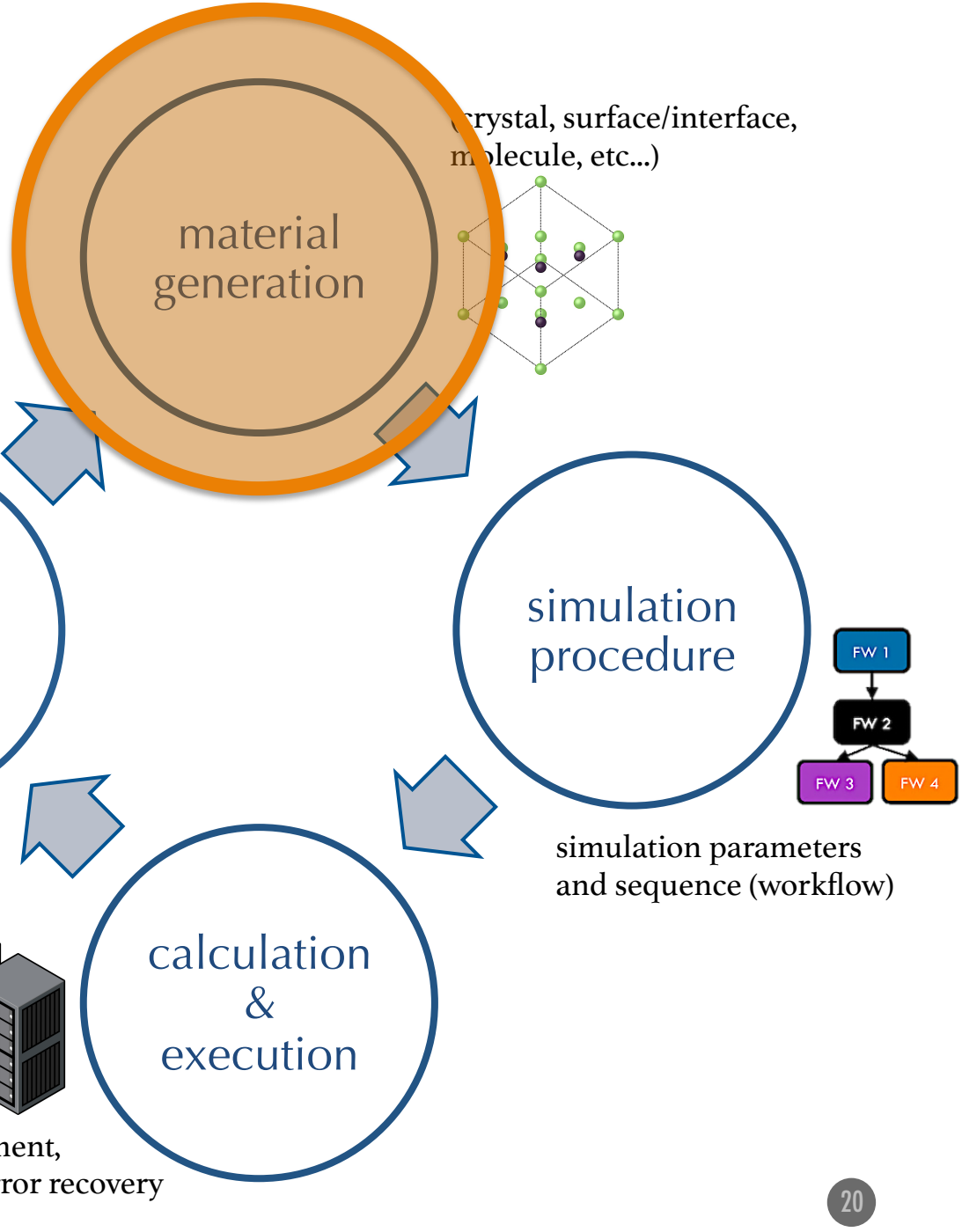
These are all open-source:



Software stack for theory & simulation



Steps for theory & simulation

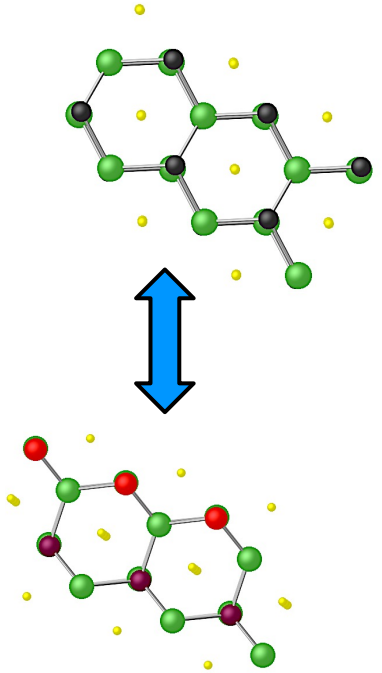


Step 1: materials generation

material generation

pymatgen

- In yesterday's tutorial, we already saw how pymatgen could help generate models for:
  - crystal structures
  - molecules
  - systems (surfaces, interfaces, etc.)
- Tools include:
  - order-disorder (shown at right)
  - interstitial finding
  - surface / slab generation
  - get structures from Materials Project
- Won't cover materials generation in this presentation!

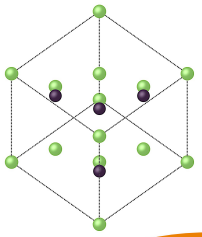


Example: Order-disorder
resolve partial or mixed occupancies into a fully ordered crystal structure (e.g., mixed oxide-fluoride site into separate oxygen/fluorine)

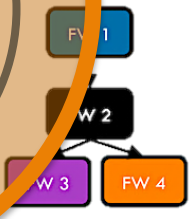
Steps for theory & simulation

material generation

(crystal, surface/interface, molecule, etc...)



simulation procedure



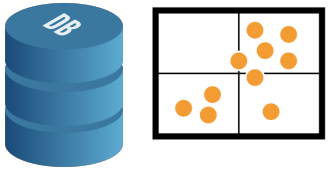
imulation parameters and sequence (workflow)

calculation & execution

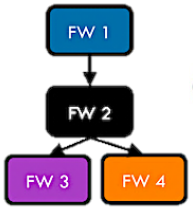
job management, execution, error recovery

data analysis

databases, plotting, analysis, insight



Step 2: simulation procedure



atomate
pymatgen

- Simulation procedure requires knowing:
  i. what sequence of calculations do I need to perform to compute my desired materials property?
  ii. for each calculation, what is a good set of parameters to use (e.g., functional, energy cutoff, k-point density, etc.)

- Item (i) is largely handled by atomate

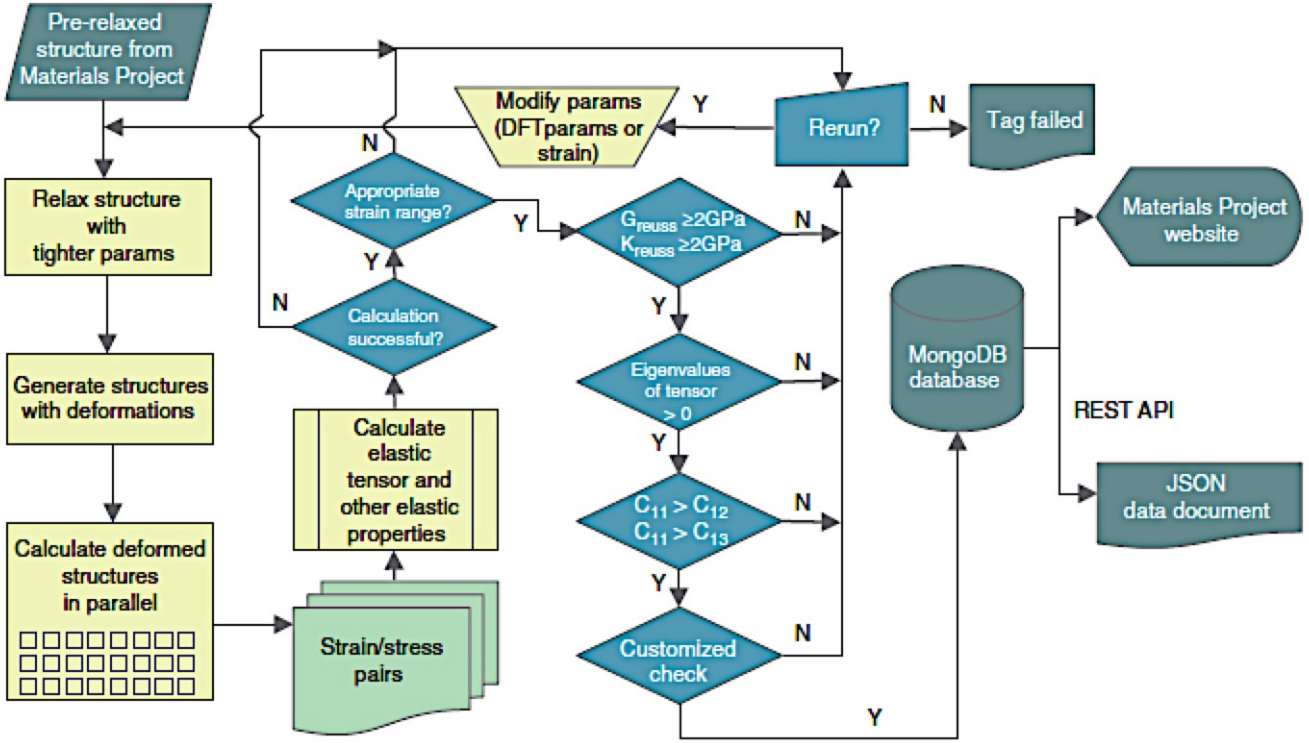
- Item (ii) is largely handled by pymatgen, e.g. in the VaspInputSet classes and won't be covered here – although atomate also contains some of this information

