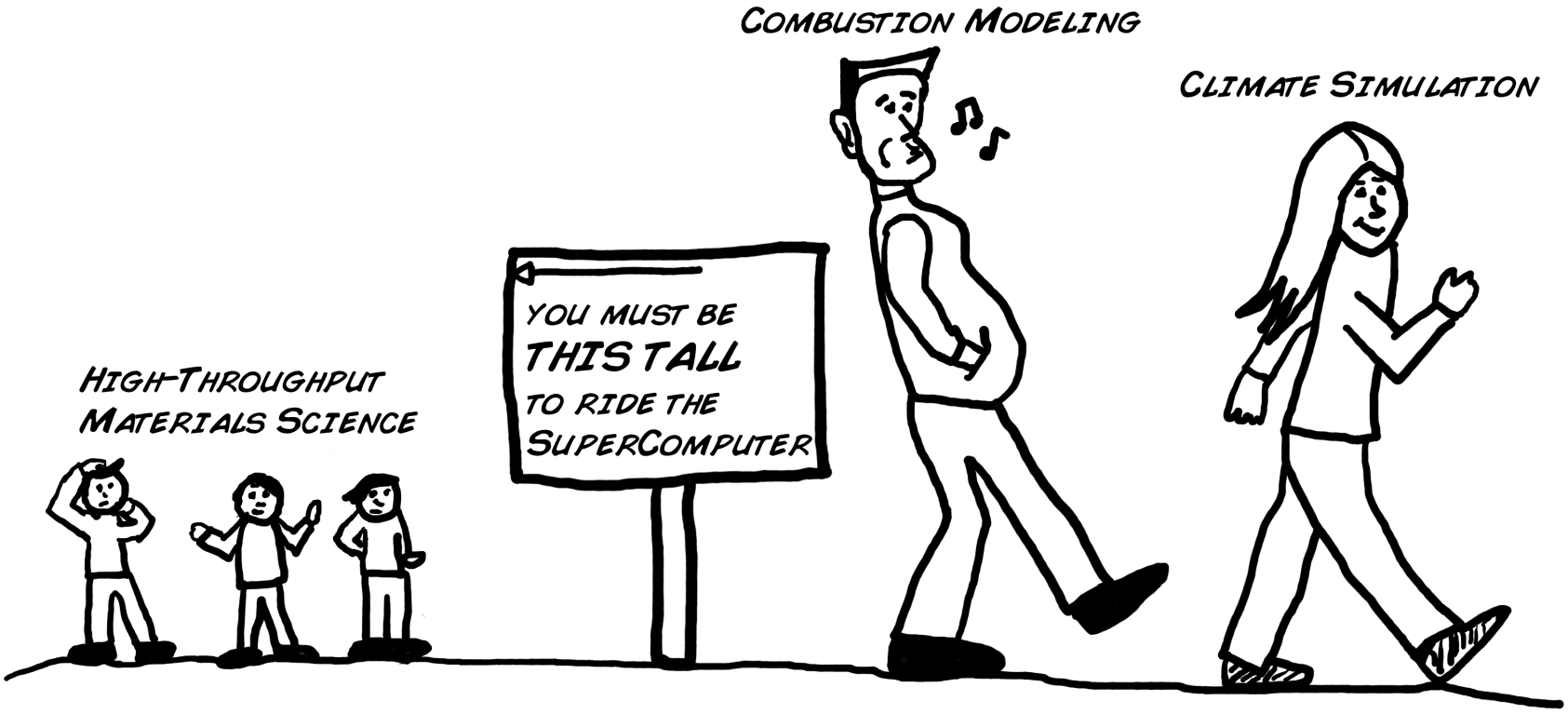
The Materials Project, FireWorks, and high-throughput computing at NERSC

NERSC User Day | Feb. 2014

Anubhav Jain
Energy & Environmental Technologies
Berkeley Lab

Should big computers run small calculations?



