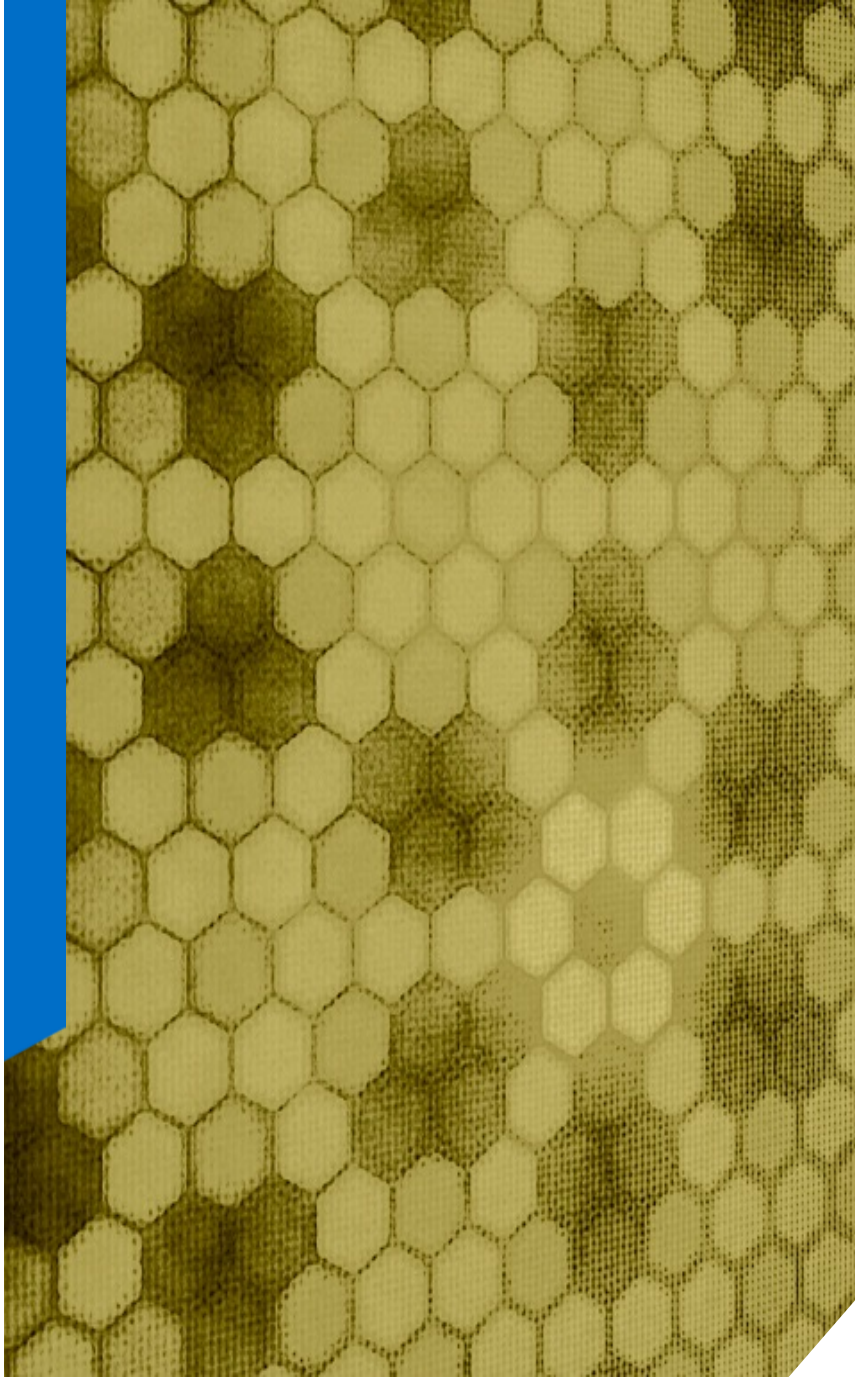
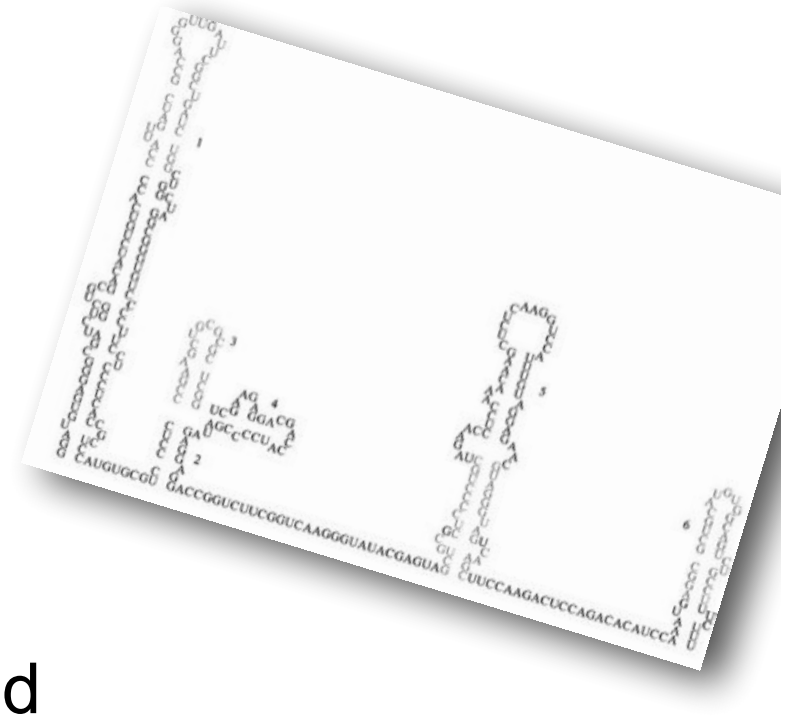


