WELCOME!

Virtualization and containerization at Swiss Post

WHAT ARE WE DOING TODAY?

- Introducing round
- Middleware Engineering
- Docker Introduction
- Docker Exercise
- Docker @ Post
- Docker Exercises
- Containerisation with Kubernetes incl. Demo
- Kuberneres @ Post

INTRODUCTION ROUND

PATRICK BÜHLMANN

Middleware Engineering

Task area:
- OpenShift Engineering
- Kubernetes Engineering
- DevOps CI/CD
- Docker / AWS

NATHANAEL LIECHTI

Lernender Middleware Engineering

Task area:
- OpenShift Engineering
- Kubernetes Engineering
- Automatisierung (GO)

PHILIP SAHLI

Middleware Engineering

Task area:
- Stv. Team leader
- Kafka / Go / Terraform
- Docker, AWS, Kubernetes

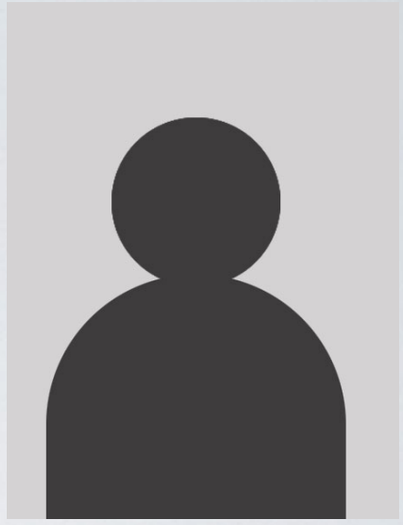
WHO ARE YOU?

What do you know about containers or dockers?

MIDDLEWARE-ENGINEERING















WHAT IS ENGINEERING?

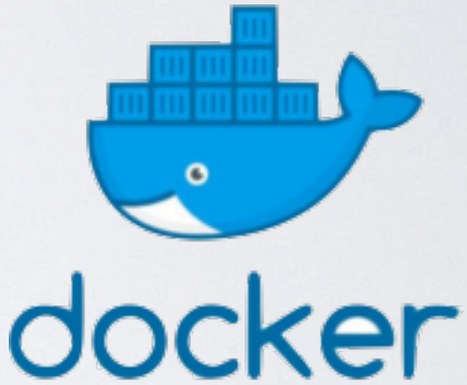
- Systematic design of computer science systems
- Create and implement concepts

ICT-System-Ingenieur:

Planning, procurement, commissioning, testing and acceptance of platforms (hardware, system software, networks, including cloud environments) for the operation of ICT systems; definition of the operational requirements

https://www.berufe-der-ict.ch/berufe/entwicklung/ict-system-ingenieur

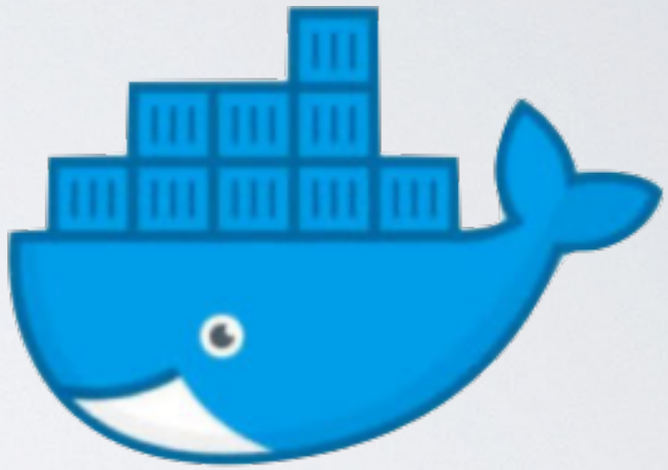
ENGINEERING TOOLS



WHAT DOES AN ENGINEER LIKE?

