Docker Birmingham November 2019

Make Data Science Great Again

docker

https://github.com/docker-birmingham/policies

Coffee Ops

Come and talk about dev and ops and people.

Tues 0730 urban coffee church st.

DEVOPSDAYS

Birmingham 2021

calendar.birmingham.io

Next Meetup - 4th Dec 2019

Thanks to our Sponsor!

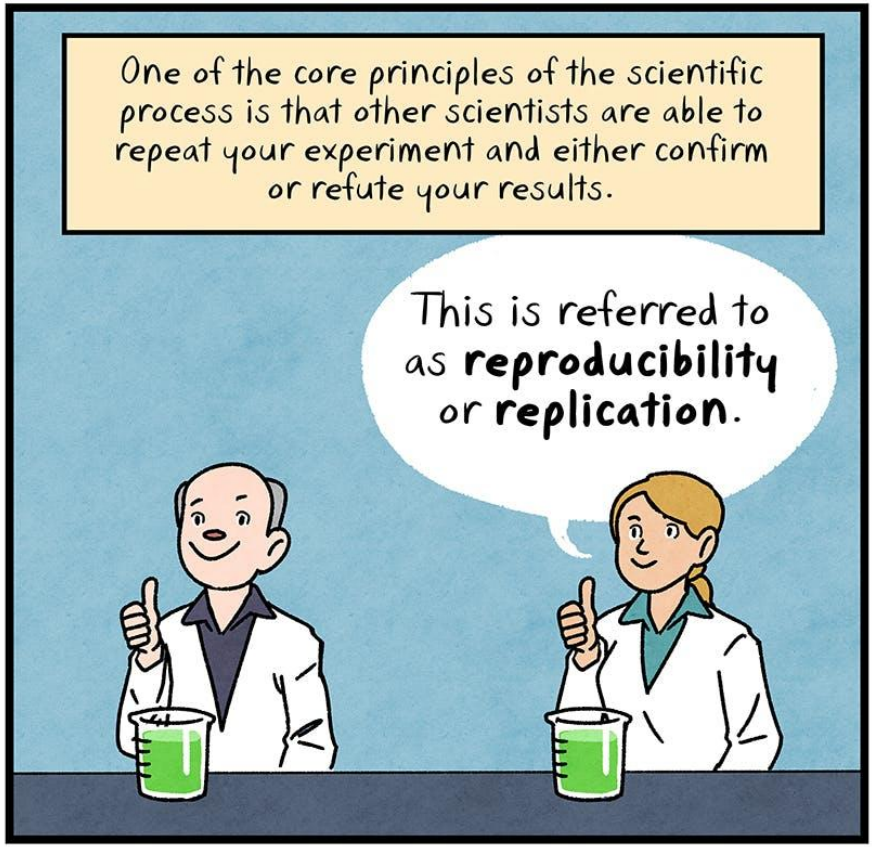
BlackCat /

Science!

