B3Africa Webinar 1
IT Focal Points

29 September, 14:00-15:30 CEST (UTC+2/ Paris)

B3 Africa
BRIDGING BIOBANKING AND BIOMEDICAL RESEARCH
ACROSS EUROPE AND AFRICA

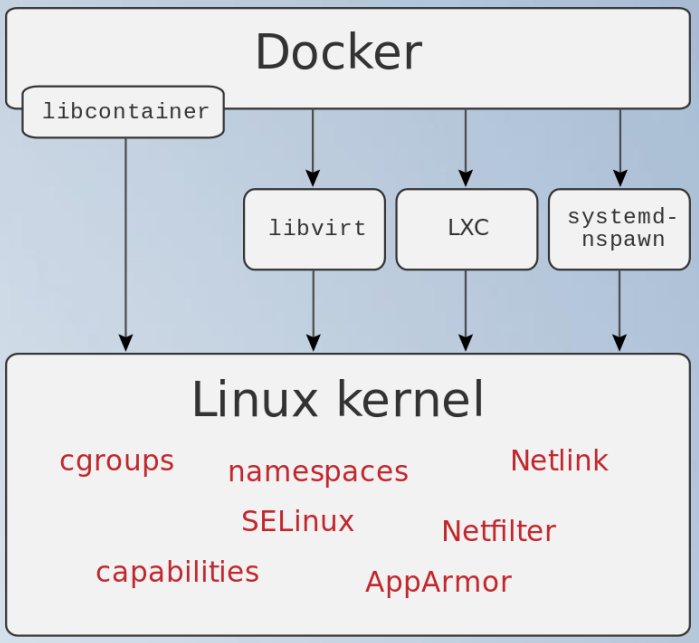
Overview

TOPIC 2: How to manage BIBBOX/eB3kit Dockers and Containers

- What is Docker
- Difference to VMs
- Workflow
- Run
- Dockerfile
- Docker Compose
- Logging
- Backup

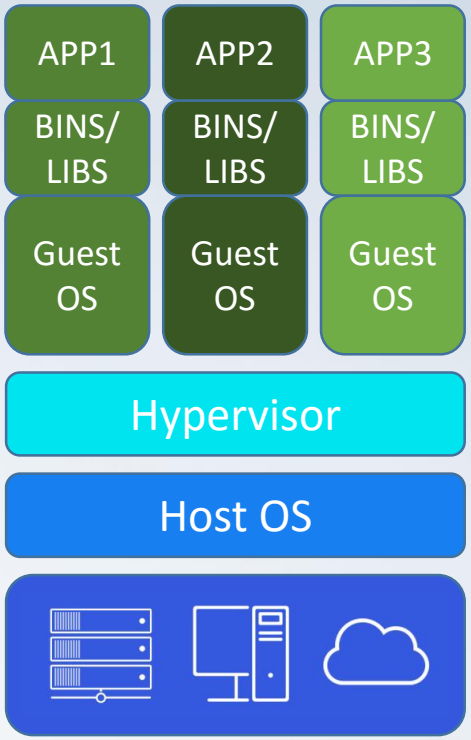
What is Docker

- Solomon Hykes started Docker 2013
- Open-Source
- Implements a high-level API to provide lightweight containers that run processes in isolation.
- Operation System Level Virtualization based on Linux
- Uses cgroups and kernel namspaces for isolation
- Linux virtualization features: LXC Linux container, libcontainer, libvirt and systemd-nspawn

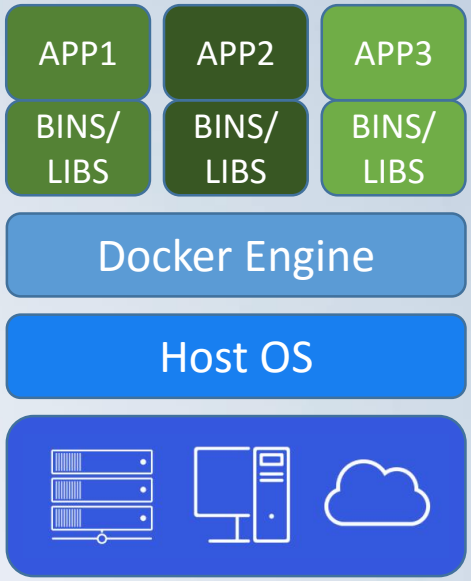


Top Contributors: The Docker team, Red Hat, IBM, Google, Cisco Systems and Amadeus IT Group

What is the difference to VMs



Includes application, binaries, libraries and guest operating system



Includes application, binaries, libraries Share the kernel

Container have similar resource isolation and allocation benefits as virtual machines but a different architectural approach allows them to be much more portable and efficient.