- Overall, we are trying to standardize and automate simulation procedures as much as possible

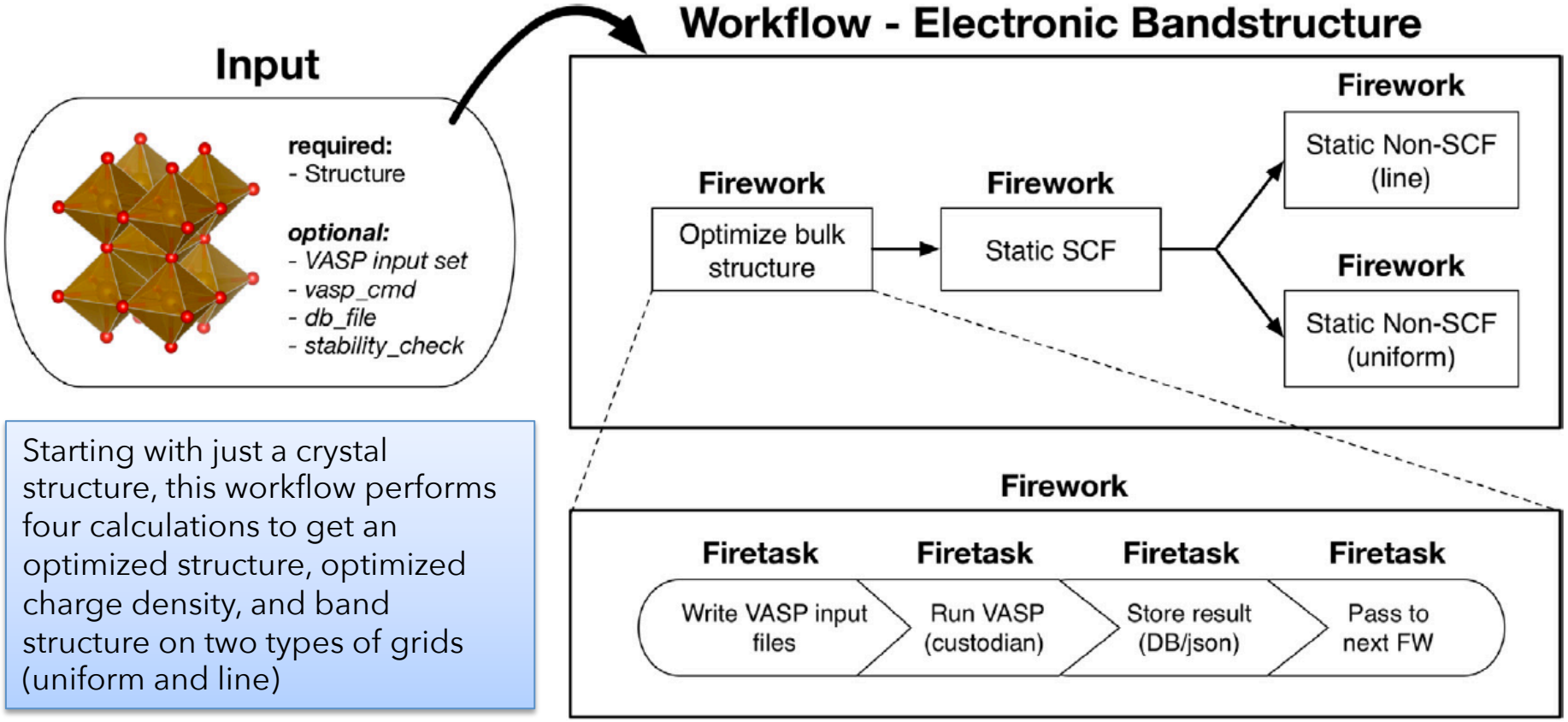
Atomate knows the sequence of calculations needed to compute many kinds of materials properties

atomate

quickly and automatically translate PI-style (minimal) specifications into well-defined FireWorks workflows



Each simulation procedure in atomate is composed of multiple levels of detail / abstraction

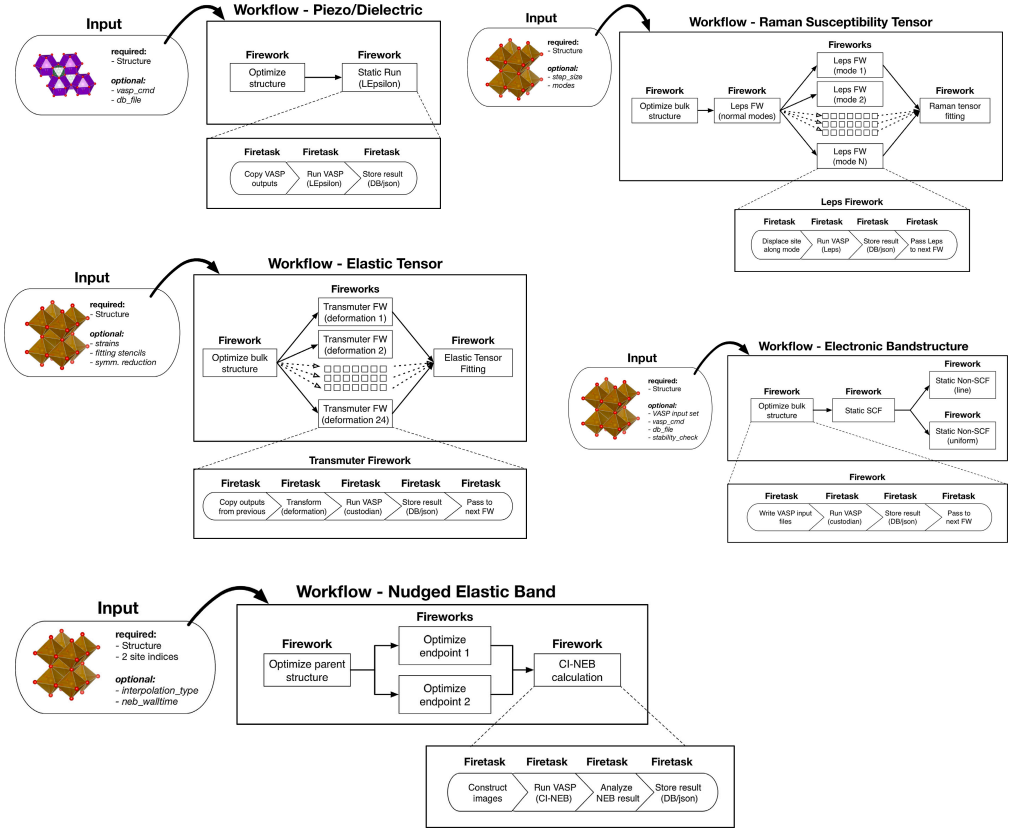


Starting with just a crystal structure, this workflow performs four calculations to get an optimized structure, optimized charge density, and band structure on two types of grids (uniform and line)

Workflow – complete set of calculations to get a materials property
Firework – one step in the Workflow (typically one DFT calculation)
Firetask – one step in a Firework

Many types of simulation procedures are already available!

Workflows currently available in atomate!



- band structure
- BoltzTraP transport
- spin-orbit coupling
- hybrid functional calcs
- elastic tensor
- piezoelectric tensor
- Raman spectra
- NEB
- GIBBS method
- QH thermal expansion
- AIMD

- FEFF method
- LAMMPS MD

atomate allows you to leverage the prior efforts and knowledge of many researchers

atomate +  → 

All past and present knowledge, from everyone in the group, everyone previously in the group, and our collaborators, about how to run calculations