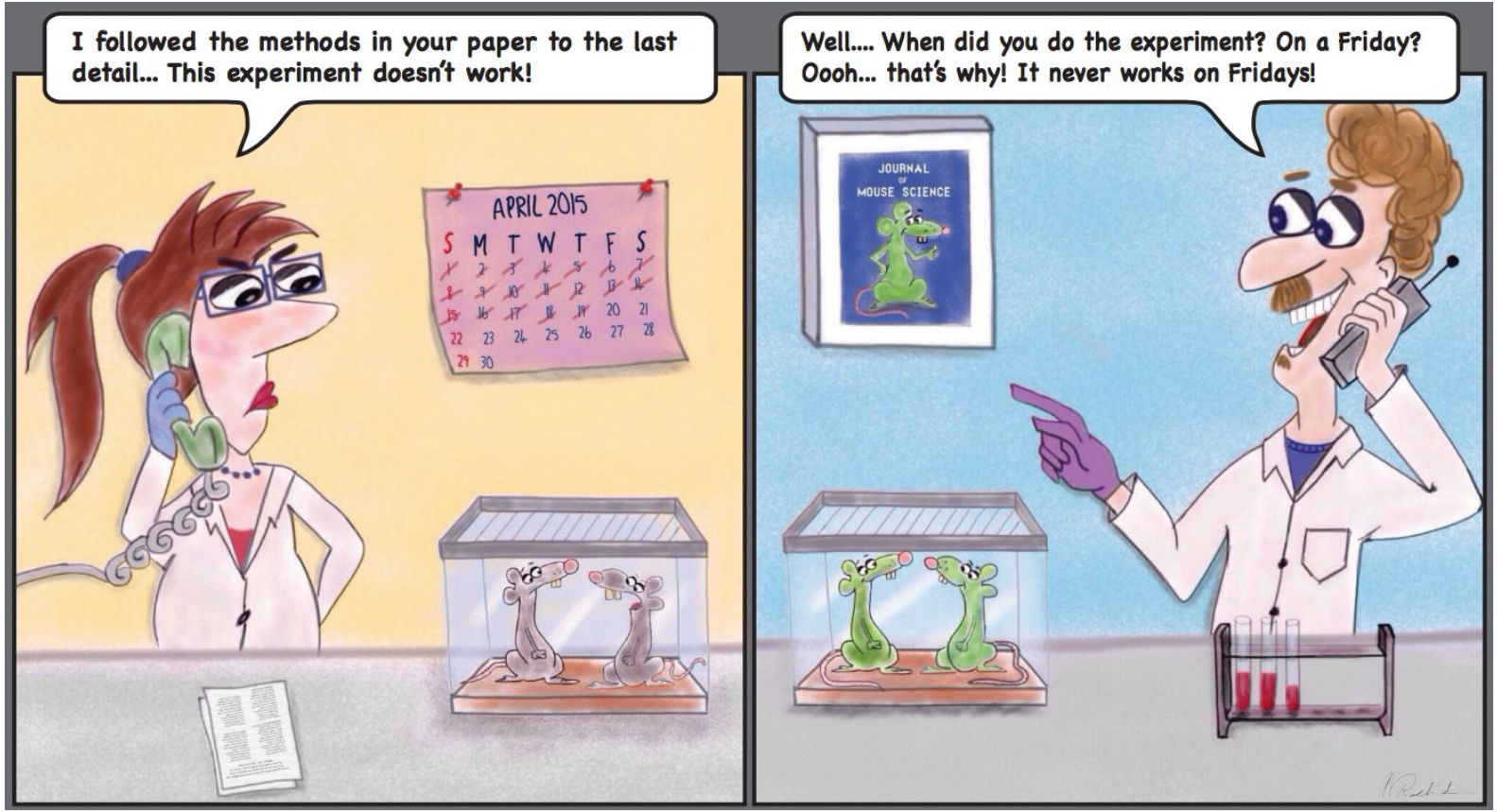
(Part 2!)

The Scientific Method

1. Define a question
2. Gather information and resources (observe)
3. Form an explanatory hypothesis
4. Test the hypothesis by performing an experiment and collecting data in a reproducible manner
5. Analyze the data
6. Interpret the data and draw conclusions that serve as a starting point for new hypothesis
7. Publish results
8. Retest (frequently done by other scientists)



