

CONTAINER TECHNOLOGIES

Docker and Singularity

Introduction

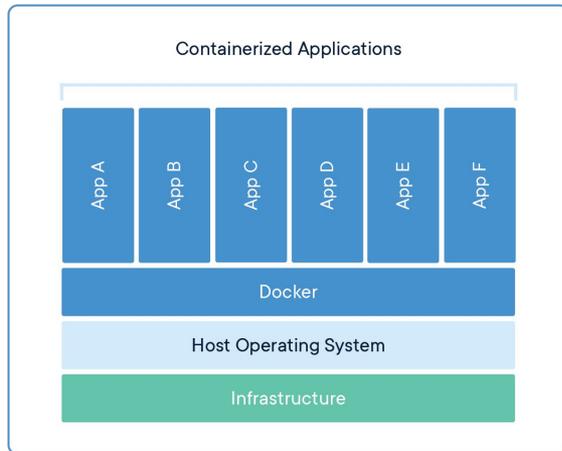
- 1. Container - Overview**
- 2. Container und VMs**
- 3. Docker**
- 4. Singularity**
- 5. Docker v Singularity**

Container - Overview

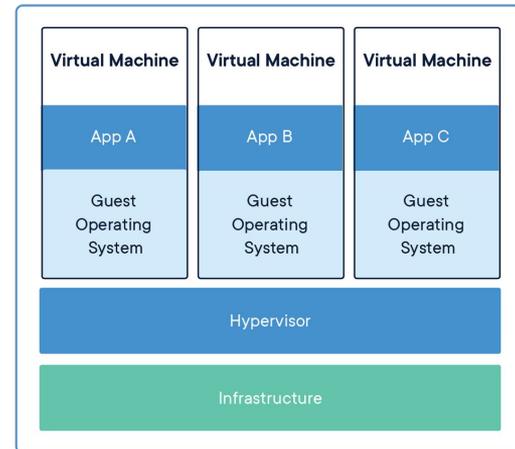
- Autonomer Betrieb
- Aufhebung von externen Abhängigkeiten
 - System (OS)
 - Plattform
 - Netzwerk
 - Prozesse
- Hohe Portabilität und Flexibilität
- ressourcensparender Betrieb
 - geringer Overhead
- Skalierbarkeit
- Isolierung von Microservices



Container und VMs



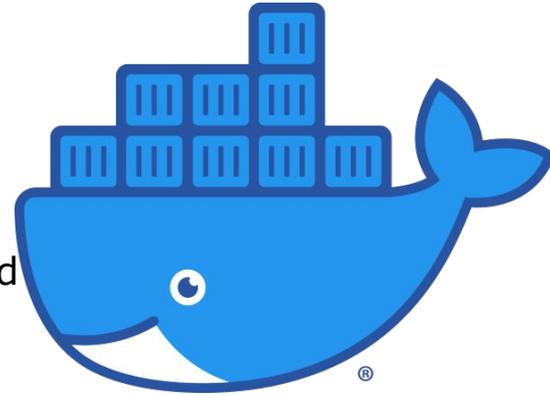
- Mehrere Container laufen auf einer Maschine
- Container teilen sich OS Kernel
- Isolierte Prozesse



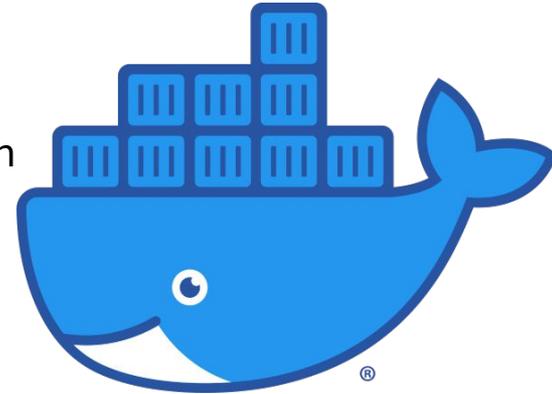
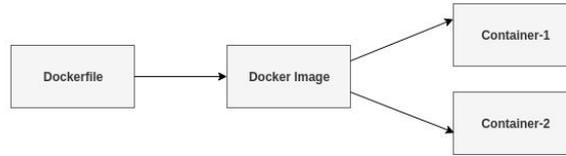
- Mehrere VMs laufen auf einer Maschine
- Jede VM betreibt ein komplettes OS
- Alle Abhängigkeiten müssen in VM installiert werden