(a) No, because only massively parallel simulations require large computers

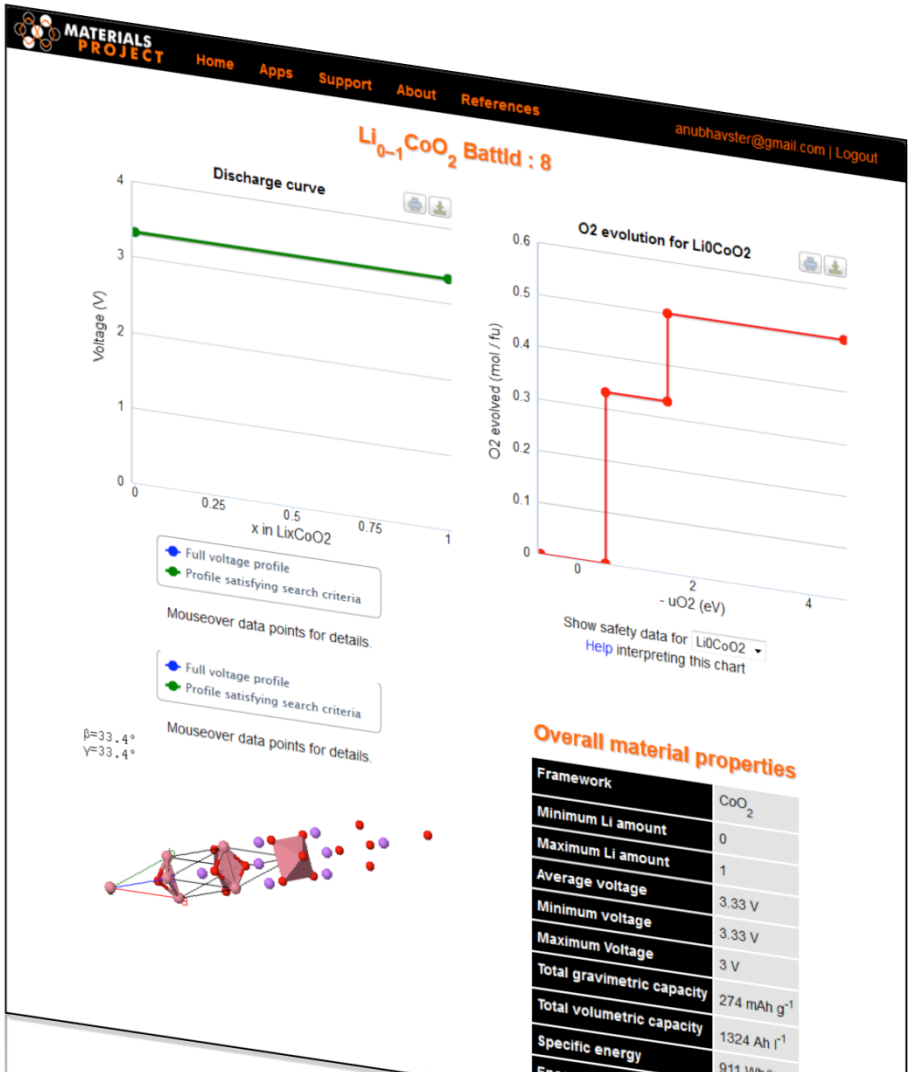
(b) No, because massively parallel jobs are more important than smaller jobs