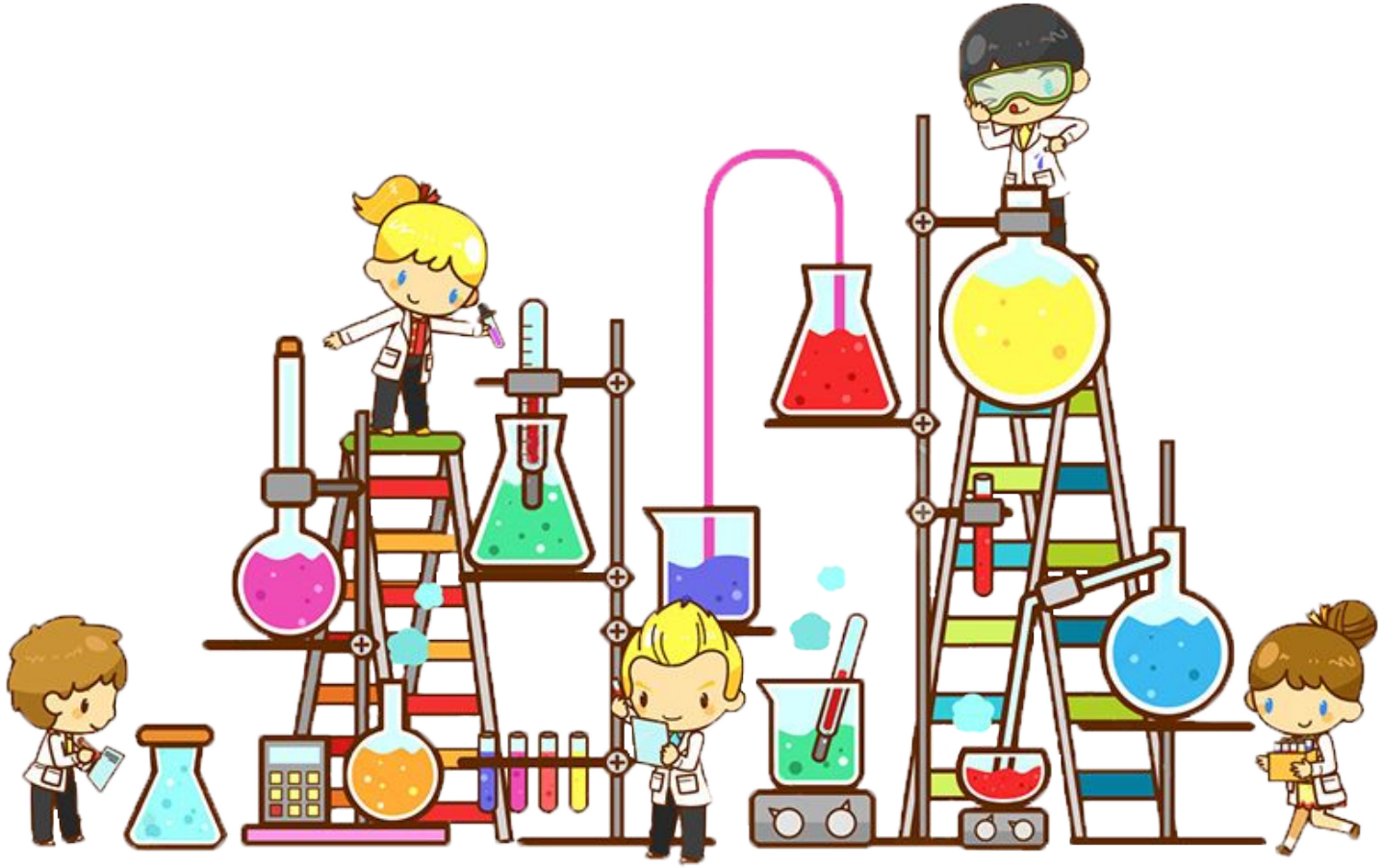
Reproducibility / Replication



Issues with Reproducibility

Software Lifecycle TDD

Designing ***Repeatable*** and ***Reproducible*** Experiments is ***hard.***



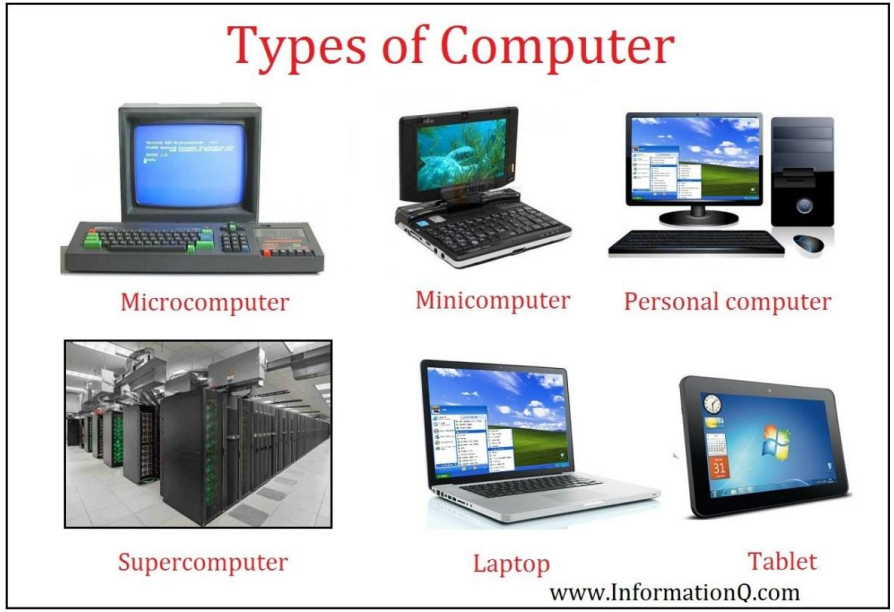
Physical Science Lab

Computer Version

Why is this hard?