K. Mathew


J. Montoya

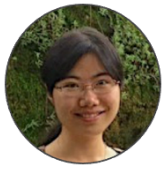

S. Dwaraknath


A. Faghaninia


B. Bocklund


T. Smidt


M. Aykol


H. Tang


I.H. Chu


M. Horton


J. Dagdalen


B. Wood


Z.K. Liu

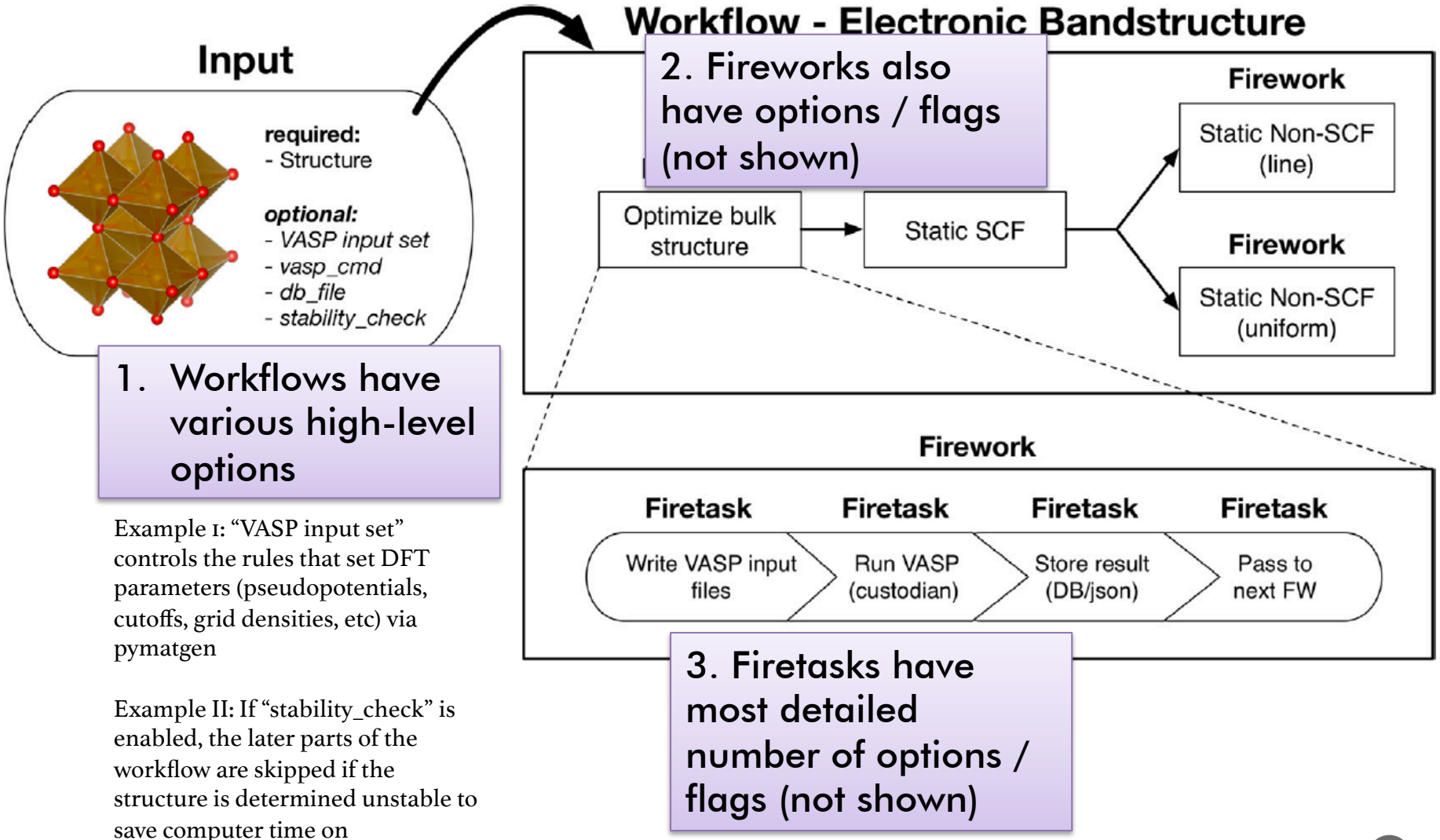

J. Neaton


S.P. Ong


K. Persson


A. Jain

Workflow parameters can be customized at multiple levels of detail



Example I: "VASP input set" controls the rules that set DFT parameters (pseudopotentials, cutoffs, grid densities, etc) via pymatgen

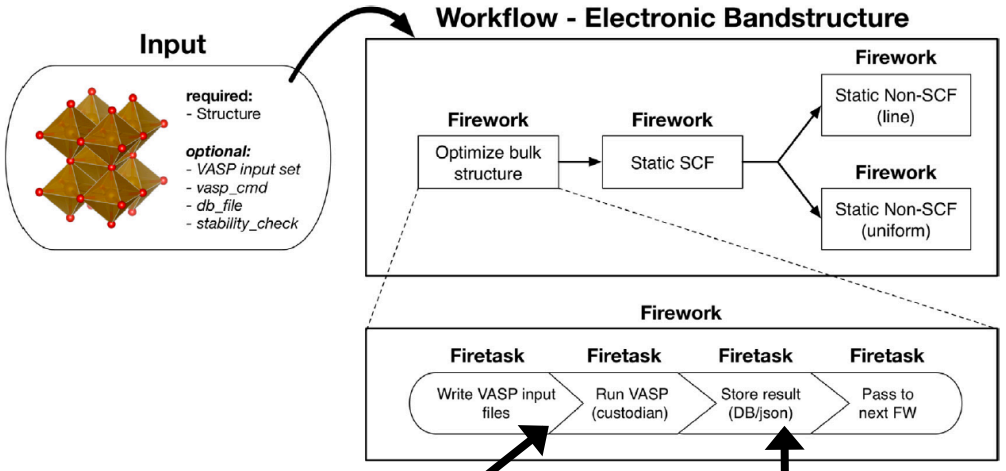
Example II: If "stability_check" is enabled, the later parts of the workflow are skipped if the structure is determined unstable to save computer time on uninteresting structures

"Powerups" can also add special features to a workflow, resulting in customization

- Atomate's philosophy: base workflow implementations should be clean
  - they only contain the necessary steps to perform a simulation

- What about user preferences like adding tags to certain compounds or tracking the progress of output files in the database?

- "Powerups" act like decorators – they take in a workflow and return a workflow with modified properties



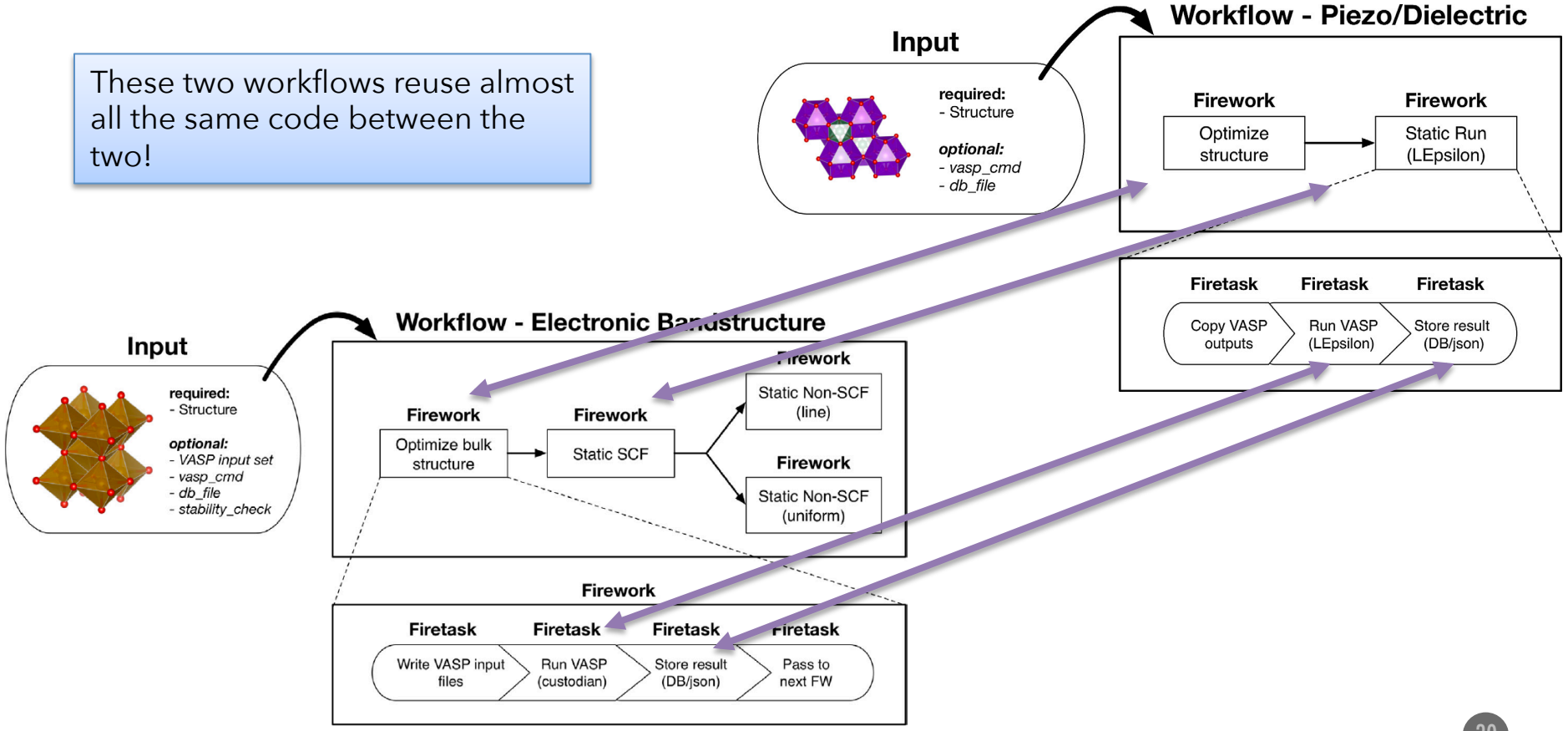
Example: the "add_modify_incar" will let you quickly override defaualt parameters in the VASP INCAR file.

Example: "add_tags" powerup will store user-defined tags/ metadata in the database along with the normal calculation information by adding parameters to the "store result" Firetask.

You can build workflows from scratch or reuse components to assemble workflows

Multiple workflows are built with the same components stacked together in different ways like Legos

These two workflows reuse almost all the same code between the two!