(c) Yes - but small jobs should be second class citizens

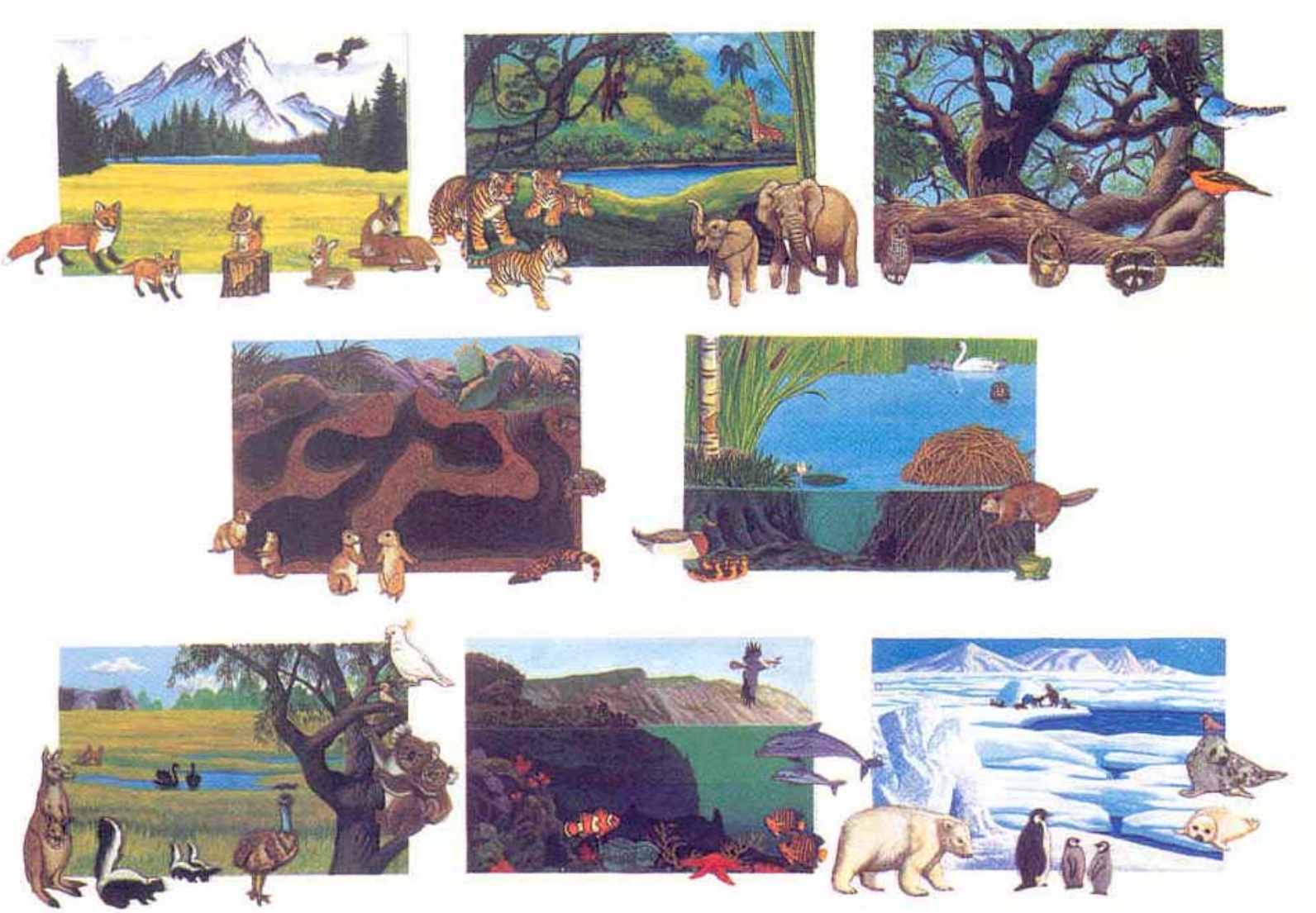
(d) Yes, and small jobs should have equal rights!

The Materials Project requires lots of computing, but not massively parallel

- 20+ million CPU-hours so far
  - 200,000+ individual electronic structure calculations
  - more is always needed
- 50+TB disk so far
  - always hitting disk limits
- 5600+ users so far
  - they need more computed data!



We built our own workflow software – why?

