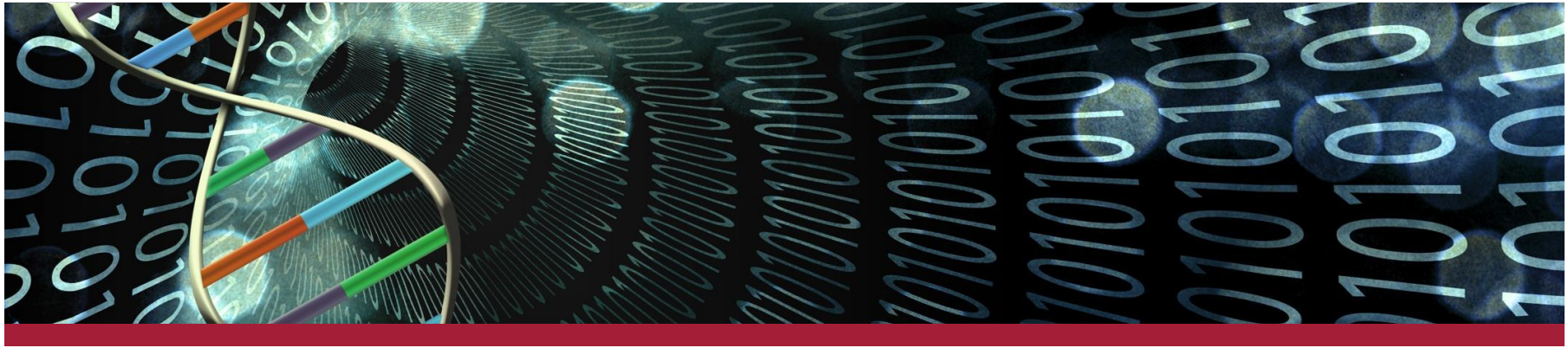
EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Quantitative Biology Center
(QBiC)

QBiC

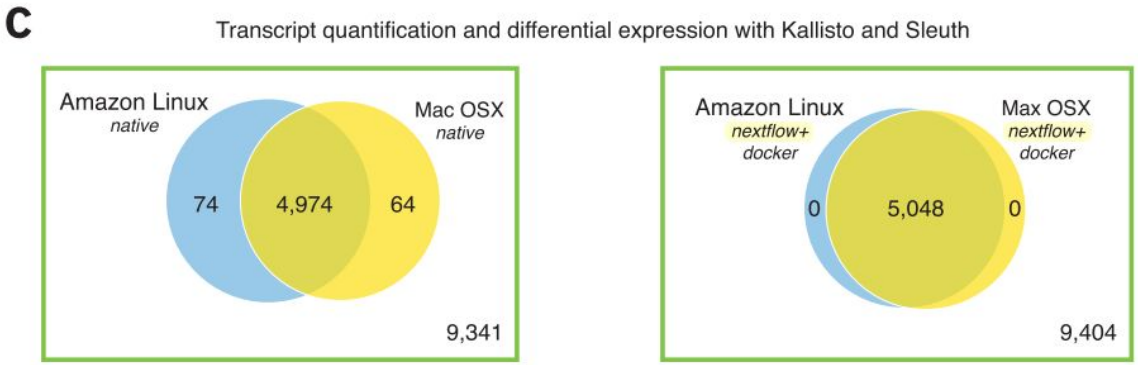


Computational workflow seminar: Understanding Docker, Singularity and containerization

Workflows and reproducibility

Number of differentially expressed genes without Nextflow and containerization

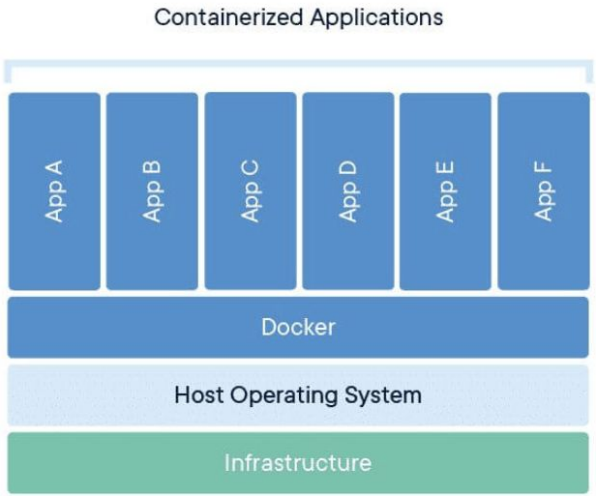
Number of differentially expressed genes with containerization



DI Tommaso, et al. (2017). Nextflow enables reproducible computational workflows. *Nature Biotechnology*.