Atomate simulation procedures: the main ideas

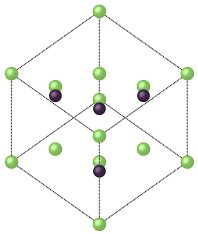
- Atomate allows you to start with a material (e.g., crystal structure) and get a full workflow of calculations needed to compute many types of materials properties

- Atomate tries to guess sensible defaults in terms of calculation parameters, but you can always customize these parameters using:
  - Workflow options
  - Firework options
  - Firetask options
  - powerups
  - going back to pymatgen library

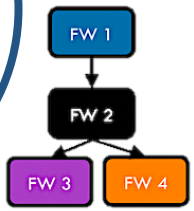
Steps for theory & simulation

material generation

(crystal, surface/interface, molecule, etc...)



simulation procedure



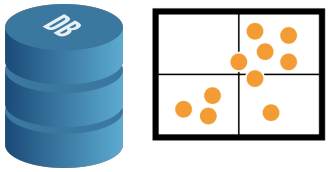
simulation parameters and sequence (workflow)

calculation & execution

job management, execution, error recovery

data analysis

databases, plotting, analysis, insight



Step 3: calculation & execution

FireWorks Custodian

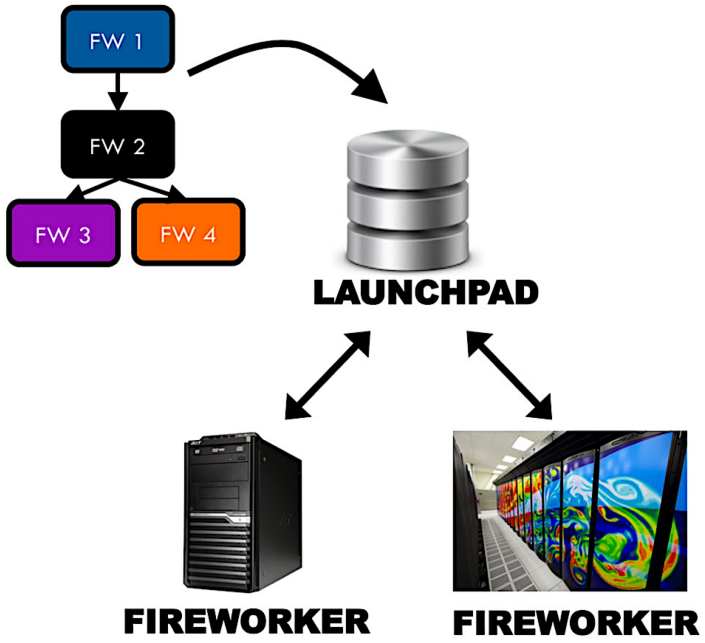
- Once you have the material and the simulation procedure (Workflow), you need to actually execute the workflow on your computing resource

- This includes tasks like:
  - submission to calculation queues
  - customization of any computing-specific parameters
    - e.g., path to VASP executable, number of CPUs to parallelize over
  - recovering from failures / job resubmission
  - coordinating jobs across computing centers
  - managing location of jobs
  - tracking the progress of jobs

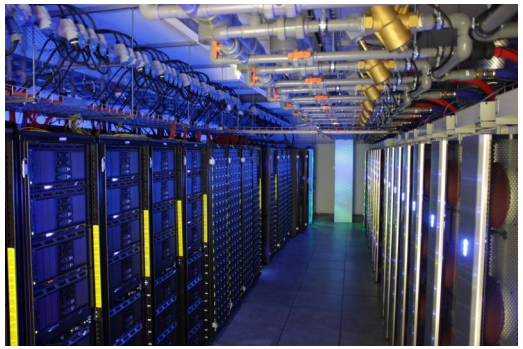
- Almost all of this is handled by FireWorks (custodian is used for encoding job recover procedures)

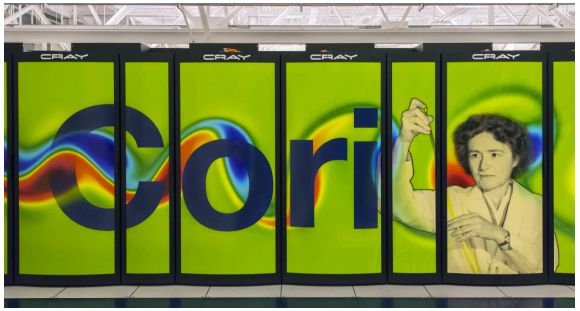
FireWorks – scientific workflow software

- Once you have a Workflow, FireWorks will execute it for you
- FireWorks is an open-source scientific workflow software
- Materials Project, JCESR, and other projects manage their runs with FireWorks
  - >1 million jobs and >100 million CPU-hours for internal projects alone
  - multiple computing clusters
- You can write any kind of workflow
  - e.g., FireWorks is used for graphics processing, machine learning, document processing, and protein folding
  - #1 Google hit for "Python workflow software", top 5 for general scientific workflow software
- Detailed tutorials are available that cover all the details



FireWorks allows you to write your workflow once and execute (almost) anywhere



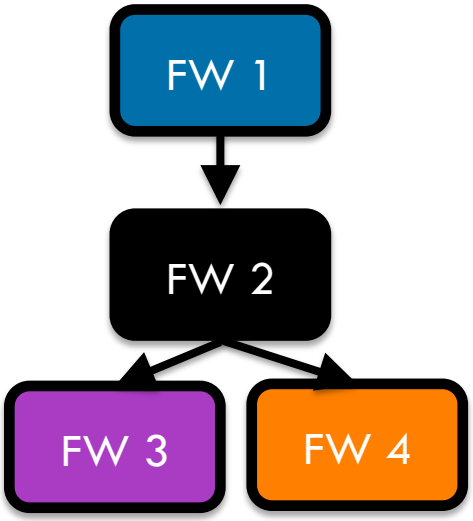


- Execute workflows locally or at a supercomputing center
- Queue systems supported
  - PBS
  - SGE
  - SLURM
  - IBM LoadLeveler
  - NEWT (a REST-based API at NERSC)
  - Cobalt (Argonne LCF)

Job execution (animation)

ROCKET LAUNCHER / QUEUE LAUNCHER

Directory 1 | Directory 2



LAUNCHPAD

Dashboard with status of all jobs

FireWorks

Workflow Dashboard

Newest Workflows

B4 C1 READY ID: 1573745

- B4_C1--Controller_add_Electronic_Structure_v2
- B4_C1--VASP_db_insertion
- B4_C1--GGA_optimize_structure_(2x)
- B4_C1--Add_to_SNL_database

Ba2 Fe1 Nb1 O6 READY ID: 1573739

- Ba2_Fe1_Nb1_O6--Controller_add_Electronic_Structure_v2
- Ba2_Fe1_Nb1_O6--VASP_db_insertion
- Ba2_Fe1_Nb1_O6--GGAU_optimize_structure_(2x)
- Ba2_Fe1_Nb1_O6--VASP_db_insertion
- Ba2_Fe1_Nb1_O6--GGA_optimize_structure_(2x)
- Ba2_Fe1_Nb1_O6--Add_to_SNL_database

Ba2 Fe1 Nb1 O6 READY ID: 1573733

- Ba2_Fe1_Nb1_O6--Controller_add_Electronic_Structure_v2
- Ba2_Fe1_Nb1_O6--VASP_db_insertion
- Ba2_Fe1_Nb1_O6--GGAU_optimize_structure_(2x)
- Ba2_Fe1_Nb1_O6--VASP_db_insertion
- Ba2_Fe1_Nb1_O6--GGA_optimize_structure_(2x)
- Ba2_Fe1_Nb1_O6--Add_to_SNL_database

Current Database Status

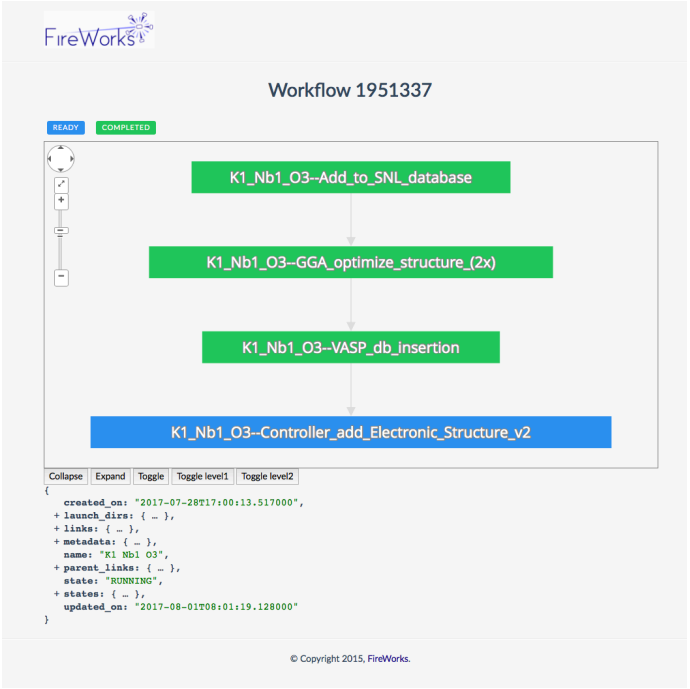
	Fireworks	Workflows
RUNNING	994	4,728
ARCHIVED	109,576	22,992
WAITING	167,134	0
FIZZLED	76,949	44,682
READY	24,199	18,197
RESERVED	799	0
COMPLETED	1,187,041	111,402
DEFUSED	6,588	3,656
TOTAL	1,573,280	205,657

Summary Reports

Get a report of all jobs for the past:

- 30 minutes
- 24 hours
- 7 days
- 30 days
- 6 months
- 24 months
- 10 years

For more reporting options, use the "lpad report --help" command line tool.