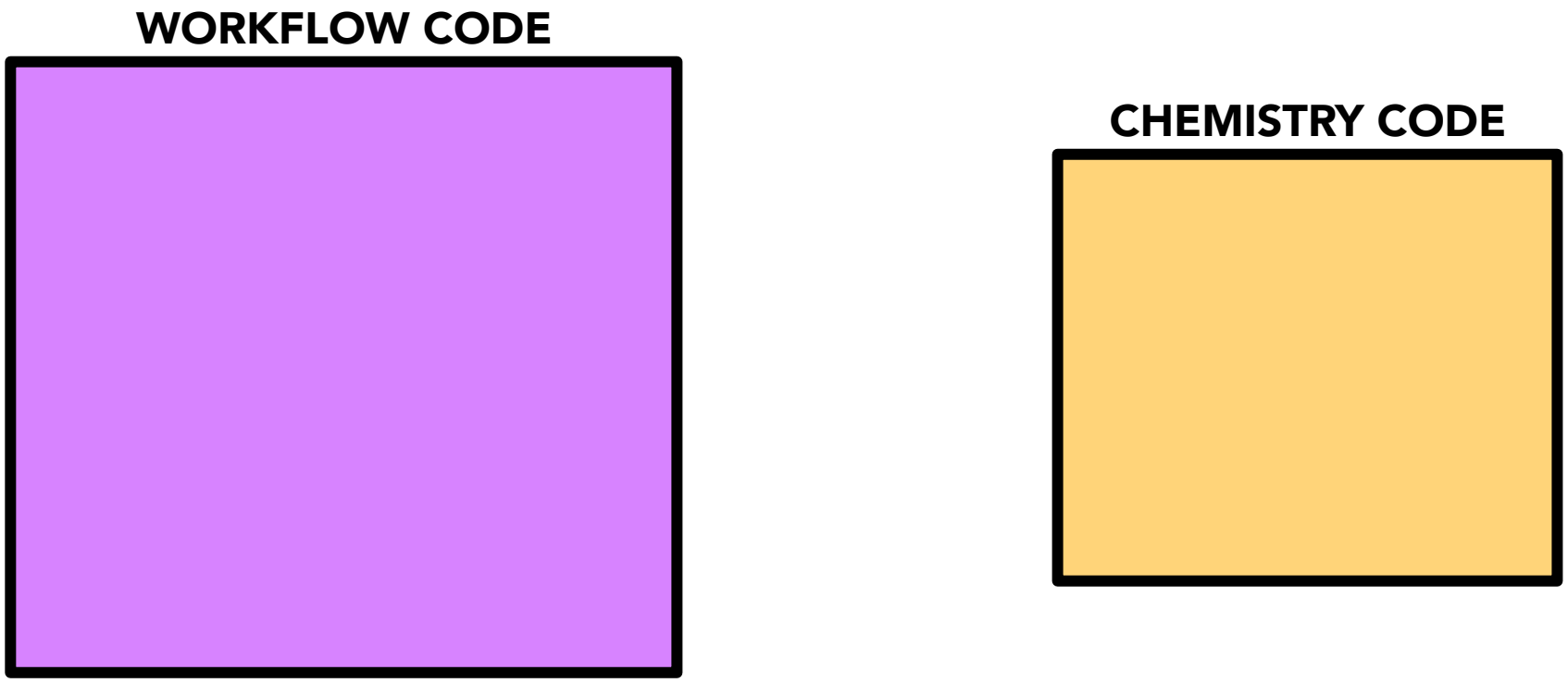


http://www3.canisius.edu/~grandem/animalshabitats/animals.jpg

Our workflow computing "habitat"

- Failures are common; search status of every job over the course of years
  - like UPS packages, database is a necessity
- Very dynamic workflows
  - results of a calculation greatly affect workflow, e.g. "self-healing" detours or "metal vs. insulator" flows
- Intelligently handle collisions/duplicates
  - people submitting the same material, perhaps with some calculations in common and some distinct
- Runs on a laptop or a supercomputer
- Can learn it by yourself without help/support