Containers

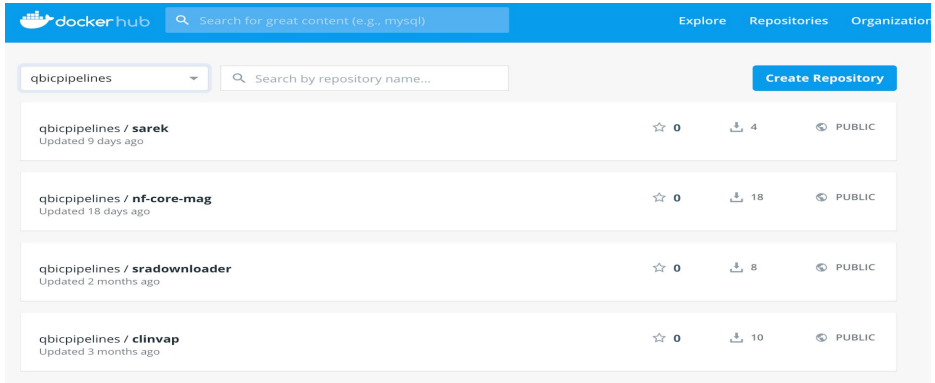
- Isolation of a runtime environment containing all tools and their dependencies and libraries with fixed versions
- Portable to any platform that supports a container engine
- **Container image**: portable snapshot of all dependencies, used to create a **container**
- Compared to Virtual Machines (VMs)
  - No guest OS, use the host's kernel
  - More lightweight, faster
  - Easier to ship



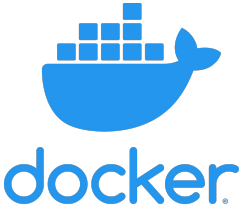
Container engines

Docker

- Since 2013, huge community
- Public repository for container sharing (DockerHub)
- Uses Dockerfiles
- Install Docker and you can run any docker container you want, without modifying it



Container engines



Docker

- Dockerfile, a recipe for building a container
- Example: Official Linux CentOS image:

```
FROM scratch
ADD centos-7-docker.tar.xz /
LABEL name="CentOS Base Image" \
      vendor="CentOS" \
      license="GPLv2" \
      build-date="20210630"


CMD ["/bin/bash"]
```

Container engines

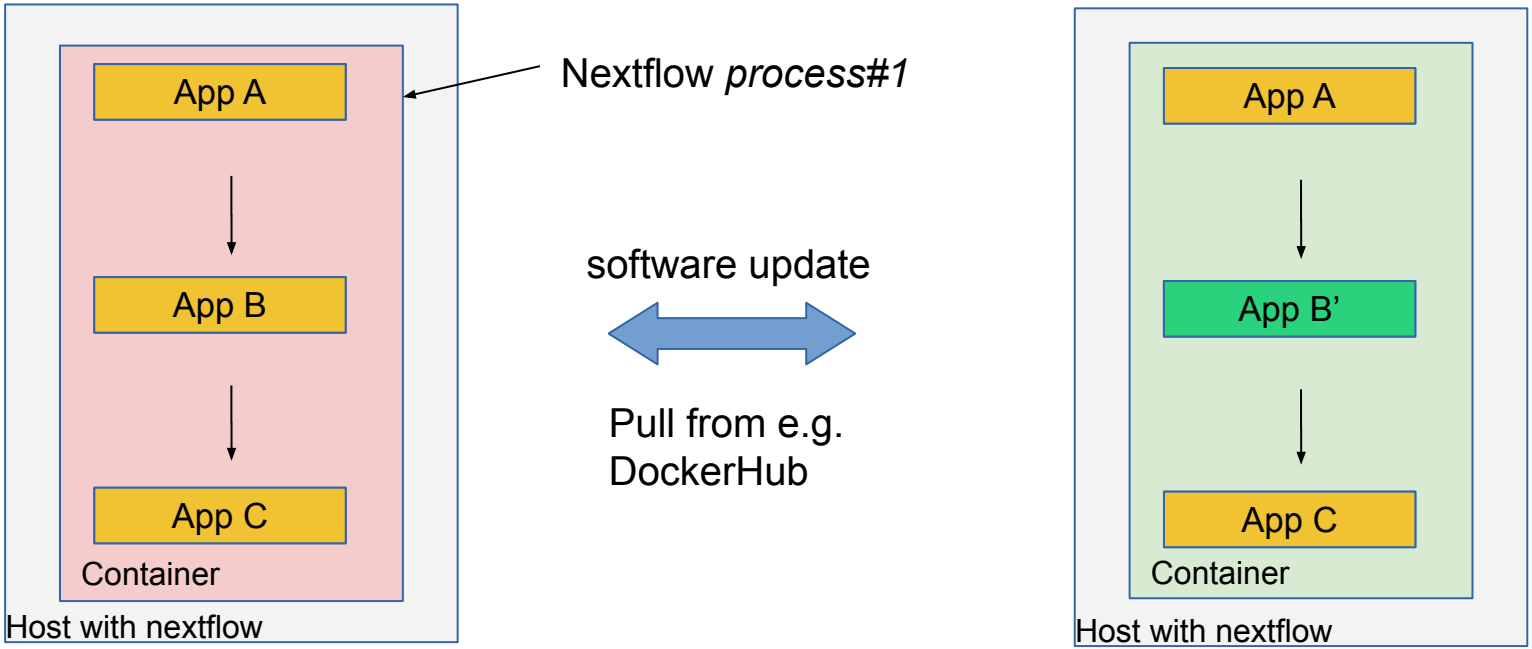
Docker - Cons

- Requires a daemon (and a privileged user to set it up)
- Security risk because of the direct access to the host OS kernel
- Docker devs addressed many issues in the past, but misconfiguration can still present a surface for various attack types (Linux capabilities, Network, Kernel Security Systems)
- Alternatives: Singularity, Apptainer, Podman, Charliecloud, Shifter …