Versions, versions, versions…….

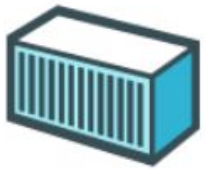
- Type of Machine
- Version of Programming Language
- Version and state of OS
- Version of Libraries / Dependencies
- Data location & drift



Docker Science Lab

Build

Develop an app using Docker containers with any language and any toolchain.



Ship

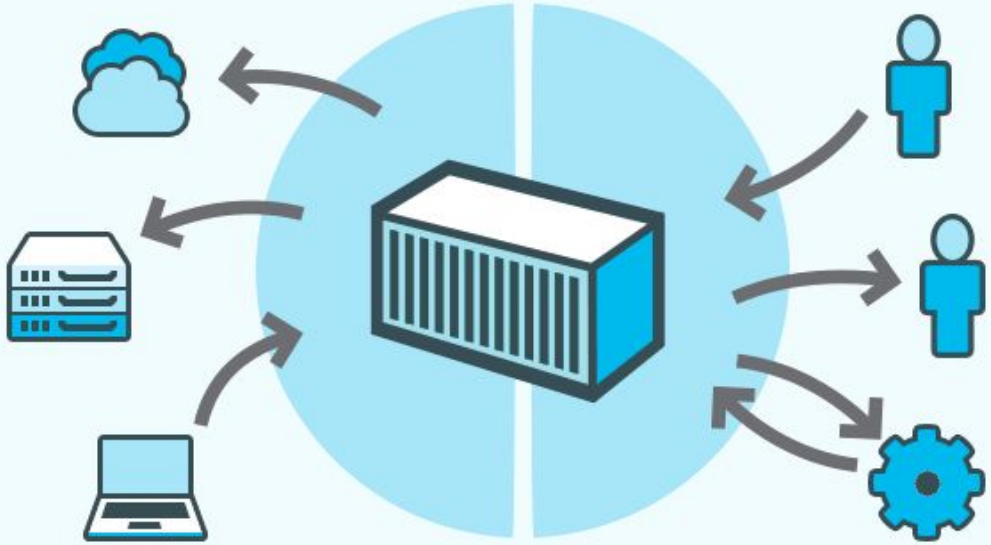
Ship the “Dockerized” app and dependencies anywhere - to QA, teammates, or the cloud - without breaking anything.

Run

Scale to 1000s of nodes, move between data centers and clouds, update with zero downtime and more.

What Is Docker?

An open platform for distributed applications



Docker Engine

A portable, lightweight application runtime and packaging tool.

Learn More

Docker Hub

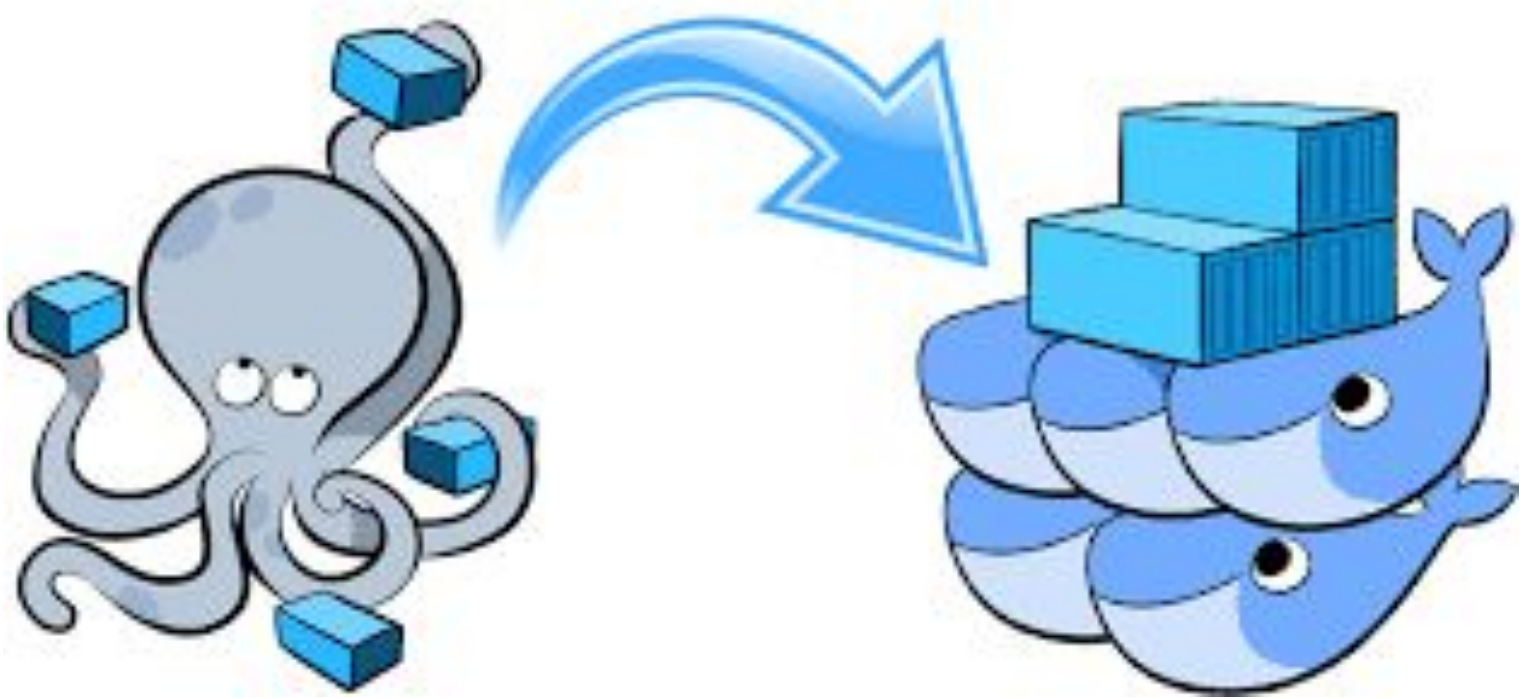
A cloud service for sharing applications and automating workflows.