Job provenance and automatic metadata storage

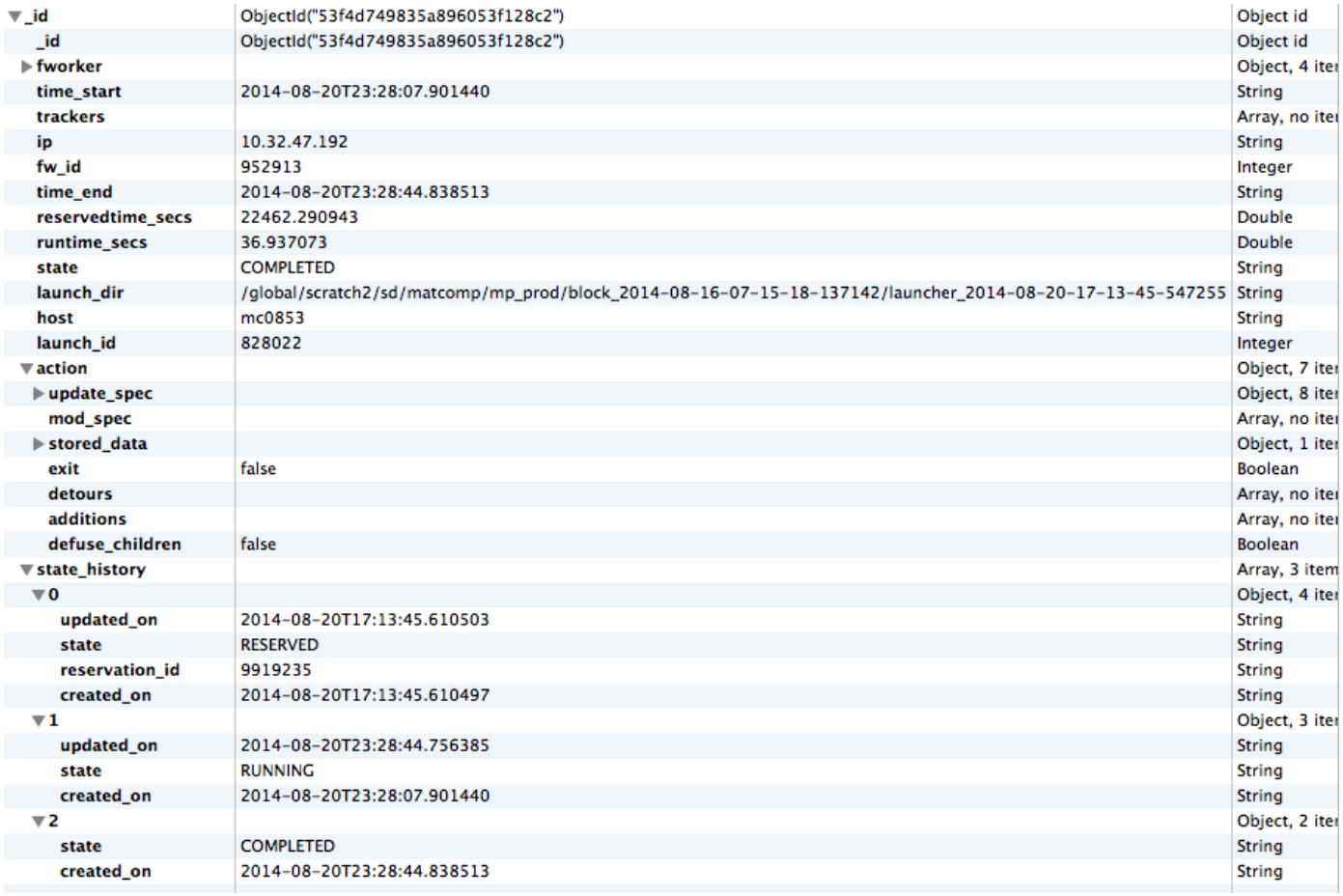
what machine
what time
what directory

what was the output

when was it queued

when did it start running

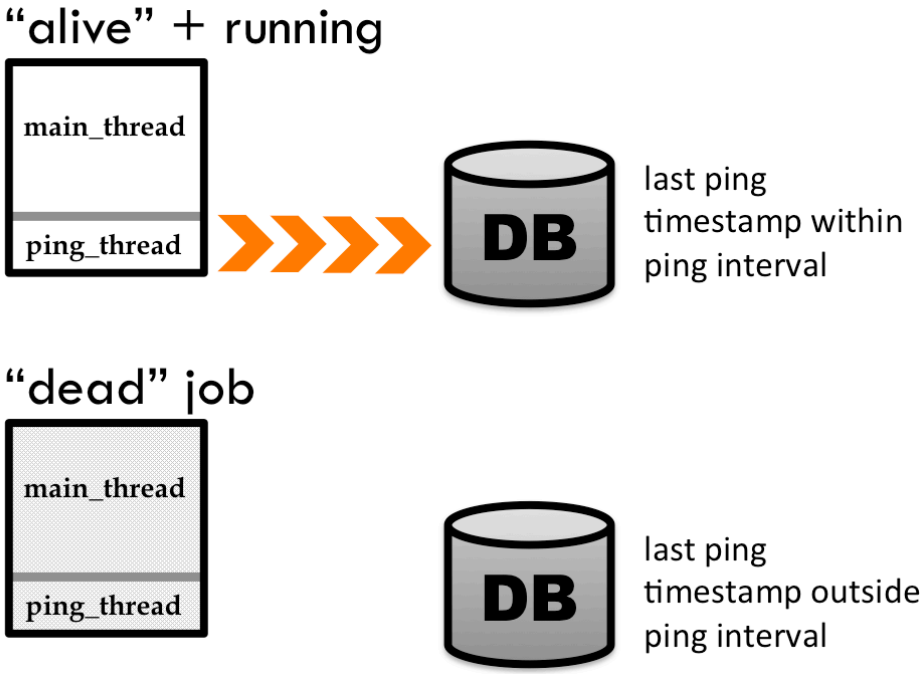
when was it completed



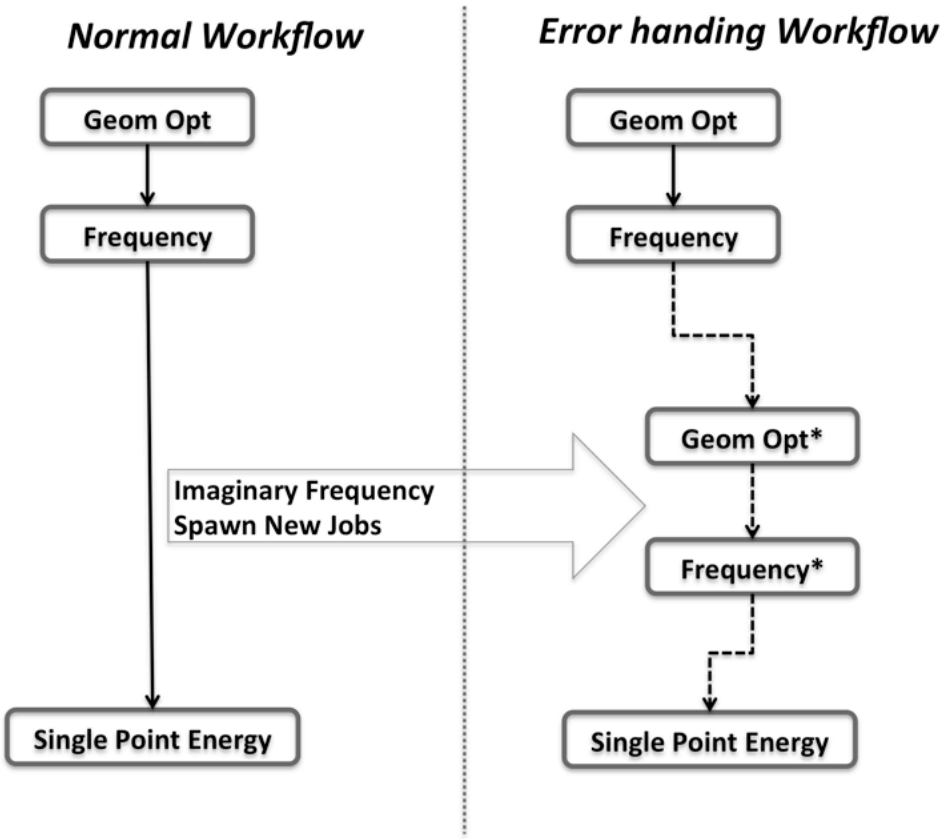
▼ _id	ObjectId("53f4d749835a896053f128c2")	Object id
_id	ObjectId("53f4d749835a896053f128c2")	Object id
▶ fworker		Object, 4 ite
time_start	2014-08-20T23:28:07.901440	String
trackers		Array, no ite
ip	10.32.47.192	String
fw_id	952913	Integer
time_end	2014-08-20T23:28:44.838513	String
reservedtime_secs	22462.290943	Double
runtime_secs	36.937073	Double
state	COMPLETED	String
launch_dir	/global/scratch2/sd/matcomp/mp_prod/block_2014-08-16-07-15-18-137142/launcher_2014-08-20-17-13-45-547255	String
host	mc0853	String
launch_id	828022	Integer
▼ action		Object, 7 ite
▶ update_spec		Object, 8 ite
mod_spec		Array, no ite
▶ stored_data		Object, 1 ite
exit	false	Boolean
detours		Array, no ite
additions		Array, no ite
defuse_children	false	Boolean
▼ state_history		Array, 3 item
▼ 0		Object, 4 ite
updated_on	2014-08-20T17:13:45.610503	String
state	RESERVED	String
reservation_id	9919235	String
created_on	2014-08-20T17:13:45.610497	String
▼ 1		Object, 3 ite
updated_on	2014-08-20T23:28:44.756385	String
state	RUNNING	String
created_on	2014-08-20T23:28:07.901440	String
▼ 2		Object, 2 ite
state	COMPLETED	String
created_on	2014-08-20T23:28:44.838513	String