Docker Workflow Components

Docker Image

- Holds the environment and the application

Docker Container

- Created from the image (start, stop, move, delete)

Docker Registry

- Public and private repository used to store images

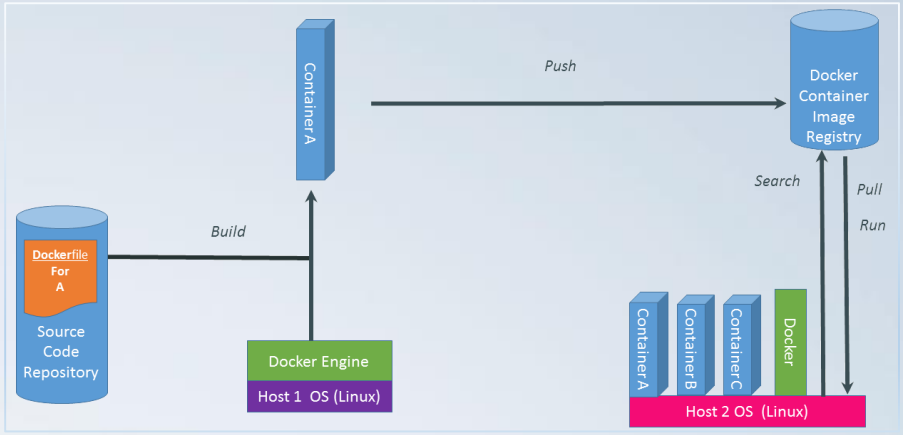
Dockerfile

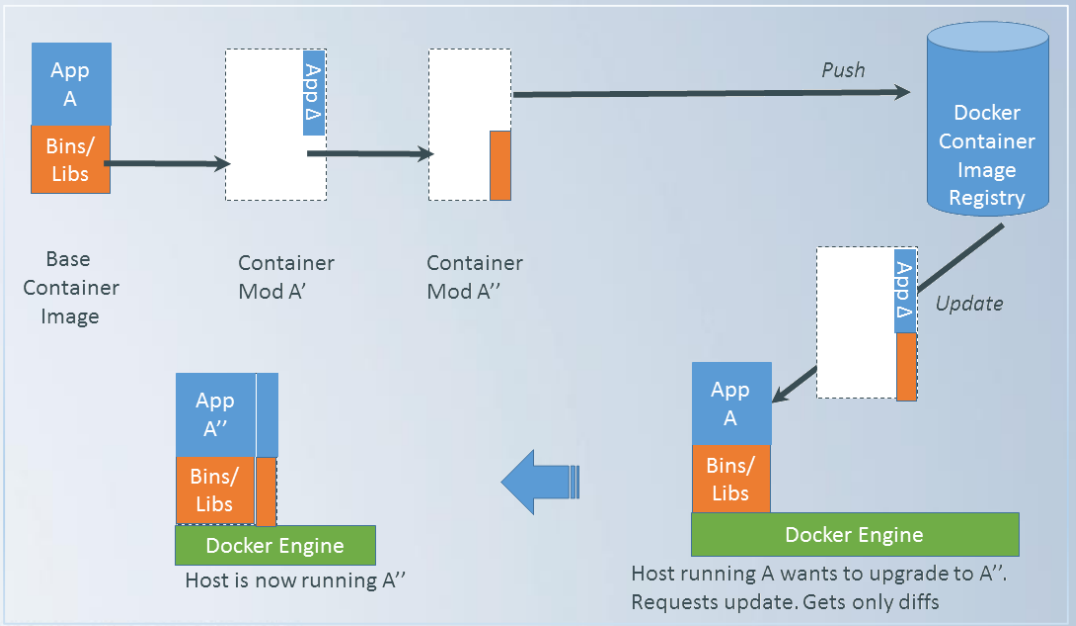
- Automate image construction

Docker Compose

- Defining and running multi-container applications

Run and Update a Docker Container





B3 Africa
BRIDGING BIOBANKING AND BIOMEDICAL RESEARCH
ACROSS EUROPE AND AFRICA

Run Docker (demo)

docker run [OPTIONS] IMAGE [COMMAND] [ARG...]

