JOINT GENOME INSTITUTE

A DOE OFFICE OF SCIENCE USER FACILITY

Reproducibility at JGI as an organizational challenge

Edward Kirton
JAWS technical lead
Advanced Analysis Group

FAIR Data Standards

JGI-generated data must be FAIR, as mandated by our funding agency (US Department of Energy)

- **F**indable : searchable, rich metadata
- **A**ccessible : readily available on the internet
- **I**nteroperable : standard file formats, vocabularies/keywords
- **R**eusable : clear usage license, provenance, follow accepted standards
- **R***eproducible : published software and documented parameters*

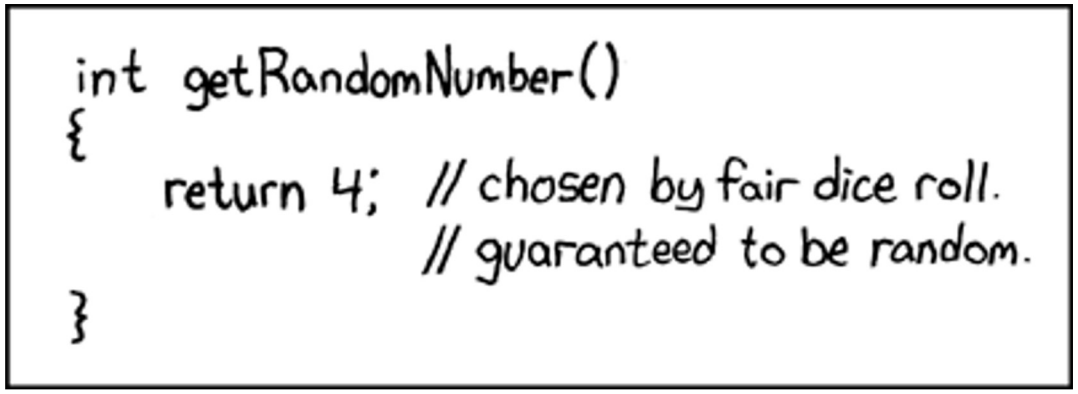
- **Challenge**: Without published workflows, it may be necessary to reprocess raw data in order for results to be combined/compared.
  - with large amounts of data this may be expensive to unfeasible
  - tools/methods also improve over time, requiring old data to be reprocessed

Reproducibility in bioinformatics

JGI JOINT GENOME INSTITUTE

What is “acceptable reproducibility” in bioinformatics and how do you test it?

- Easy case: `diff`; may need to specify random seed (if possible)
- Commonly have small- and medium-sized test sets we know well
- Often: Near-identical results may be acceptable or even expected
  - Comparing results between versions of a tool requires expertise
  - Producing mock data with the same (ever-changing) error profiles as real data is nontrivial; we prefer real data
- When the data is reanalyzed, one should come to the same conclusions (because who can say if the minor differences are real or noise?)
  - results often vetted by subsequent analysis scripts or pipelines