- Scripts, Automate
- Terminal, Shell
- Searching and developing new solutions
- Try something new
- Questioning the Existing
- Interest in system / network

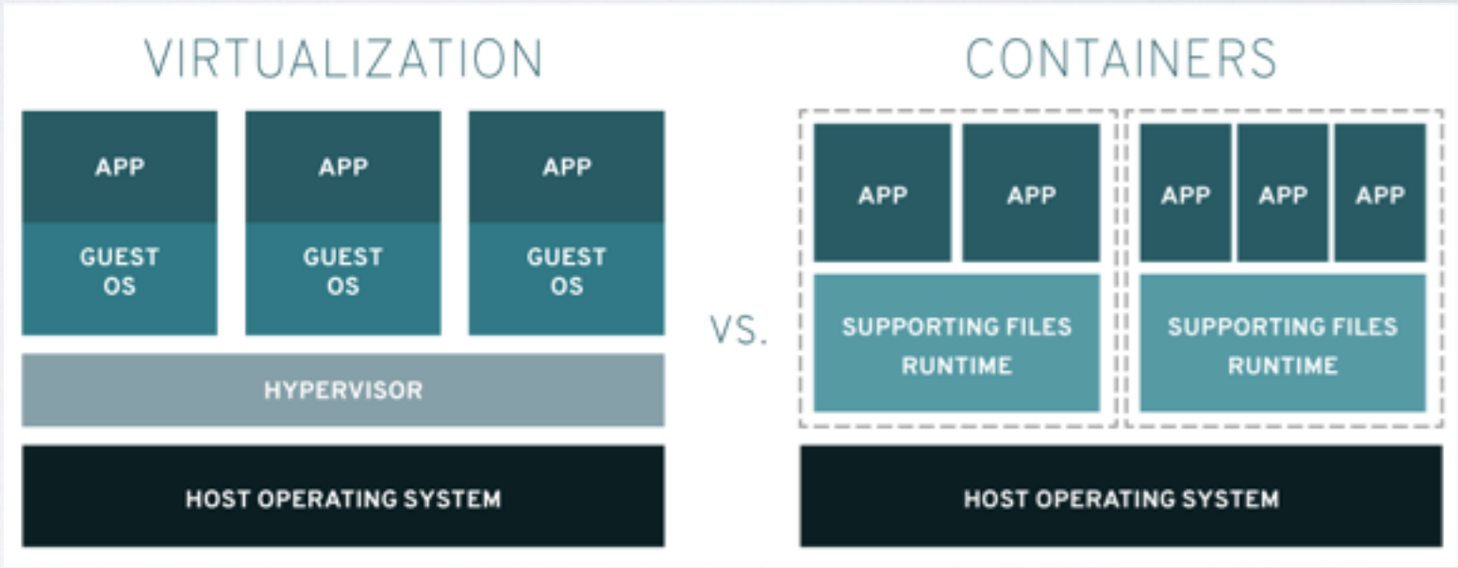
DOCKER INTRODUCTION



docker

WHAT IS A CONTAINER

- Share kernel and isolate the application processes
- Extremely portable
- Very lightweight
- One process in a container
- No data in a container (use volumes)