Docker

- 2013 - Docker Release (Open Source) by dotCloud
- 2014 - Google initiiert Kubernetes Orchestrierung von Docker Containern
- 2022 - Netflix, Spotify, Paypal, Uber
- Jährliche DockerCon



Docker



- Dockerfile
 - Blueprint nach dem der Container zusammen gebaut wird
- Docker Image
 - Ein zusammengesetzter Container
- Linking
 - Docker Container können über offene Ports oder Ordner miteinander kommunizieren
- Docker Container
 - Container haben einen abgeschlossenen Zweck und beenden sich mit erfüllen selber
- Docker Daemon (docker -d)
 - Für Server und Services die weiterlaufen müssen

Docker

```
# syntax=docker/dockerfile:1
FROM node:12-alpine
RUN apk add --no-cache python2 g++ make
WORKDIR /app
COPY . .
RUN yarn install --production
CMD ["node", "src/index.js"]
EXPOSE 3000
```

Dockerfile

build



Docker Image

run



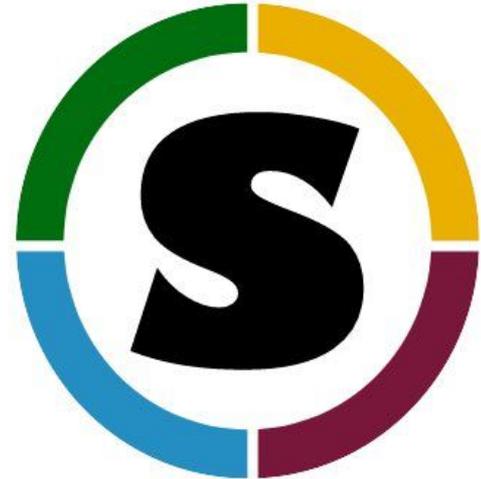
Docker Container

```
$ docker build -t getting-started .
```

```
$ docker run -dp 3000:3000 getting-started
```

Singularity

- 2016 - Release (Open Source)
 - Lawrence Berkeley National Laboratory
- Eingesetzt von Harvard, Stanford, Fraunhofer-Institut
- Fokus auf wissenschaftliches Rechnen und HPC
- Integriert / Integrierbar in existierende Resource Manager und Scheduler die von HPC System genutzt werden
- Dockerfiles können in Singularity-files umgewandelt werden



Vergleich Docker - Singularity (HPC)

	Docker	Singularity
Focus	<ul style="list-style-type: none"> • Entwicklung und DevOps • Microservices, einzelne Services 	<ul style="list-style-type: none"> • Wissenschaft • komplizierte Stacks, komplizierte Abhängigkeiten
Konzept	<ul style="list-style-type: none"> • Isolierte Container 	<ul style="list-style-type: none"> • Integrierte Container
HPC	<ul style="list-style-type: none"> • Kaum Interaktion mit Ressourcen Verwaltung möglich • Sicherheitslücken 	<ul style="list-style-type: none"> • Interaktion mit anderen Containern und Host möglich • keine Root-Rechte benötigt auf Production-System
Ohne HPC	<ul style="list-style-type: none"> • State of the Art Tech 	<ul style="list-style-type: none"> • Nicht genutzt

Quellen

1. <https://hpc.fau.de/files/2020/05/2020-05-12-singularity-containers.pdf>
2. <https://www.rackspace.com/de-de/blog/containers-101>
3. <https://t3n.de/magazin/ueber-container-technologie-wissen-musst-docker-gehts-240047/#:~:text=Docker%20geh%C3%B6rt%20auf%20den%20Tisch,Container%20zwischen%20Plattformen%20zu%20verschieben.>
4. <https://dzone.com/articles/understanding-dockerfile>
5. <https://www.docker.com/company/newsroom/media-resources>
6. <https://docs.docker.com/>
7. https://syllabs.io/guides/3.0/user-guide/quick_start.html