Bui, Thanh. “Analysis of Docker Security.” arXiv:1501.02967 [Cs], January 13, 2015. http://arxiv.org/abs/1501.02967.

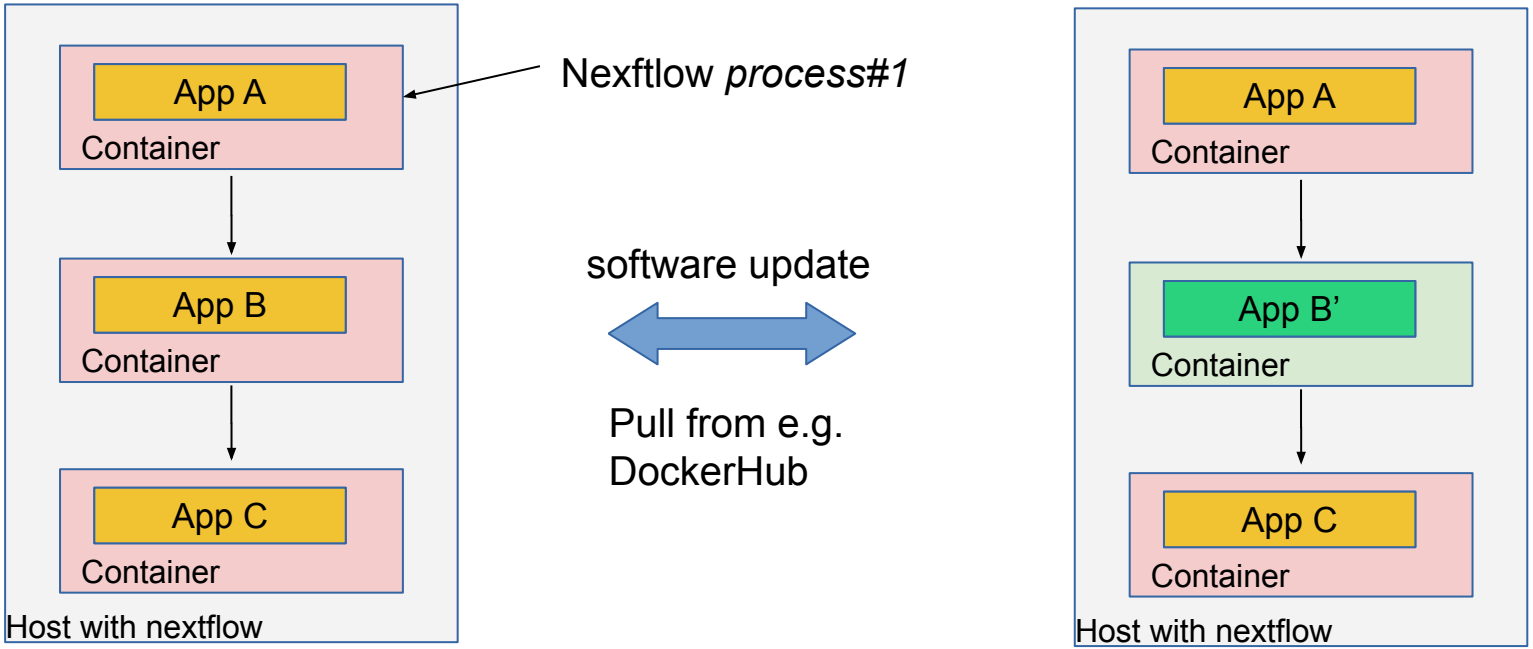
Container engines

Containers and Workflows – Fat container approach



Container engines

Containers and Workflows – Slim container approach



BioContainers project

For each tool (and release) available at BioConda, a minimal container (docker and singularity) is automatically built

BIOCONDA

VM vs Containers

Virtual Machines (VMs)

- Complete OS with its own kernel.
- Resource-heavy, slower startup, larger size.

Containers

- Shared OS kernel with isolated user spaces.
- Lightweight, fast startup, smaller size.

Getting your Material for today

Step 1: Save your work to **your fork**

```
$ git add <file1> <file2>


$ git commit -m "adds files"


$ git push
```

Step 2: Get today's notebook **from upstream**

```
# pull from upstream

$ git pull upstream main

# To leave vi:
# ":" -> "q" -> "w" -> "enter"
```