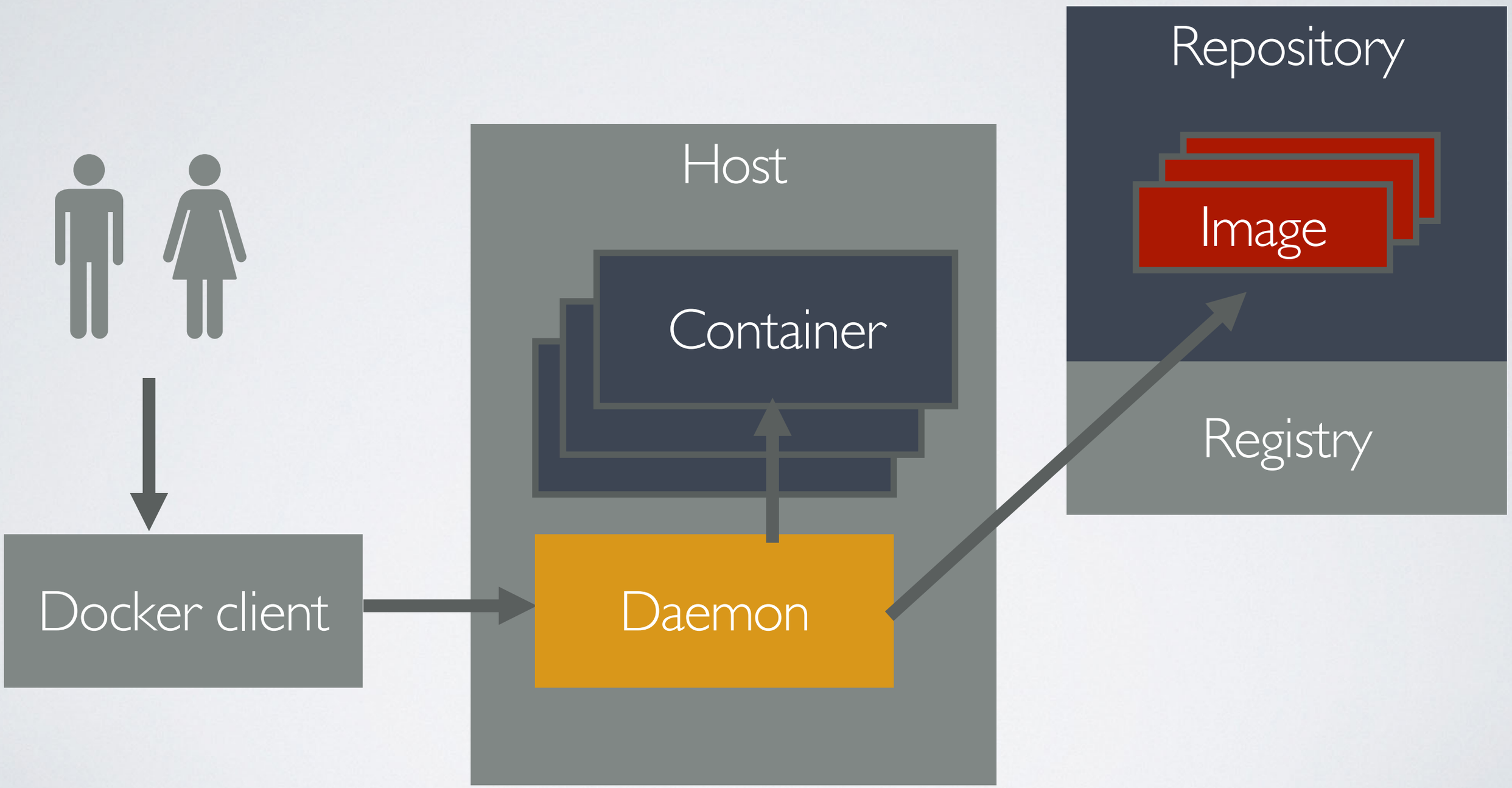
Quelle: https://www.redhat.com/en/topics/containers/whats-a-linux-container

DOCKER

Docker is a software platform designed to **make it easier** to create, deploy, and **run applications** by using containers. It allows **developers to package up** an application with all the parts it needs in a container, and then ship it out as **one package**.

