Challenge of Reproducible Pipelines

Pjotr Prins

11th Biohackathon 2018
Matsue, Japan, December 9th,

UMC Utrecht/UTHSC GeneNetwork.org
Challenge

Reproducible analysis starts with software
Software deployment is boring
Avoid

Programmers prefer to look away
Reproducibility

What about Docker?

- Docker is a binary blob
- Also creating Docker images is not reproducible
- Nor are Debian, Conda, Brew etc. reproducible
- It is all about fixating dependencies (and bootstrapping)
- Building on shifting sands
GNU Guix

- Guix gives reproducible software installation
- Guix is easy
- Guix has versioning
- Guix give real control over the full dependency graph
- it just works (tm)
- Guix creates reproducible binaries with dependencies AND even Docker containers
Confession

I love GNU Guix
Goal

Write a pipeline using CWL and Guix and document it
Goal

Write a pipeline using CWL and Guix and document it

- CWL reference runner
- Software graph is reproducible (from source)
- Data is content-addressable
- Metadata: software and data origins/descriptions (wikidata)
- See if we can embed it in Shogun - block chain
ENV

Never go it alone

- CWL (Michael Crusoe a.o.)
- Galaxy (Conda support, CWL support, RStudio, Jupyter Labs....
- GeneNetwork.org
- Wikidata
- Blockchain scientific credit (Alexander Garcia Castro a.o.)
Wouldn’t it be amazing to have fully reproducible and shareable pipelines

- It can be done. We have the technology
- And I have found software deployment is not boring
- Full control over the software dependency graph means things get fixated in time - you can move forward