adesso business. people. technology.

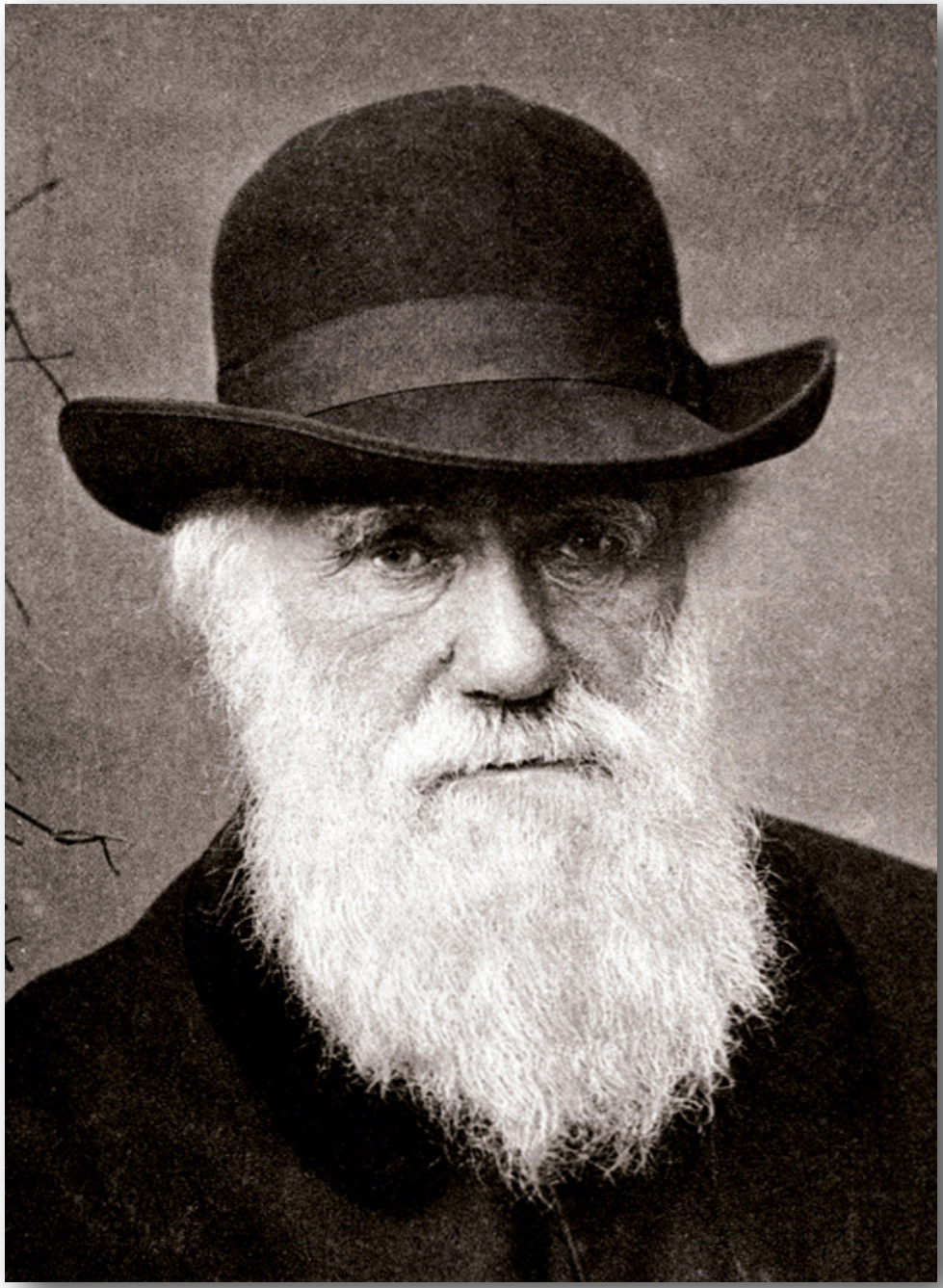
Docker revisited - Leichtgewichtige Orchestrierung