Detect and rerun failures

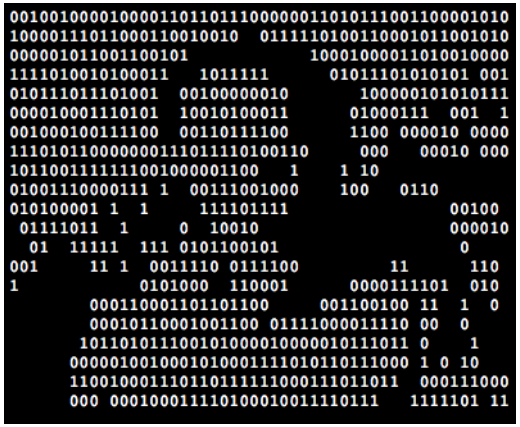
- All kinds of failures can be detected and rerun
  - Soft failures (job quits with error code)
  - hard failures (computing center goes down)
  - human errors



"Dynamic workflows" let you program intelligent, reactive workflows



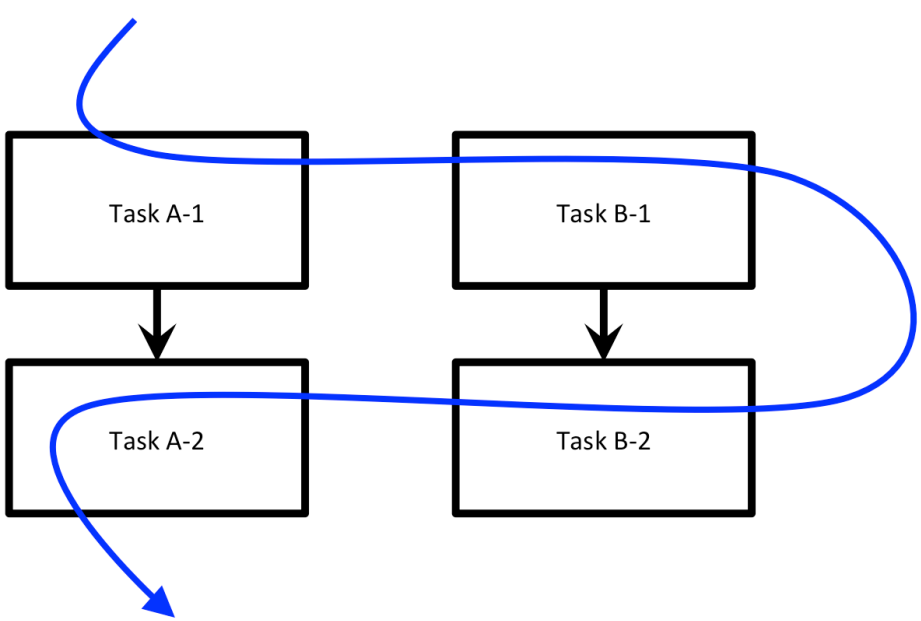
Xiaohui can replace himself with

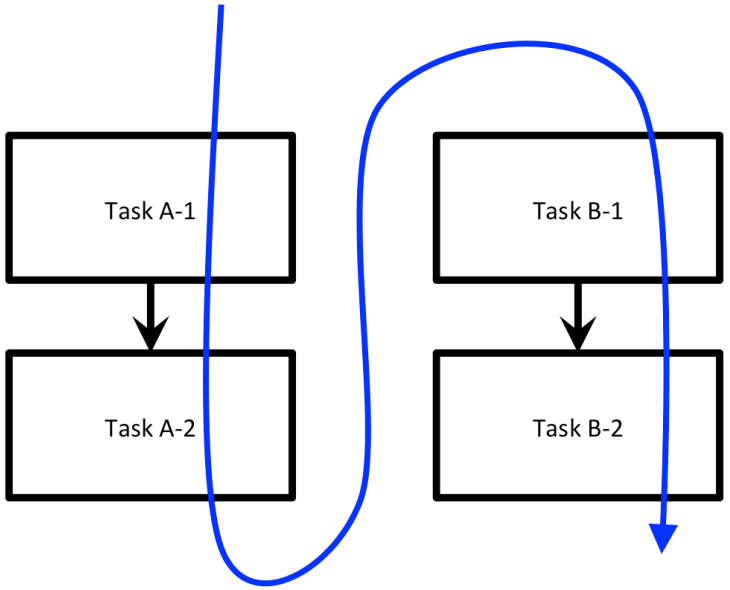


digital Xiaohui, programmed into FireWorks

Customize job priorities

- Within workflow, or between workflows
- Completely flexible and can be modified / updated whenever you want





Track output files remotely

- Can bring up the last few lines of each of your output files – and can be combined with queries / filters

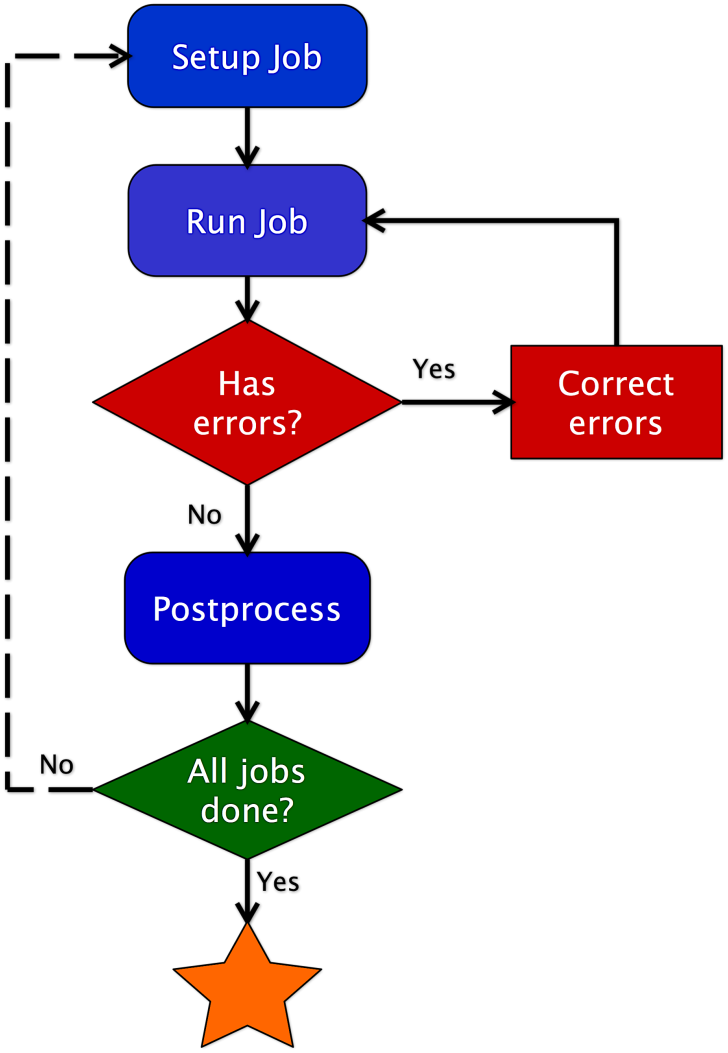
Now seems like a good time to bring up the last few lines of the OUTCAR of all failed jobs…

FireWorks workflow software: the main ideas

- FireWorks is used to execute Workflow objects at supercomputing centers
- It is very well-suited to high-throughput applications and has been used to execute millions of jobs and is well tested
- There are many features and advantages to using FireWorks as your workflow manager, which has made it one of the most popular scientific workflow software in use today

An aside on "custodian": instead of running VASP, have custodian run VASP on your behalf

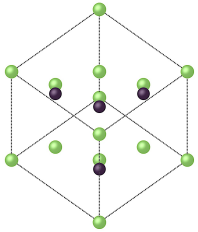
- Custodian can wrap around an executable (e.g., VASP)
  - i.e., run custodian instead of directly running VASP
- During execution, custodian will monitor output files and detect errors / problems
  - e.g., if ZPOTRF error detected, rerun with ISYM=0
  - ever-expanding library of fixes (currently over 30 fixes for VASP!)
- There is more you can do with custodian (e.g., simple workflows), but it won't be covered here
- Currently has fixes for: VASP, Q-Chem, NWChem



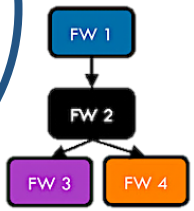
Steps for theory & simulation

material generation

(crystal, surface/interface, molecule, etc...)



simulation procedure



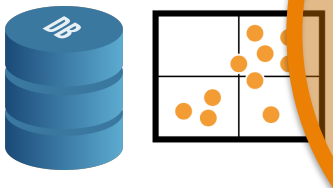
simulation parameters and sequence (workflow)

calculation & execution

job management, execution, error recovery

data analysis

databases, plotting analysis, insight



Step 4: data analysis

pymatgen
atomate
matminer

- Data analysis can involve:
  i. Creating databases of materials and their properties for easy searching and data retrieval
  ii. Conventional scientific analyses and scientific plotting (e.g., generating phase diagrams, plotting band structures)
  iii. Data mining methods