Building a Digital Scientists Lab

”Home Lab is the Dopest Lab”

Jessie Frazelle
https://blog.jessfraz.com/post/home-lab-is-the-dopest-lab/

Service Orchestration

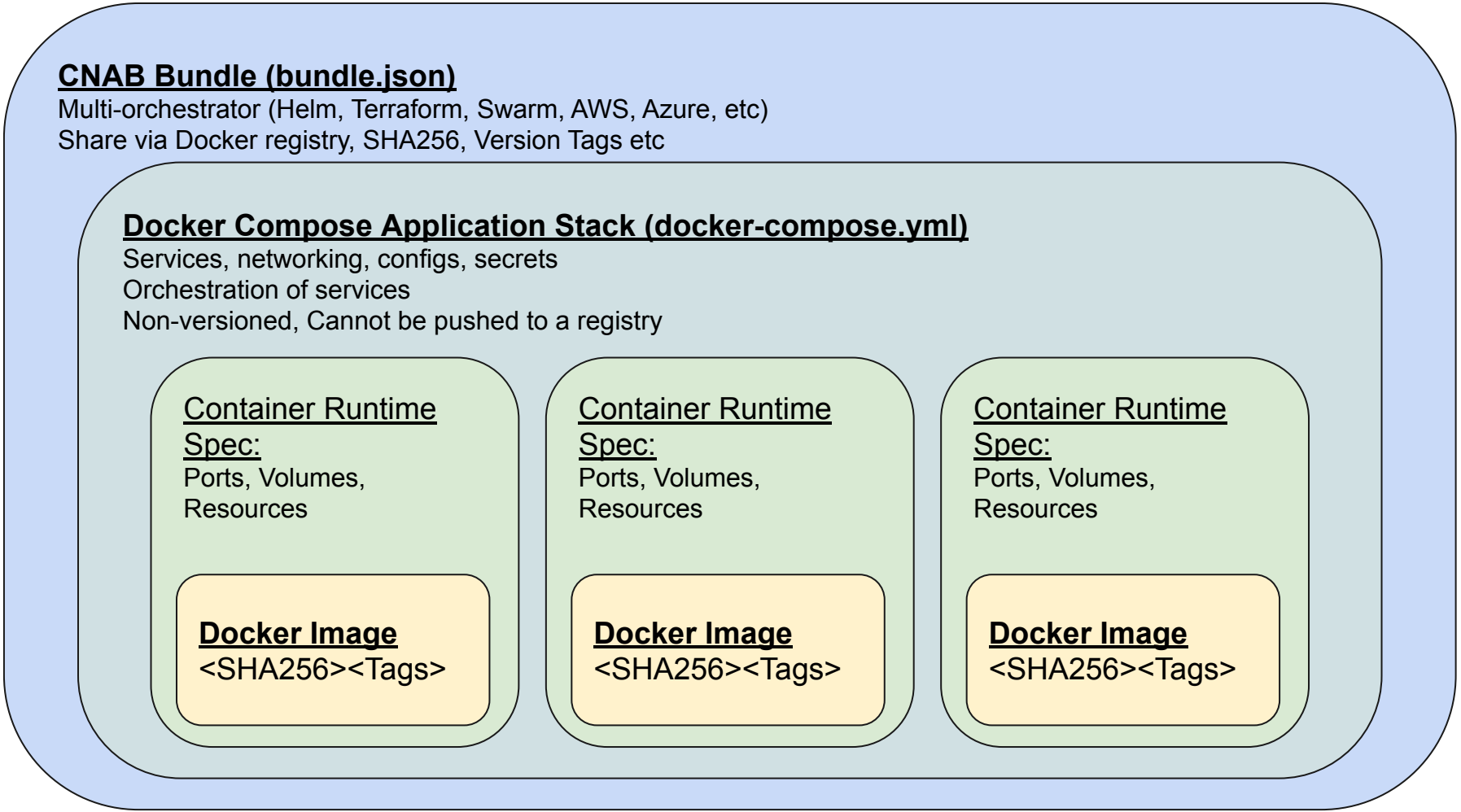


Cloud Native Application Bundle

A spec for packaging distributed apps.

“CNABs facilitate the bundling, installing and managing of container-native apps — and their coupled services.”

““As modern applications continue to grow in complexity, there’s an immediate need to simplify how these multiservice, distributed applications are built, shared and run”


CNAB Bundle (bundle.json)
Multi-orchestrator (Helm, Terraform, Swarm, AWS, Azure, etc)