Dr. Halil-Cem Gürsoy
@hgutwit
adesso AG - Dortmund

Über mich



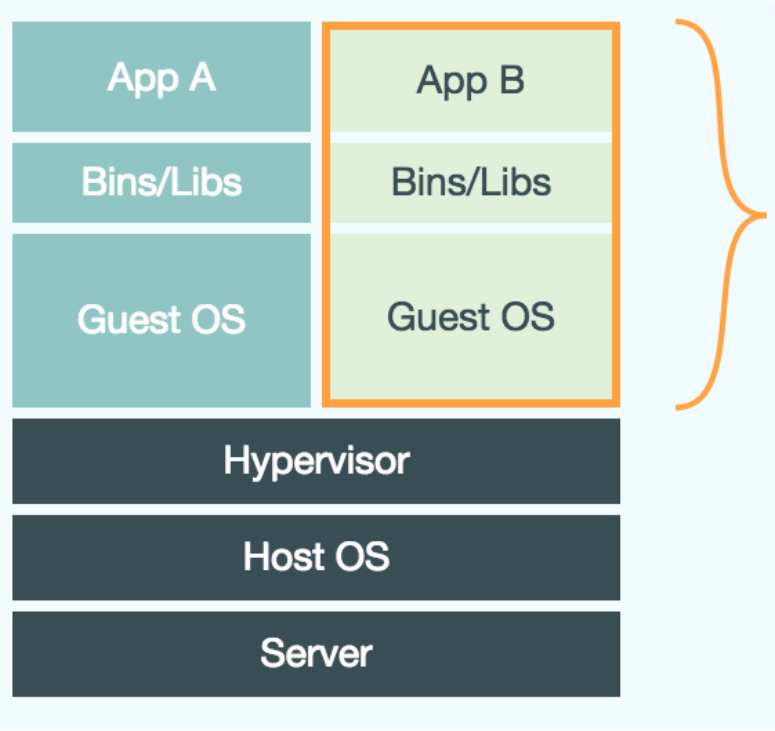
- Principal Architect @ adesso AG
- seit 15 Jahre Software-Entwicklung
  - > davor in wissenschaftlichem Umfeld
- Verteilte Enterprise-Systeme
- Persistenz / Build & Deployment