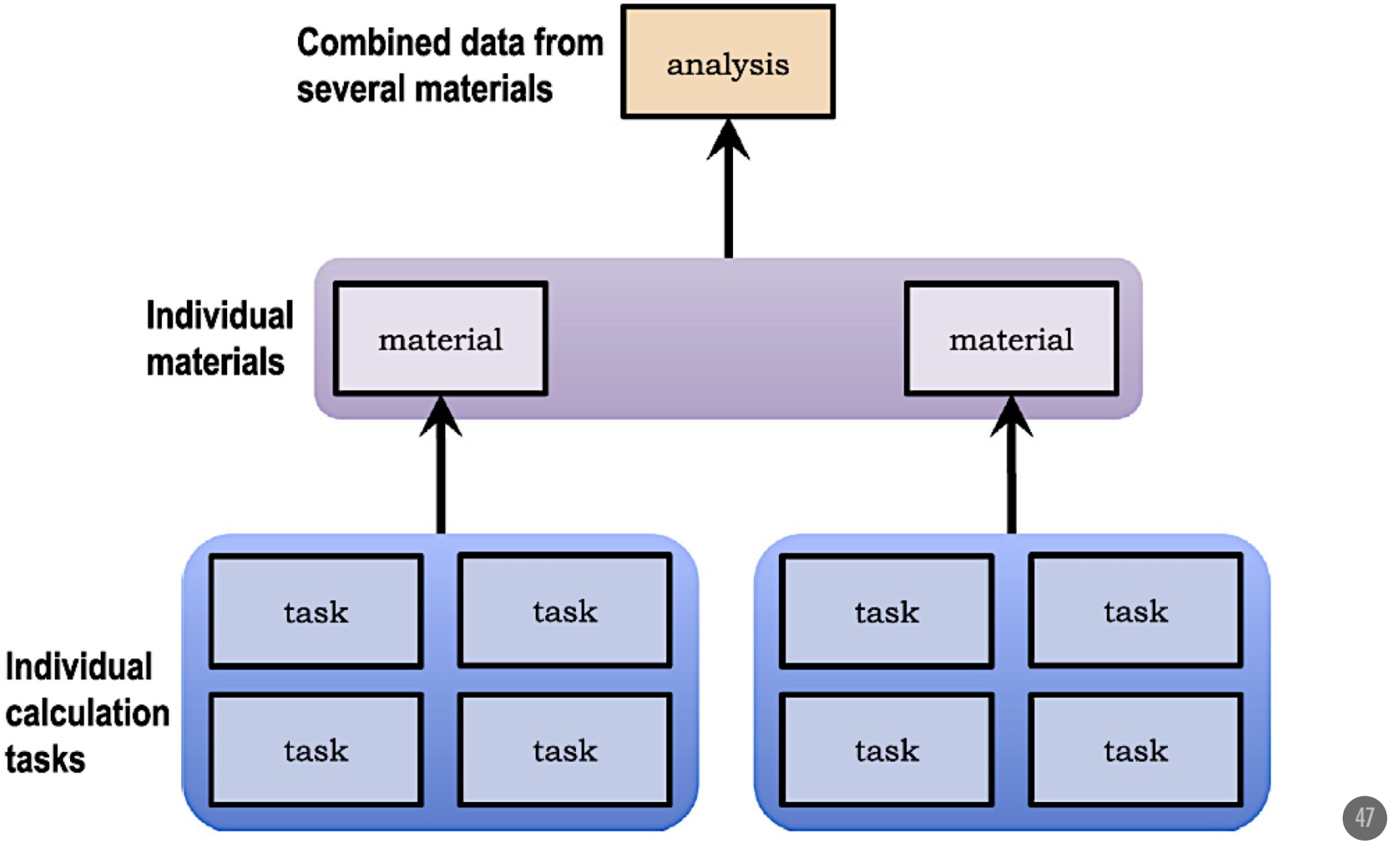
- Item (i) is largely handled by atomate

- Item (ii) is largely handled by pymatgen

- Item (iii) is largely handled by matminer

Atomate – builders framework

"Builders" start with base collections in a database and create higher-level collections that summarize information or add metadata



What kinds of builders are available in atomate?

- "Materials builder" – combine all calculations ("tasks") on a single crystal (as determined automatically by structure matching algorithms) into a single document
  - e.g., combine data from band structure, dielectric, and elastic constant workflows performed on the same material into a single report on that material that contains all information

- Add new information based on algorithmic analysis to the database, e.g. structure dimensionality or phase stability

- Merge external information such as user-defined tags, experimental data (from a spreadsheet), or data from Materials Project to the computational report of a material

- If you are using atomate without using builders, you are missing out on one of the most useful features!

Data analysis – the role of atomate

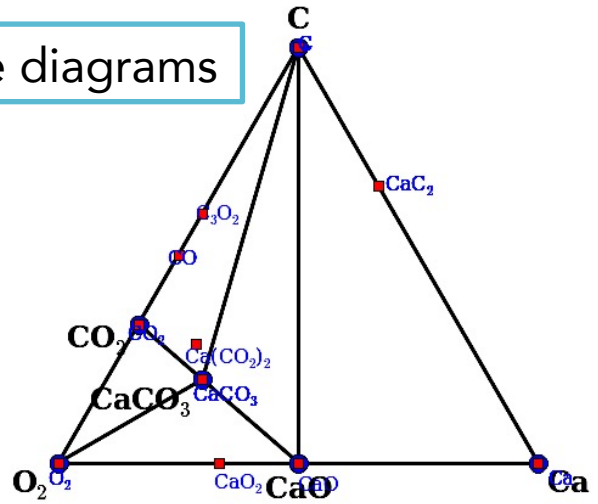
- Atomate helps organize computational data into databases

- However, atomate does not do scientific analysis or plotting of the data – it can just generate and grab the data (often as pymatgen objects)

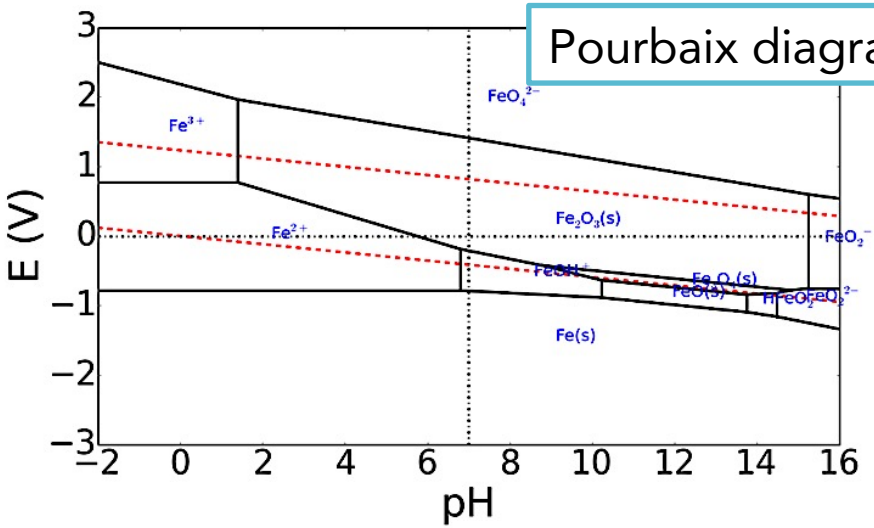
- Pymatgen is used to analyze the data
  - we won't cover pymatgen capabilities in depth since it is covered elsewhere in the workshop