Share via Docker registry, SHA256, Version Tags etc
Docker Compose Application Stack (docker-compose.yml)
Services, networking, configs, secrets
Orchestration of services
Non-versioned, Cannot be pushed to a registry
Container Runtime Spec:
Ports, Volumes, Resources
Docker Image
<SHA256><Tags>
Container Runtime Spec:
Ports, Volumes, Resources
Docker Image
<SHA256><Tags>
Container Runtime Spec:
Ports, Volumes, Resources
Docker Image
<SHA256><Tags>

Define **docker-compose** stack

Initialize Docker Application definition

Install application into “cluster”, go science!

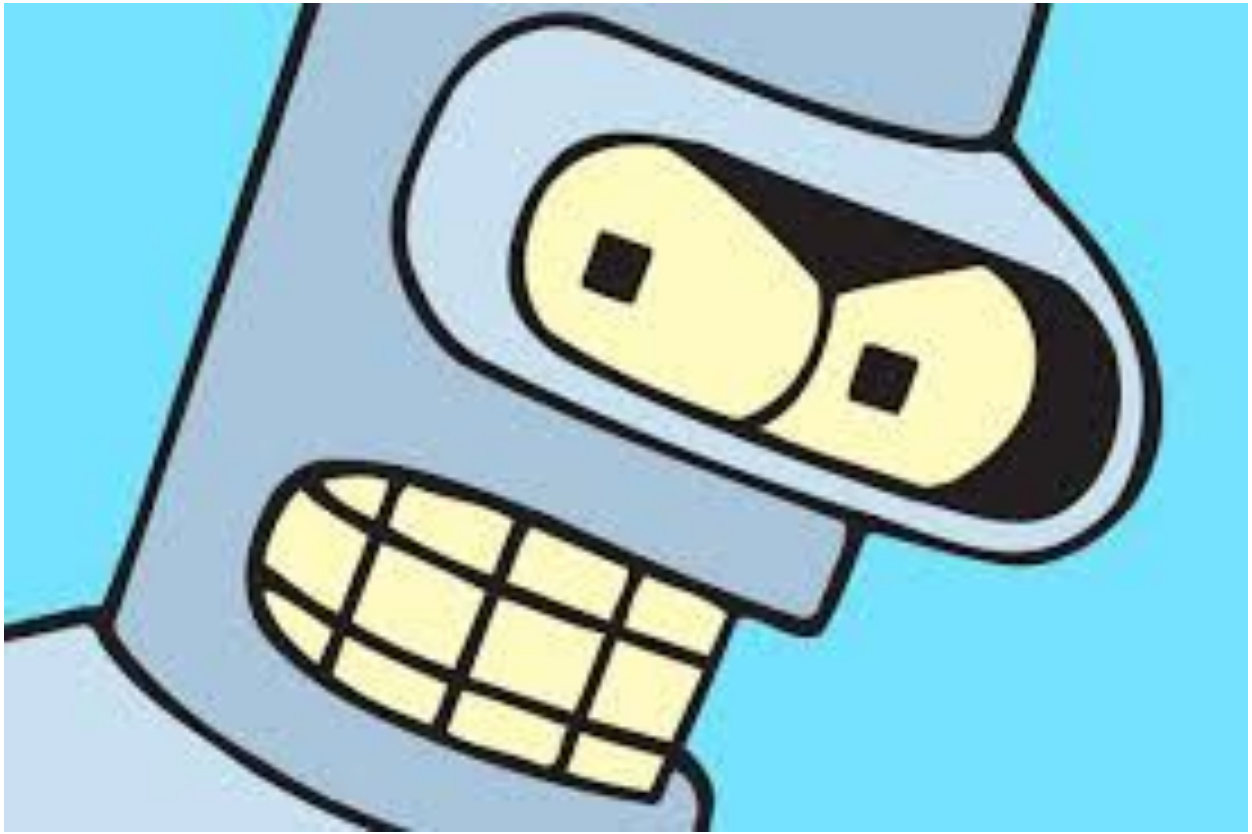
Push an application package to a shared registry (Docker Hub)