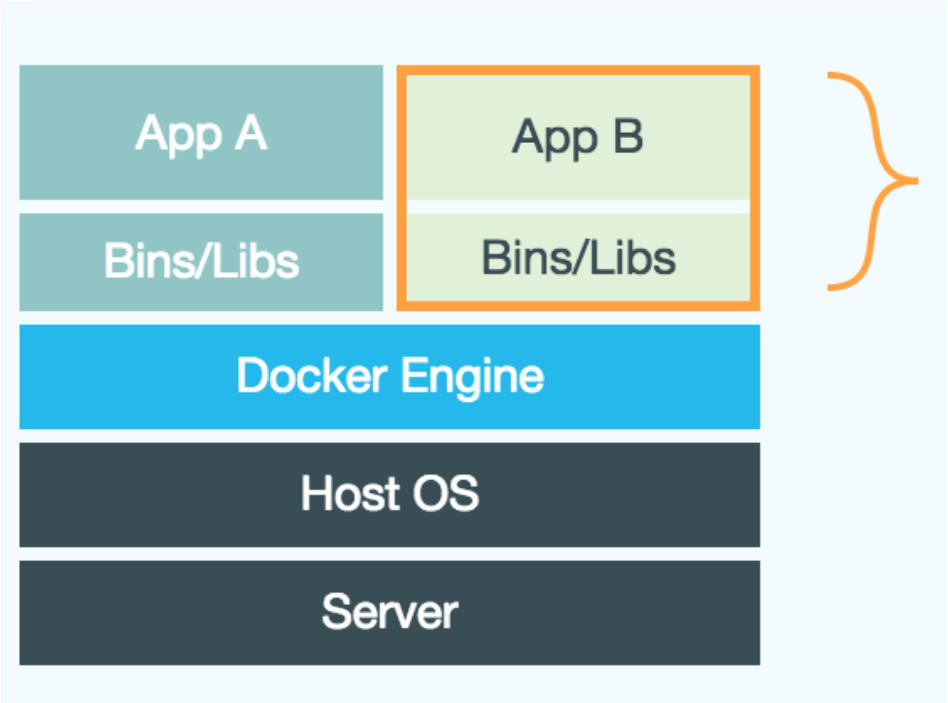


„Our highest priority is to satisfy the customer through early and ***continuous delivery*** *of* ***valuable software****.“*