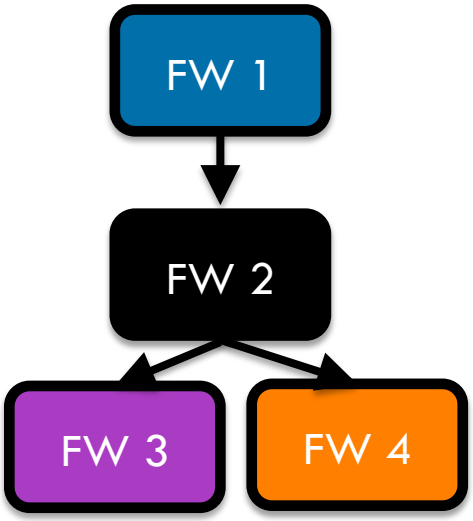
Hierarchical codebases:
FireWorks is our general workflow code



How FireWorks operates

ROCKET LAUNCHER / QUEUE LAUNCHER

Directory 1 | Directory 2



LAUNCHPAD

Workflows are simple JSON/YAML documents that have very little "fluff" (no ugly XML)

(this is YAML, a bit prettier for humans but less pretty for computers)

```
fws:
- fw_id: 1
  spec:
    _tasks:
    - _fw_name: ScriptTask:
      script: echo 'To be, or not to be,'
- fw_id: 2
  spec:
    _tasks:
    - _fw_name: ScriptTask
      script: echo 'that is the question:'
links:
  1:
  - 2
metadata: {}
```

The same JSON document will produce the same result on any computer (with the same Python functions).

JSON + MongoDB means you can store workflows directly and make rich queries

(this is YAML, a bit prettier for humans but less pretty for computers)

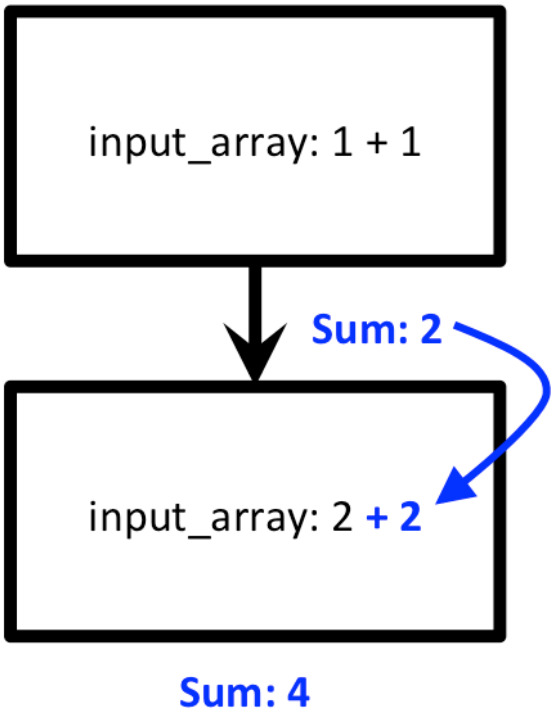
```yaml
fws:
- fw_id: 1
  spec:
    _tasks:
    - _fw_name: ScriptTask:
      script: echo 'To be, or not to be,’
- fw_id: 2
  spec:
    _tasks:
    - _fw_name: ScriptTask
      script: echo 'that is the question:’
links:
  1:
  - 2
metadata: {}
```

Just some of your search options:

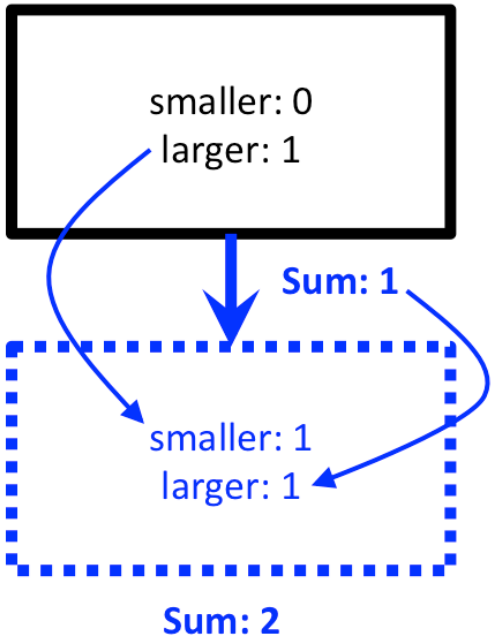
- simple matches
- match in array
- greater than/less than
- regular expressions
- match subdocument
- Javascript function
- MapReduce…

All for free, and all on the native workflow format!