Pull application package from a shared registry

Inspect metadata, parameters and a summary, ports etc

Install application, go science!

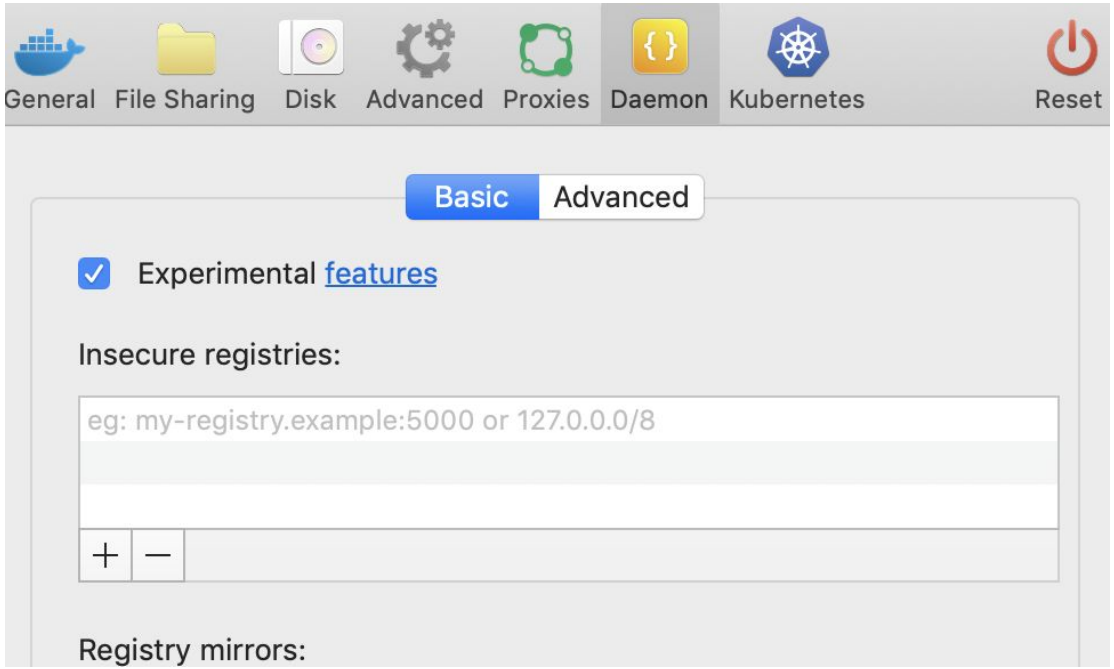
Upgrade, update, remove, collaborate



Warning, these stacks are not secure or HA by default!

Enable Docker app (CNAB)

- Via Docker Mac App
- **config.json**



https://github.com/docker-birmingham/data-pipeline


nifi


nifi


APACHE
Spark


jupyter


Flink


RabbitMQ


kafka


elastic stack