Agile Manifesto Principles

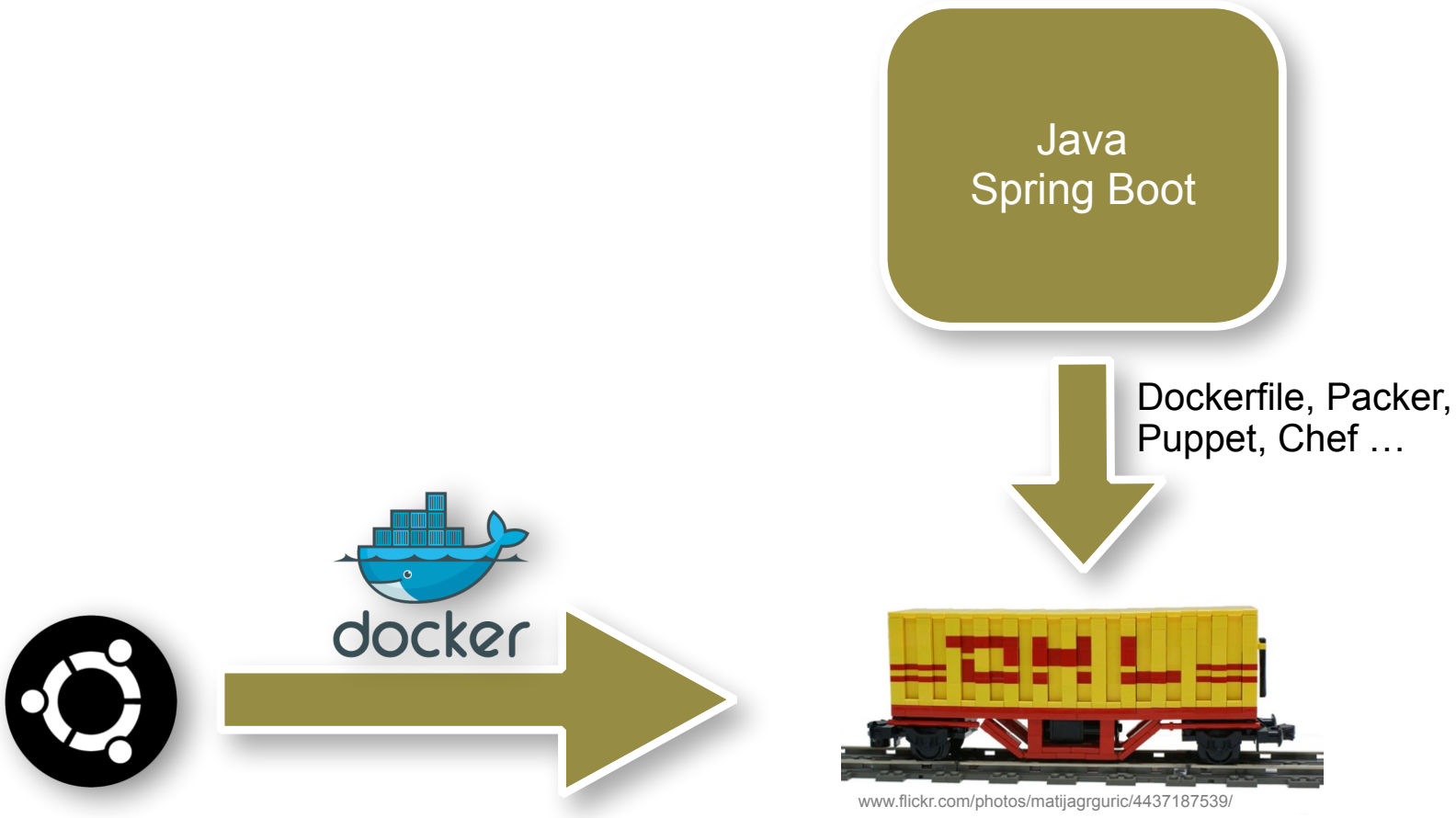
VM vs.Container



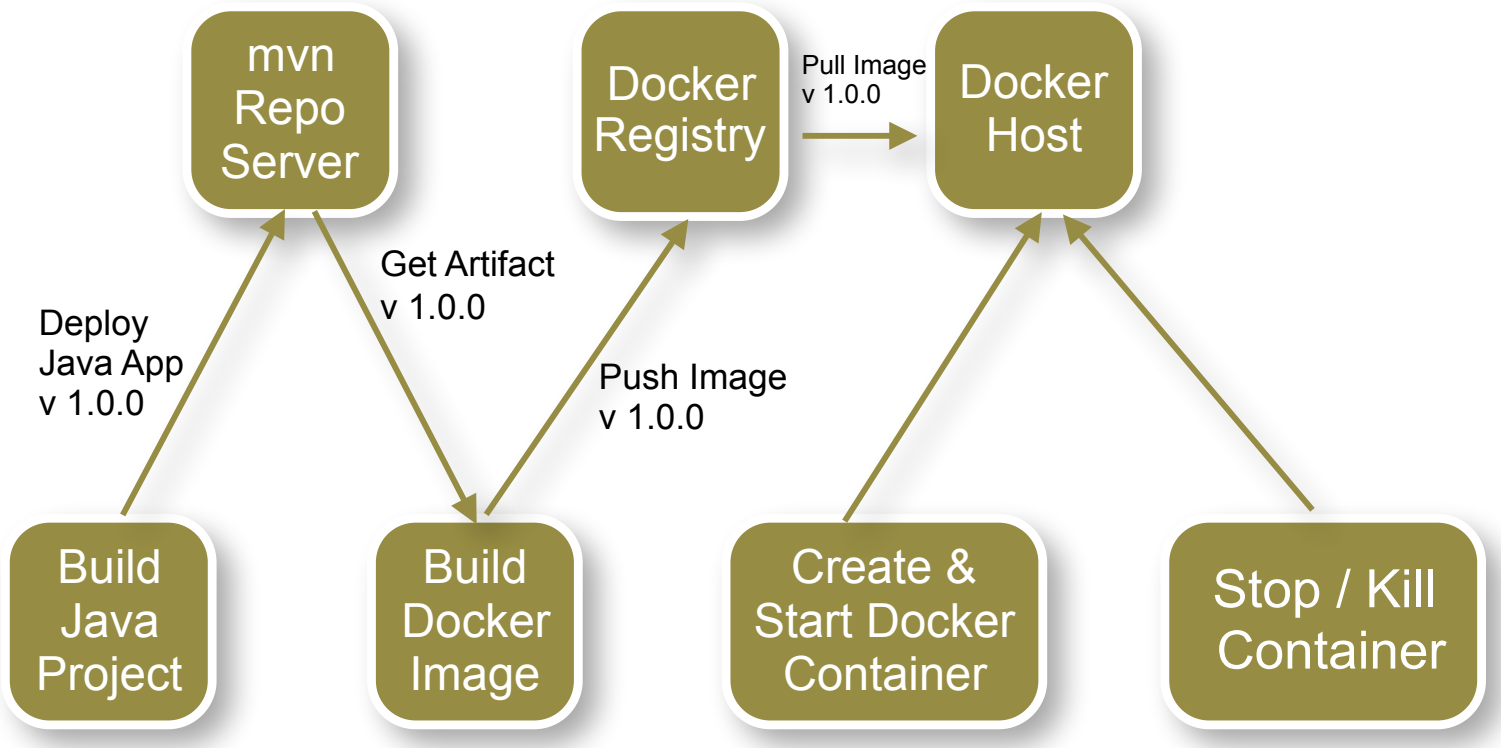


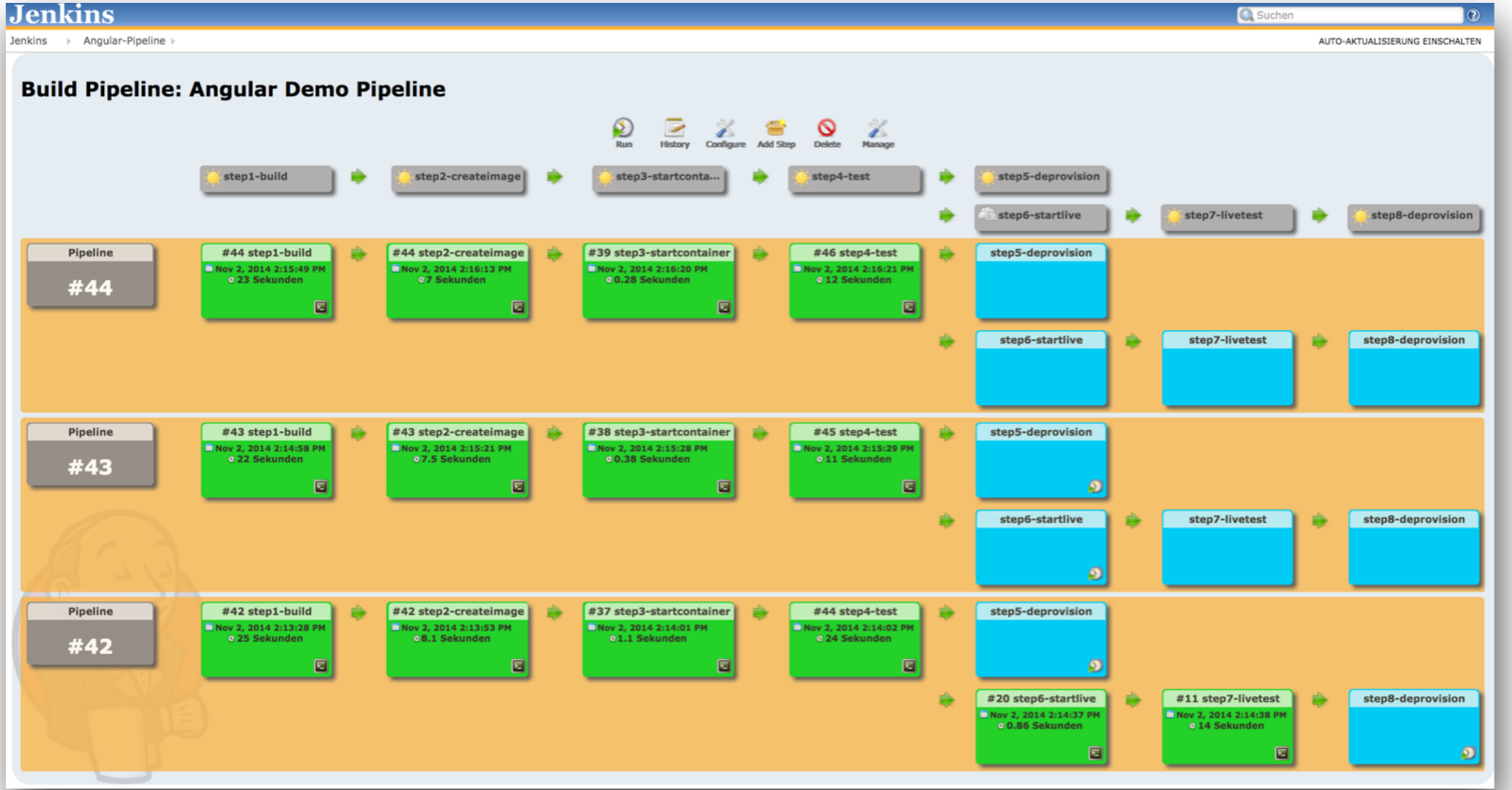
Quelle: http://docs.docker.com

Application Images



CD + Docker Workflow - supertrivial



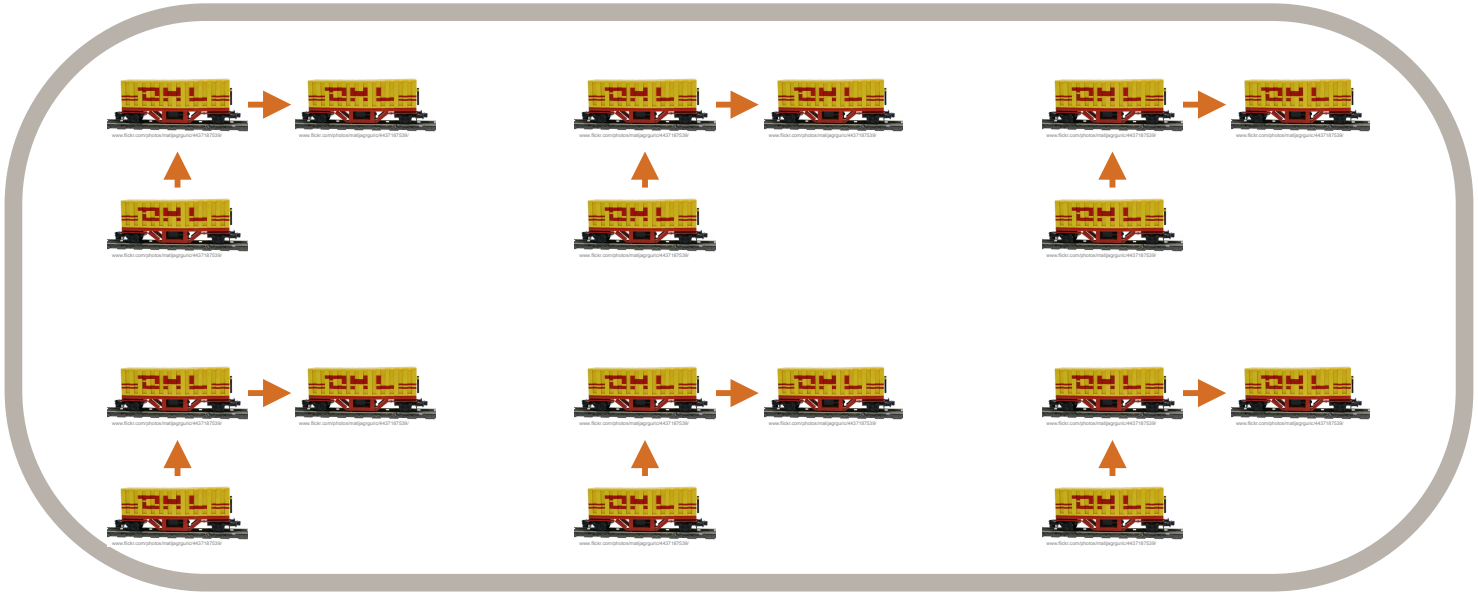

Jenkins
Suchen
Jenkins
Angular-Pipeline
AUTO-AKTUALISIERUNG EINSCHALTEN
Build Pipeline: Angular Demo Pipeline
Run
History
Configure
Add Step
Delete
Manage
step1-build
step2-createimage