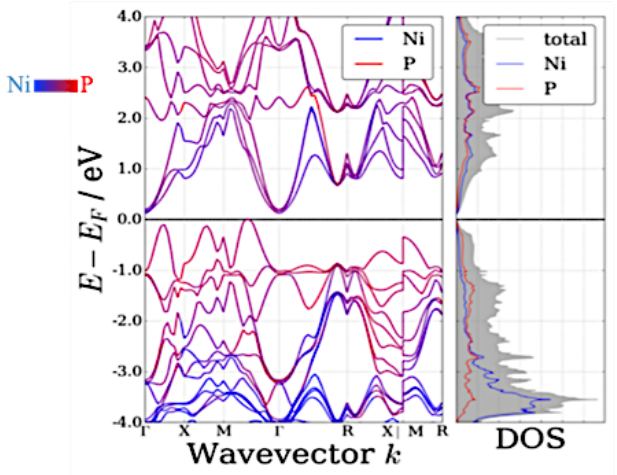
pymatgen – examples of analyses

phase diagrams

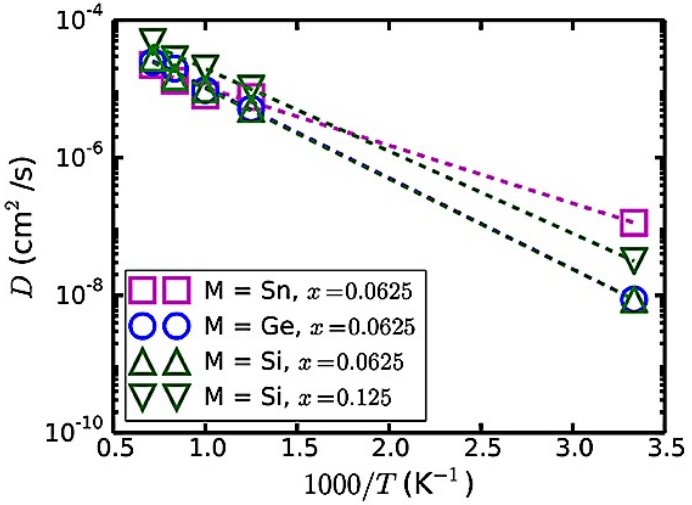


Pourbaix diagrams



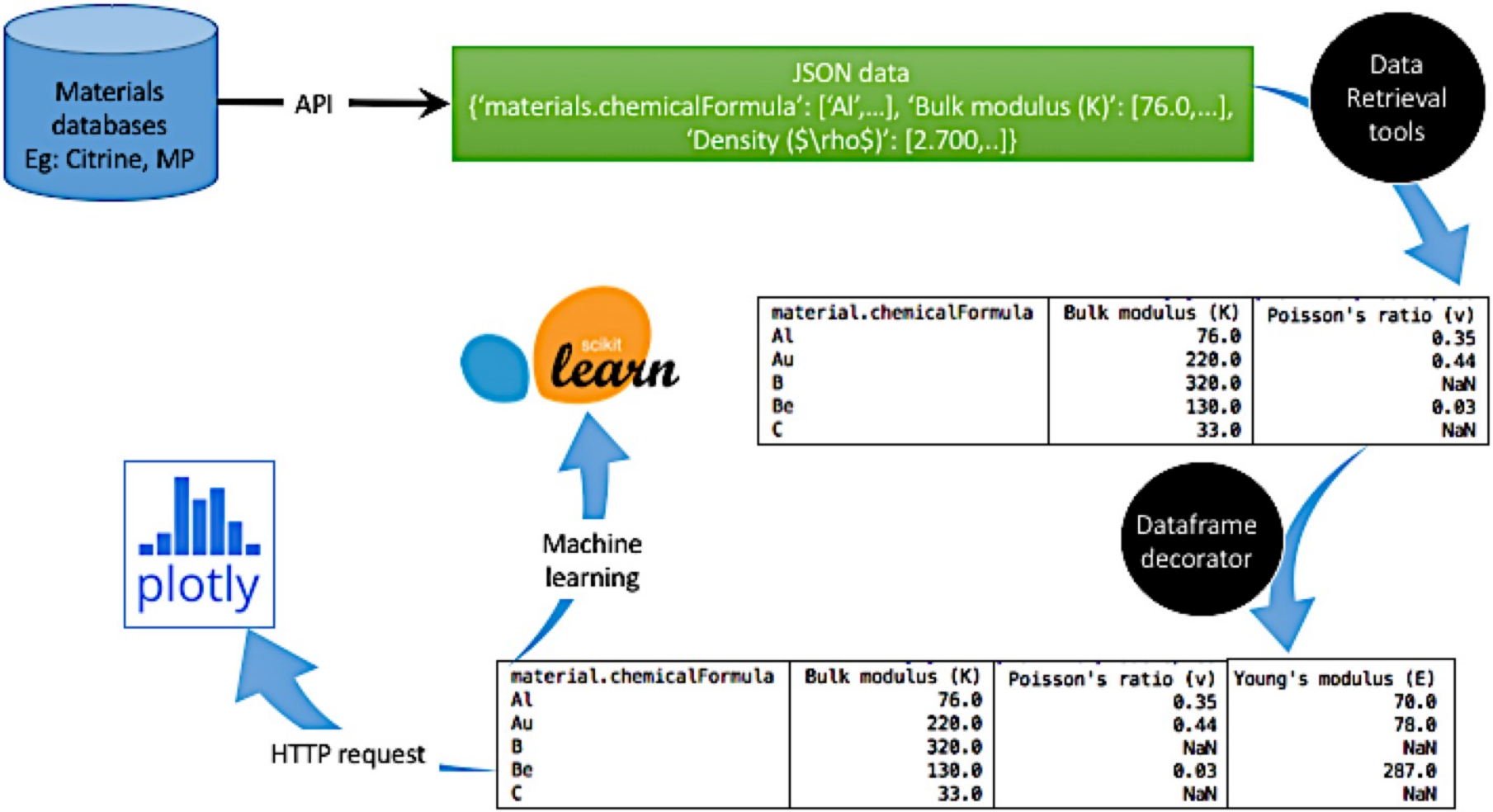


band structure analysis



diffusivity from MD

matminer helps enable data mining studies – currently in (open) pre-release



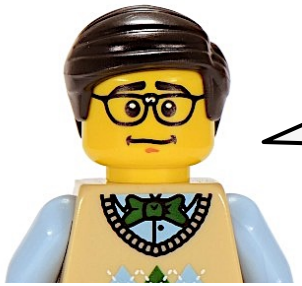
Data analysis: the main ideas

- Atomate generates databases that summarize your calculation results. Additionally The *builders* framework generates document collections that summarize data on a particular material from multiple calculations or from additional information.

- Pymatgen is a powerful software for doing conventional scientific analyses with the computational data

- Matminer, currently in pre-release, is software for doing data mining style analyses of materials

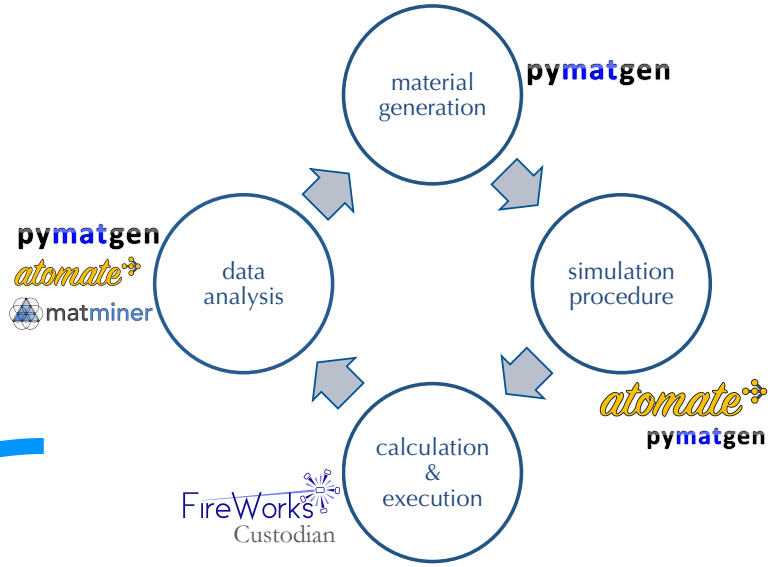
Doing simulations programmatically can become second nature – and very fast / automatic / scalable



What is the GGA-PBE elastic tensor of GaAs?

researcher

Results!



software infrastructure

Next steps

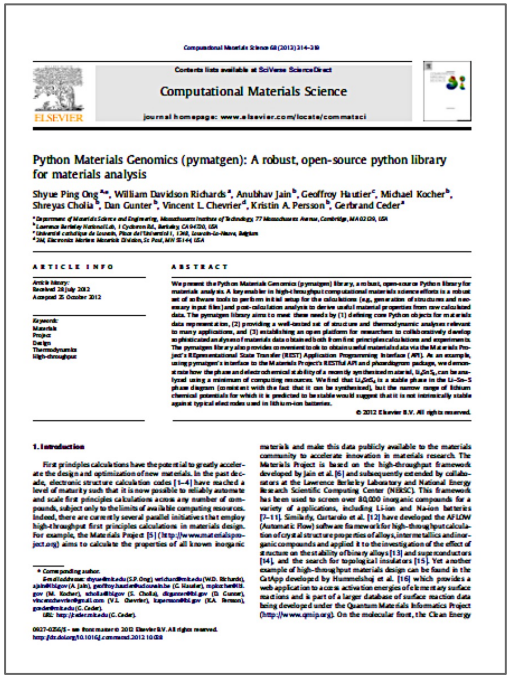
- This purpose of this presentation was to explain the benefits of the software tools available to you

- You will need to take more steps to actually gain use out of the software

- There are multiple resources to help you with this!

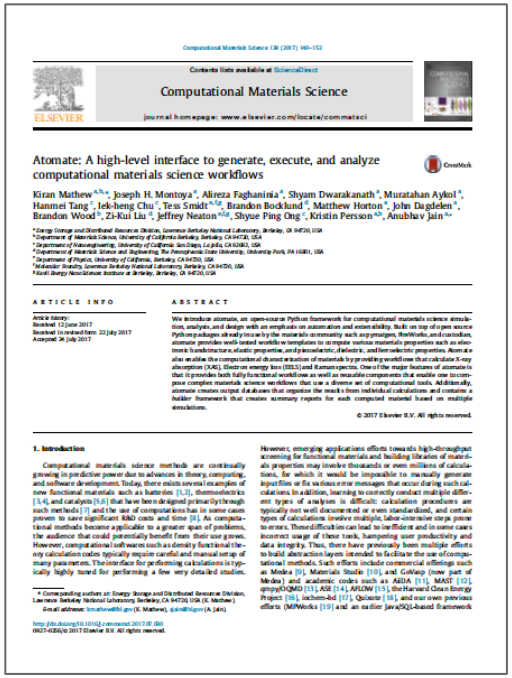
For general information / overview / citing

- There are papers on the software tools for general information (and for citation purposes)



Ong, S. P. et al. Python Materials Genomics (pymatgen): A robust, open-source python library for materials analysis. Comput. Mater. Sci. 68, 314–319 (2013).

Jain, A. et al. FireWorks: a dynamic workflow system designed for high-throughput applications. Concurr. Comput. Pract. Exp. 22, 5037–5059 (2015).



Mathew, K. et al. Atomate: A high-level interface to generate, execute, and analyze computational materials science workflows. Comput. Mater. Sci. 139, 140–152 (2017).

For installing / usage / examples

- The online documentation is best for practical usage
  - www.pymatgen.org
  - https://materialsproject.github.io/fireworks/
  - https://materialsproject.github.io/custodian/
  - https://hackingmaterials.github.io/atomate/
  - https://hackingmaterials.github.io/matminer/
- The online documentation includes installation, examples, tutorials, and descriptions of how to use the code
- If you want to do "everything", suggest starting with atomate and going from there

For help / questions

- Did you try your best to use the software but are having trouble?

- The various codebases have official channels for getting help
  - https://groups.google.com/forum/#!forum/pymatgen
  - https://groups.google.com/forum/#!forum/fireworkflows
  - https://groups.google.com/forum/#!forum/atomate
  - https://groups.google.com/forum/#!forum/matminer
  - for custodian, use Github “issues”

- Some of these lists have resolved >100 user questions, all essentially on volunteer time!

Questions? Comments?

Thanks to everyone that contributed to this software and all those who helped support / justify its development through citation!