docker run – execute a container

[OPTIONS] – options for executing the container

--name – name of the container

-d – run container in backgound

-t – allocate a pseudo-TTY

-v – mount a volume

Java:7 – specify the image for the container

Java – execute the command

-jar /opt/test.jar – arguments for the command

Run Docker (demo)

- docker logs –f CONTAINER
- docker stop CONTAINER
- docker restart CONTAINER [COMMAND]
- docker images
- docker inspect CONTAINER
- docker exec
- docker ps
- docker pull
- docker search
- docker stats

Dockerfile (demo)

FROM
MAINTAINER
RUN
CMD
ENV
WORKDIR
ADD
COPY
EXPOSE
LABEL
VOLUME
STOPSIGNAL
ONBUILD

Dockerfile

```
FROM busybox
ENV foo /bar
WORKDIR ${foo}     # WORKDIR /bar
ADD . $foo         # ADD . /bar
COPY \$foo /quux   # COPY $foo /quux
```

Build image localy:

docker build -t bibbox/testapp .

Run Docker Compose

Compose is a tool to run and link multiple docker containers together.

Compose contains multiple steps

- Devine the app's with a Dockerfile
- Devine the images that define the app in a docker-compose.yml file
- Run the compose file with docker-compose up

Docker logging

- What messages are logged?
  - Everything that is written to the container standard output/error
- How do I get my loges shipped to the logging driver?
  - By default to the default logging driver. (**json-file**)
  - **none**: disables any logging for the container
  - **syslog**: writes logging information to syslog or ships it to a syslog server
  - **journald**: ships logging information to a journald (part of systemd)
  - **gelf**: ships logging information to a gelf endpoint (Graylog or Logstash)
  - **fluentd**: ships logging information to fluentd
  - **awslogs**: is the Amazon CloudWatch Logs for docker
  - **splunk**: ships logging information to splunk using HTTP event collector
  - **etwlogs**: writes loggong information as ETW events (Windows)
  - **gcplogs**: ships logging information to Google Cloud Logging

Docker logging

json-file

- docker run --name demo1 -dt -v "$PWD/test.jar":/opt/test.jar java:7 java -jar /opt/test.jar
- docker logs demo1

  Java Programm Running

  Java ERROR...
- sudo tail -f /var/lib/docker/containers/8c..5f/8c..5f-json.log

  {"log":"Java Programm Running\r\n","stream":"stdout","time":"2016-09-28T11:41:11.065333829Z"}

  {"log":"Java ERROR...\r\n","stream":"stdout","time":"2016-09-28T11:41:11.065578959Z"}

  {"log":"Java Programm Running\r\n","stream":"stdout","time":"2016-09-28T11:41:13.066559917Z"}

  {"log":"Java ERROR...\r\n","stream":"stdout","time":"2016-09-28T11:41:13.066681392Z"}

  {"log":"Java Programm Running\r\n","stream":"stdout","time":"2016-09-28T11:41:15.067241372Z"}