build, share and run any application, anywhere

DOCKER


Repository
Image
Host
Container
Registry
Docker client
Daemon

DOCKER PRODUCTS

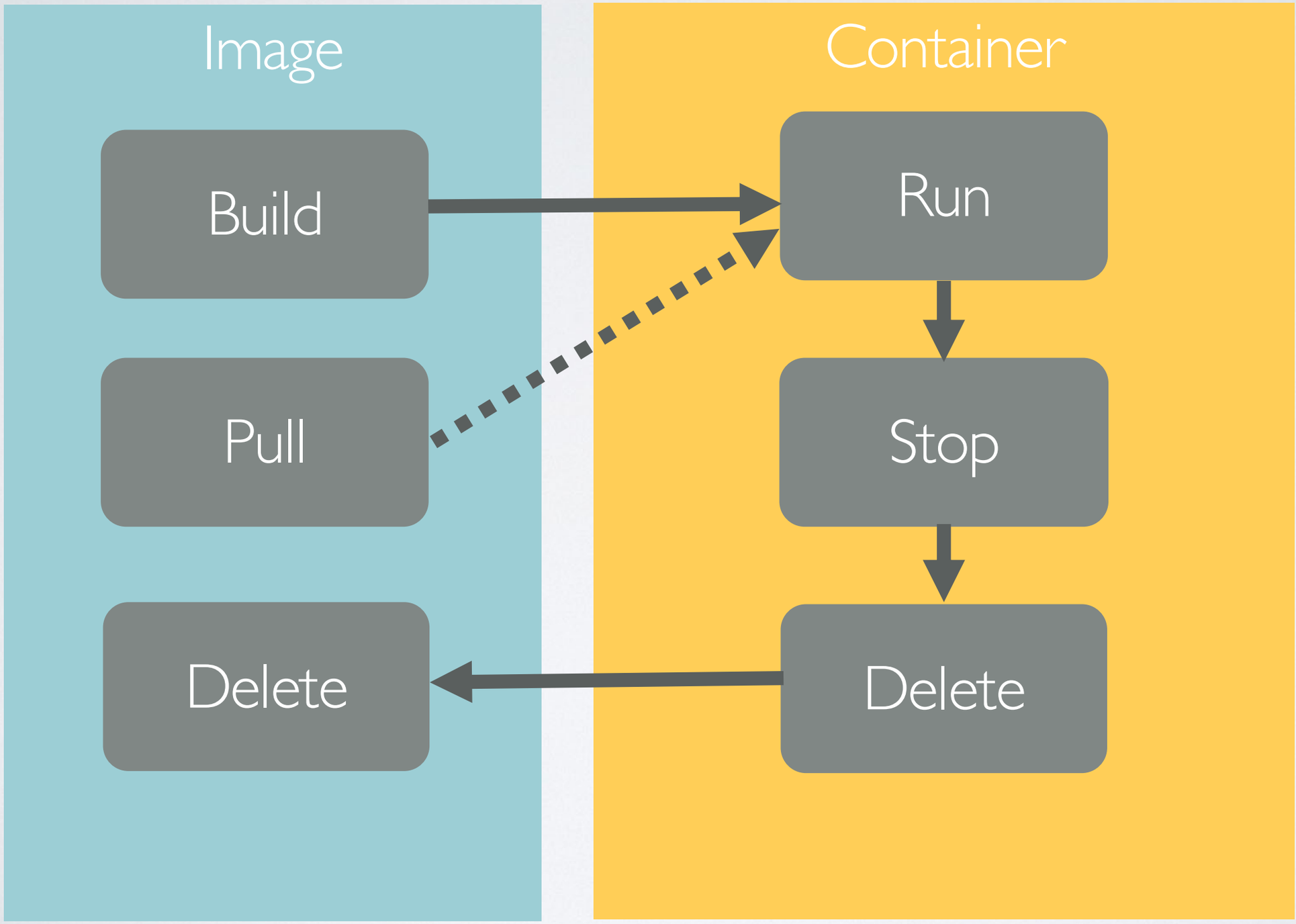
- Engine
  → Container Runtime Environment

- Compose
  → Multi-container applications

- Swarm
  → Clustering Docker Daemons

- Machine
  → Provision Docker Hosts

WORKFLOW



DOCKER ENGINE COMMANDS

- build
- pull
- tag
- rmi
- run
- start
- logs
- inspect
- ps
- stop
- kill
- rm

DOCKER RUNTIME ENVIRONMENT VARIABLES

- Environment Variables

```
sudo docker run -it \
      -e GREET=HELLO \
      —-rm centos
```

```
[root@eb558447bb9b /]# env|grep GREET
GREET=HELLO
[root@eb558447bb9b /]# echo $GREET
HELLO
```

DOCKER RUNTIME NETWORKING

- Port translation

```
sudo docker run … \
        -p 10022:22 …
```

DOCKER RUNTIME VOLUMES

- Mount a directory from the host into the container

```
sudo docker run … \
        -v /data/db:/var/lib/db …
```

DOCKERFILE

The docker file is the truth

- Instruction-based image build
  - FROM, ADD, RUN, COPY,CMD, USER
- Cached layers