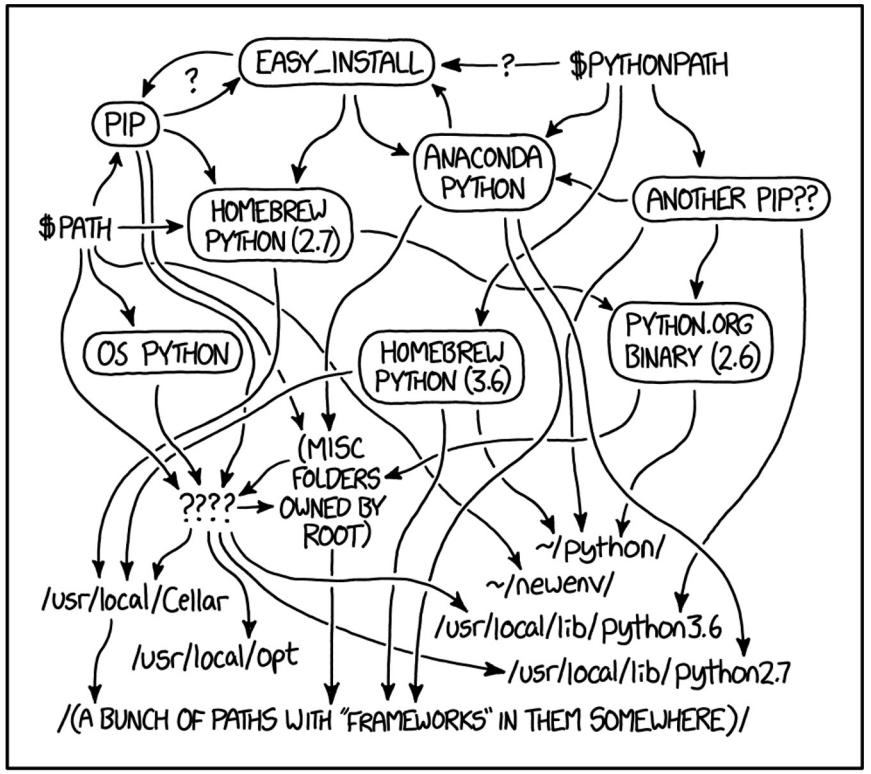


https://xkcd.com/

What limits reproducibility?

Specifically, when someone *else* wants to reproduce your work.

- Within the research community, sharing software pipelines is a real problem
  - Other labs may have different kinds of computational resources and not necessarily a lot of CS expertise
  - Sharing homegrown tools between groups in-house is sometimes also difficult!
- Pointing a colleague to a git repo isn't always sufficient
- Enter workflow frameworks to promote sharing, reuse, and collaboration



MY PYTHON ENVIRONMENT HAS BECOME SO DEGRADED THAT MY LAPTOP HAS BEEN DECLARED A SUPERFUND SITE.

https://xkcd.com/

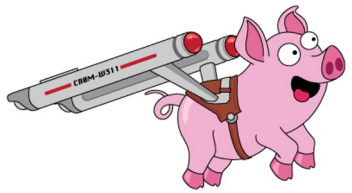
Challenge: Reproducibility shouldn't be difficult

This is an organizational challenge that requires a culture shift.

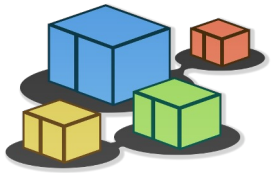
- Sharing software is sometimes difficult
- Investigators are primarily concerned with conducting and publishing their research; must make packaging and releasing software easy
- Containers (e.g. Docker) are tremendously useful
- Workflow frameworks' allow analysis to focus on the data, not glue-code
  - WDL, CWL, SnakeMake, more.
  - *Requires workflow infrastructure, training, and support*

JAWS
JGI Analysis Workflow Service

{wdl}



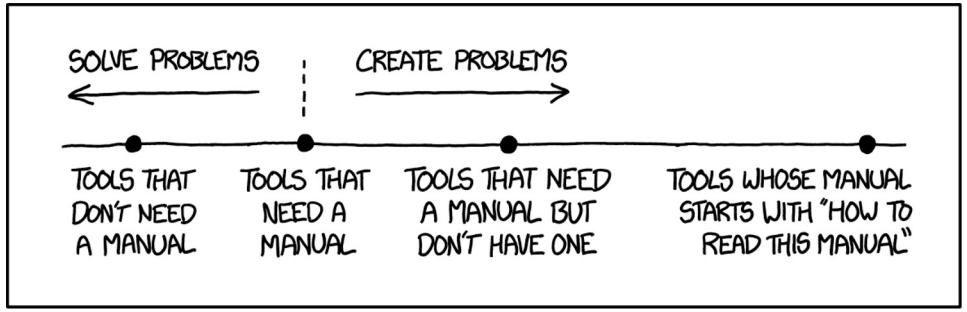
docker hub



Lessons learned

What we learned after promoting WDL+Docker at JGI for two years

- **Getting people to adopt new behaviors isn't easy, even if beneficial in the long run**
  - "I'm too busy to learn something new"
  - "What I'm doing now works fine"
  - "I don't want to rely on others"
- **Must have framework people can depend on**
  - Limit features and focus on core functionality ***(Keep It Simple, Stupid!)***
- **On-board a small number of people first.**
  - Most receptive and capable, not necessarily most in need.
- **Build a dedicated user community through mutual support**
  - let the best users drive feature development
  - spend considerable time on community-building activities (workshops, pair-programming, open office-hours, building shared sub-workflow libraries)
- **Grow**
  - add Spark integration next?



https://xkcd.com/