Jobs can return objects that modify workflows or future jobs via JSON language



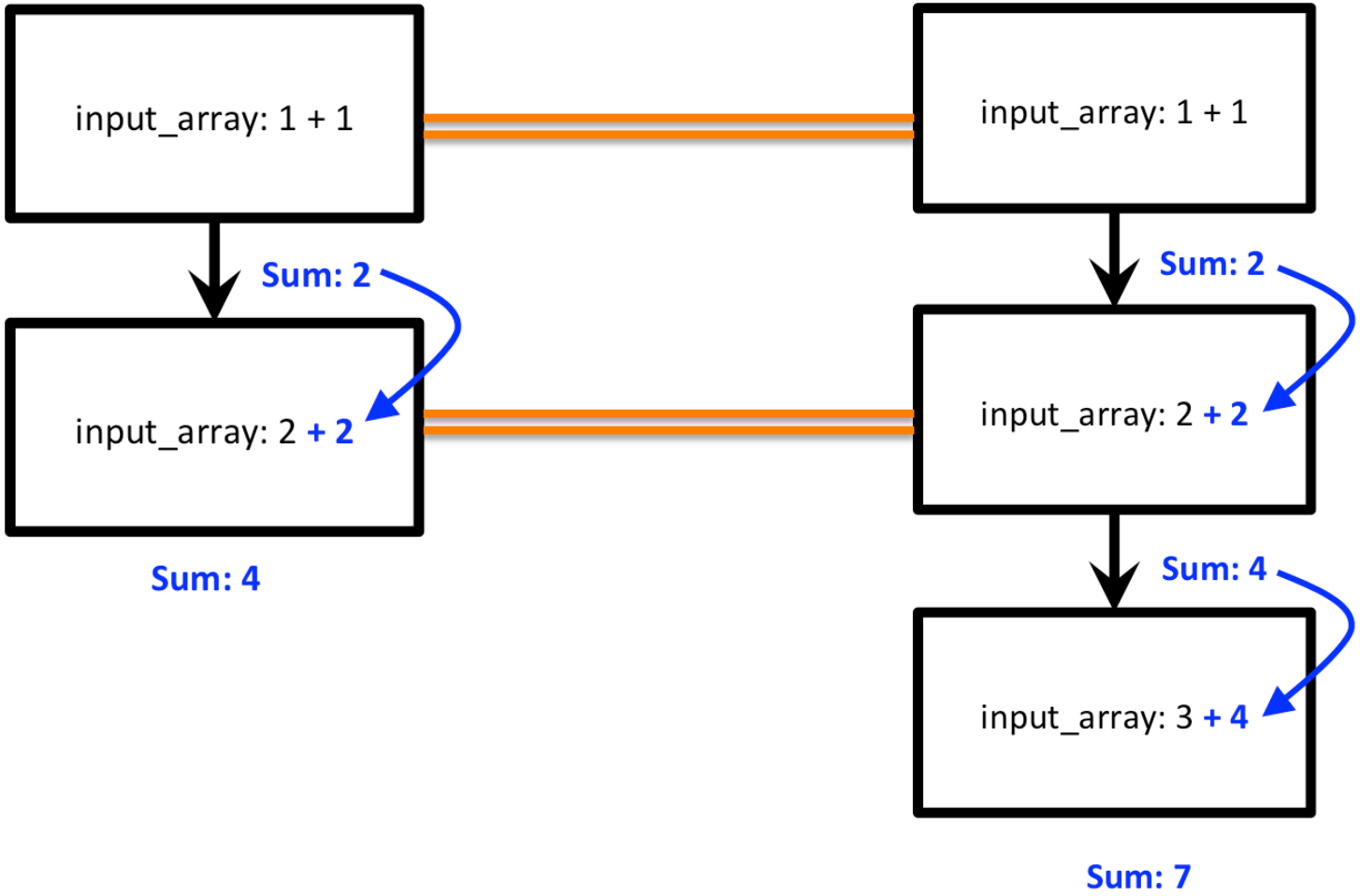
Use MongoDB's dictionary update language to allow for JSON document updates