LAB ENVIRONMENT I

Login	popcorn	microwave

LAB ENVIRONMENT 2

```
popcorn@popcorn:~$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
popcorn@popcorn:~$ docker pull alpine
Using default tag: latest
latest: Pulling from library/alpine
ca3cd42a7c95: Pull complete
Digest: sha256:ec14c7992a97fc11425907e908340c6c3d6ff602f5f13d899e6b7027c9b4133a
Status: Downloaded newer image for alpine:latest
docker.io/library/alpine:latest
popcorn@popcorn:~$ docker images
REPOSITORY   TAG       IMAGE ID       CREATED      SIZE
alpine       latest    49f356fa4513   7 days ago   5.61MB
popcorn@popcorn:~$ cowsay "Hello Docker!"
 ---------------
< Hello Docker! >
 ---------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||
```

LAB ENVIRONMENT 3

```
popcorn@popcorn:~$ docker run -ti --rm alpine sh
/ # rm -rf /*
```

OVERVIEW EXERCISES

- HelloWorld → Image builden
- Webserver → Build und Compose
- Gogs → start Git Userinterface

DOCKER EXERCISE:

„HELLO WORLD“

EXPLANATION OF THE EXERCISE

1. Choose the right & small base image (alpine, centos)
2. Create bash script
3. Build and run the container

- is there a faster way?

DOCKER @ DIE POST

DOCKER @ POST

Key figures

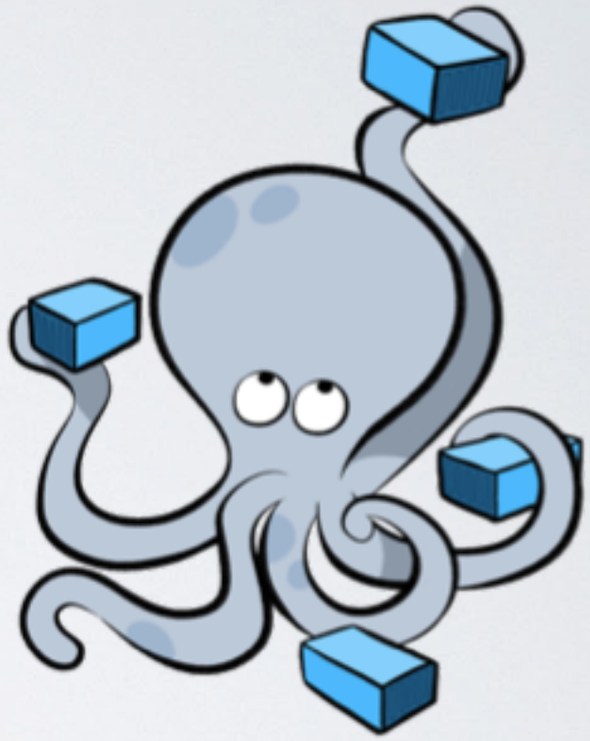
- Docker-Hosts: 257 Stk.
- Docker-Container: 1450 Stk.

Application purpose

- Portainer (Monitoring)
- Jenkins Master & Slaves (Tool Chain)
- Artifactory, Bitbucket, Jira and Confluence
- Automation

DOCKER COMPOSE

- Description in YAML format for containers
- Automatically builds image before creating container
- Used for stacks with more than one container (cache -, database -, web container)
- Scale up (create multiple web containers)



DOCKER EXERCISE: „WEBSERVER“

EXPLANATION OF THE EXERCISE

1. Choose the right & small base image (alpine, centos)
2. Install an apache http server
3. Create an index.html file and paste it to the web location
4. Build and run the container in the detached mode (background)
5. Start and stop the images „helloworld“ and „webserver“ with a single docker-compose file

DOCKER EXERCISE:

„GOGS“

PHILIP

1. Docker Image is published on https://hub.docker.com/r/gogs/gogs/

2. Watch out for „Ship with docker“

3. Pull the image / Run a container (see Usage section)

    1. Mount a directory onto /data

    2. Map Port 10022 to 22

    3. Map Port 10080 to 3000

4. Open UI

    1. Configure Installation (SQLite, Database Type, Domain, Application URL)

    2. Sign up

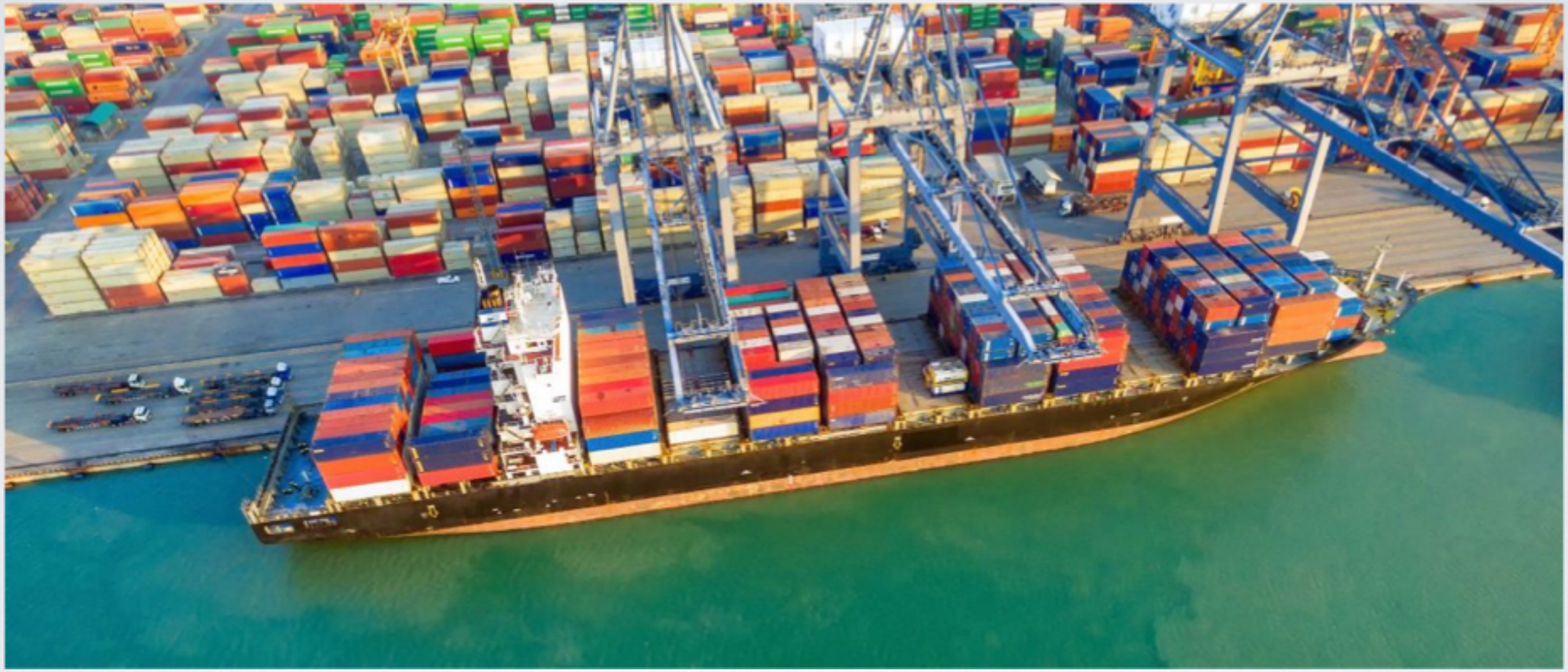
5. Create a repository

6. Clone the repository on the docker host

CONTAINERISATION WITH KUBERNETES

WHAT DOCKER LOOKS LIKE:

WHAT KUBERNETES LOOKS LIKE:



WHAT IS KUBERNETES

User Experience

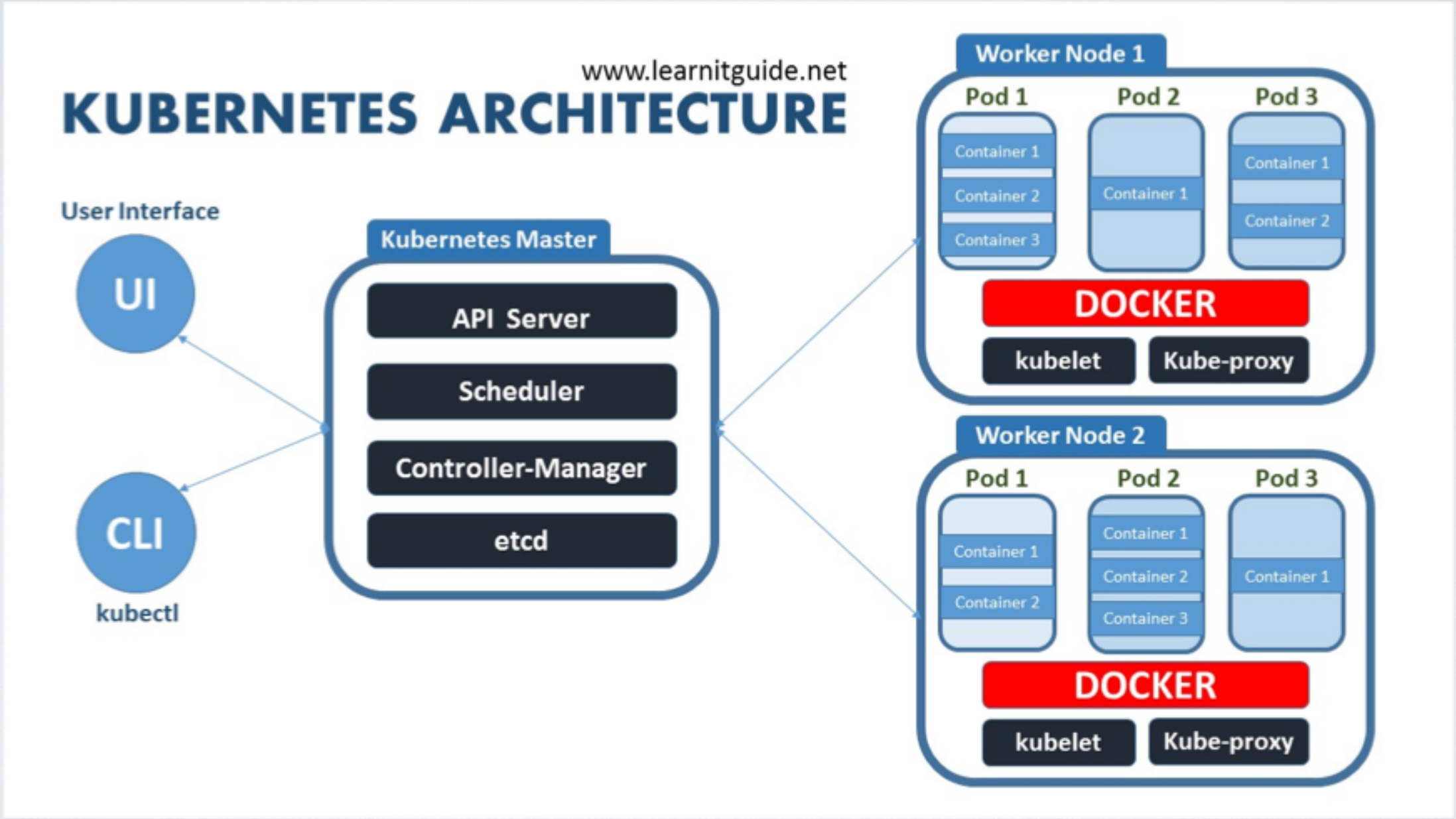
kubectl

Orchestration

Container Engine

Container Host

WHAT IS KUBERNETES



LIVE DEMO

Kubernetes

KUBERNETES @

KUBERNETES @ POST

Key figures

- Kubernetes Pods: 2'850
- Kuberentes Nodes: 148

Application purpose

- PostAuto Backend
- Chatbot & Livechat
- Build Pipeline
- Automated & Failsafe

ROADMAP CONTAINER

- New On-Prem Plattform

- Investigation in Kubernetes on Azure (AKS)

- Continuous Development on AWS EKS

QUESTIONS?

mweng@post.ch