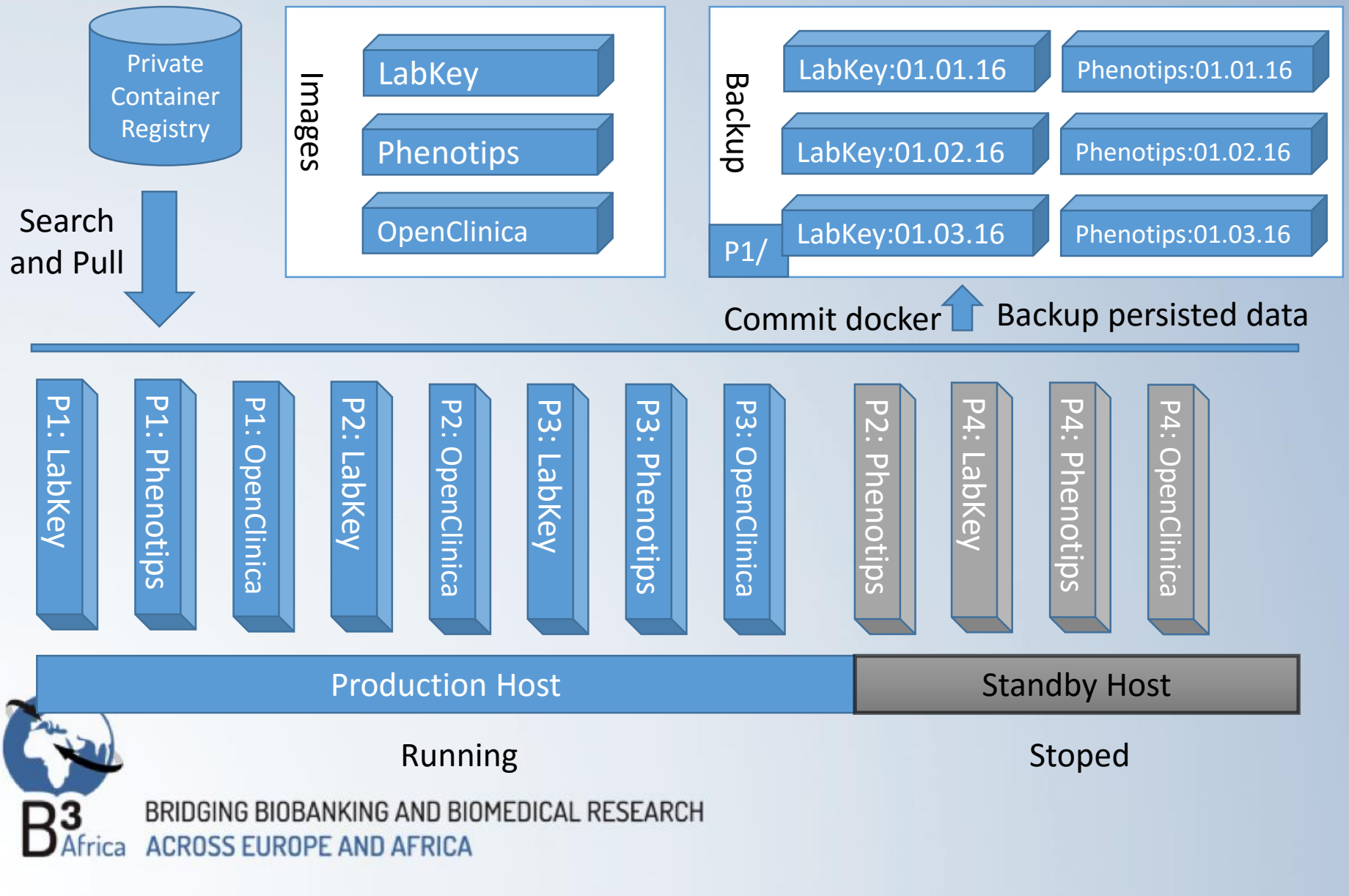
  {"log":"Java ERROR...\r\n","stream":"stdout","time":"2016-09-28T11:41:15.067357633Z"}

Docker logging

gelf

- docker run --name demo2 --log-driver=gelf --log-opt gelf-address=udp://172.23.0.3:12201 --log-opt tag="demo2" -dt -v /opt/webinar/test.jar:/opt/test.jar java:7 java -jar /opt/test.jar

Backup Docker



B3 Africa
BRIDGING BIOBANKING AND BIOMEDICAL RESEARCH ACROSS EUROPE AND AFRICA

Thank You!

Robert Reihs robert.reihs@medunigraz.at
Skype: ruppi83
B3Africa Slack

Post webinar exercise:

- Download the BIBBOX VM or install docker
- Run the bibbox/phenotips container
- Create a docker-compose file to run phenotip

BIBBOX - Links

- BIBBOX Community
  - http://bibbox.org
- Application Repository
  - https://github.com/bibbox/application-store
- Script Repository
  - https://github.com/BiBBox/bibbox-scripts
- Docker Image Repository
  - https://hub.docker.com/u/bibbox/dashboard

Glossary I

- VM – Virtual Machine
- OS – Operating System
- stdout - Standard Output
- stderr - Standard Error
- ETW - Event Tracing for Windows