Workflows can create new workflows or add to current workflow

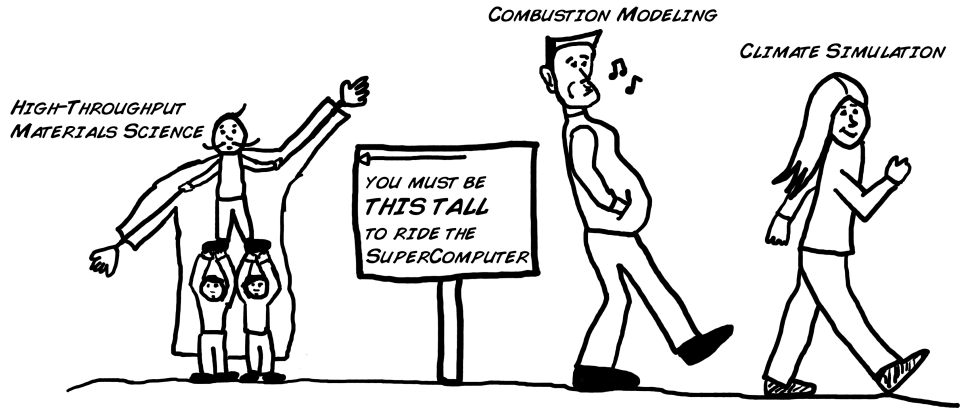
- a recursive workflow
- calculation "detours"
- branches

But wait – there's more!
JSON duplicate checking simple and automatic



Many execution modes

- Run directly on a node
- Run a single job within a PBS script
  - generic or highly tailored to the job
- Consecutively pull many jobs in a PBS script
- Run via PBS/SGE/SLURM/NEWT
- Distribute jobs over many workers
- Pack jobs and pretend to be a big job without any setup



Examples of successes

- Completed 575000+ PBS jobs worth of computations
- Gracefully recovered from many failures
  - very easy to track down which jobs failed, resubmit them with new code as needed
- Random people can submit the materials they care about and not worry about duplication
- Random people have downloaded and set it up without any outside help

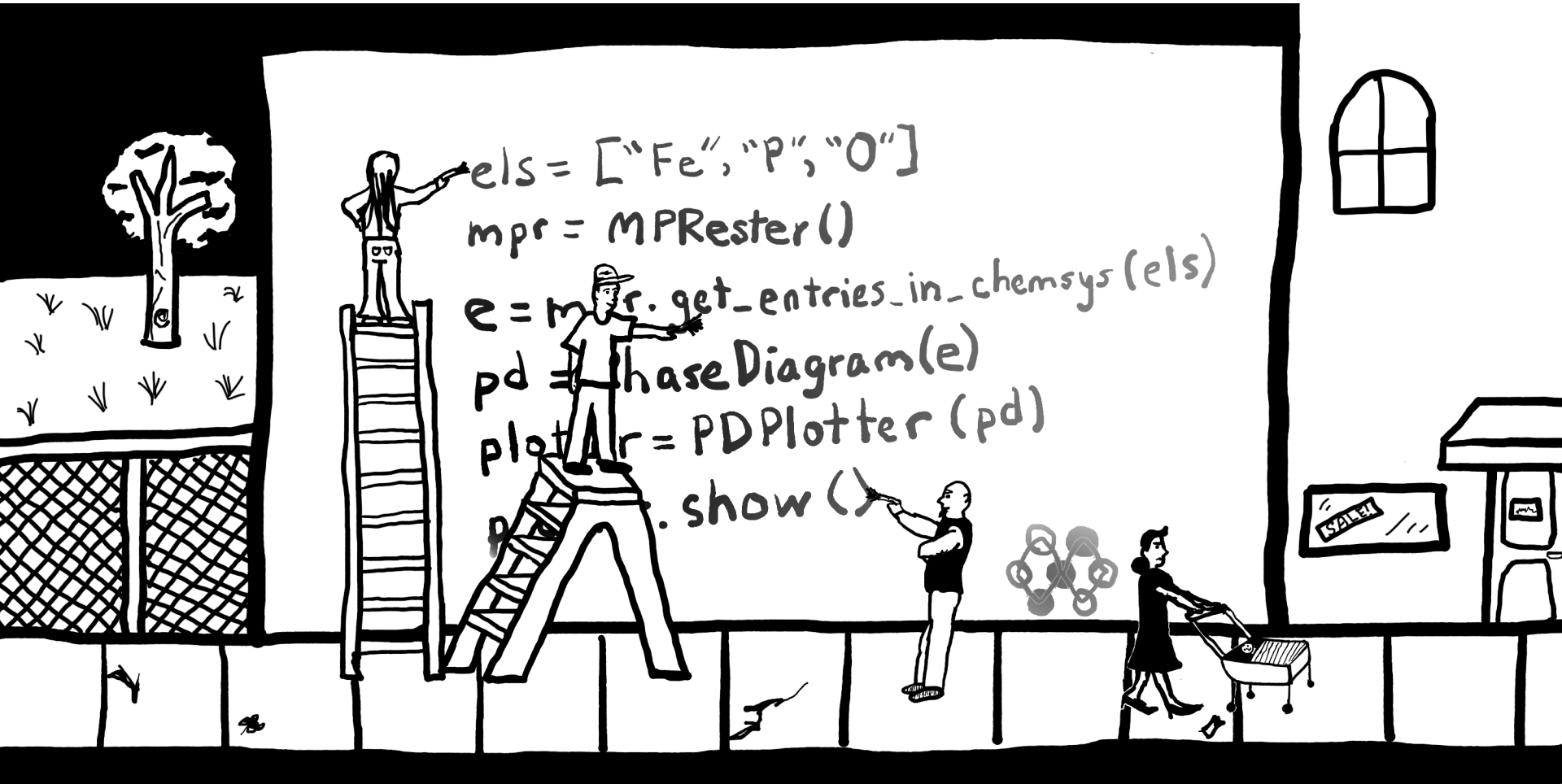
FireWorks is free and open-source

http://pythonhosted.org/FireWorks/

or Google “Python FireWorks”

You may not care about FireWorks, but you might consider open-sourcing your own code

- *Real* open source on a web site like Github
  - not “email me for code”



Things that have and haven't happened when going open source … overall very positive

HAPPENED

- I was automatically wrote better code and documentation
- Tricky but important bugs identified/fixed by community
- New bugs introduced by newcomers (but quickly fixed)
- Python 3 compatible by volunteer
- Internals became cleaner & user-friendly
- Heated arguments that resulted in improvements
- Learned about management
- Lots of good feature suggestions, some feature implementation by community
- Pace of development greatly accelerated
- Friendly users I had no relation to gradually came out of the woodwork and asked questions

DID NOT HAPPEN

- Code went viral
  - the world mostly did not notice…
- Thieves stole the code and didn't attribute it
  - I think…
- People blamed me for publishing imperfect code

FW Teammates
Shyue Ping Ong
Xiaohui Qu
Morgan Hargrove
David Waroquiers
Dan Gunter
Wei Chen
Kristin Persson (very patient funder)