step3-startconta...
step4-test
step5-deprovision
step6-startlive
step7-livetest
step8-deprovision
Pipeline
#44
#44 step1-build
Nov 2, 2014 2:15:49 PM
23 Sekunden
#44 step2-createimage
Nov 2, 2014 2:16:13 PM
7 Sekunden
#39 step3-startcontainer
Nov 2, 2014 2:16:20 PM
0.28 Sekunden
#46 step4-test
Nov 2, 2014 2:16:21 PM
12 Sekunden
Pipeline
#43
#43 step1-build
Nov 2, 2014 2:14:58 PM
22 Sekunden
#43 step2-createimage
Nov 2, 2014 2:15:21 PM
7.5 Sekunden
#38 step3-startcontainer
Nov 2, 2014 2:15:28 PM
0.38 Sekunden
#45 step4-test
Nov 2, 2014 2:15:29 PM
11 Sekunden
Pipeline
#42
#42 step1-build
Nov 2, 2014 2:13:28 PM
25 Sekunden
#42 step2-createimage
Nov 2, 2014 2:13:53 PM
8.1 Sekunden
#37 step3-startcontainer
Nov 2, 2014 2:14:01 PM
1.1 Sekunden
#44 step4-test
Nov 2, 2014 2:14:02 PM
24 Sekunden
#20 step6-startlive
Nov 2, 2014 2:14:37 PM
0.86 Sekunden
#11 step7-livetest
Nov 2, 2014 2:14:38 PM
14 Sekunden

Application „Container“

https://www.flickr.com/photos/matijagrguric/4437187539

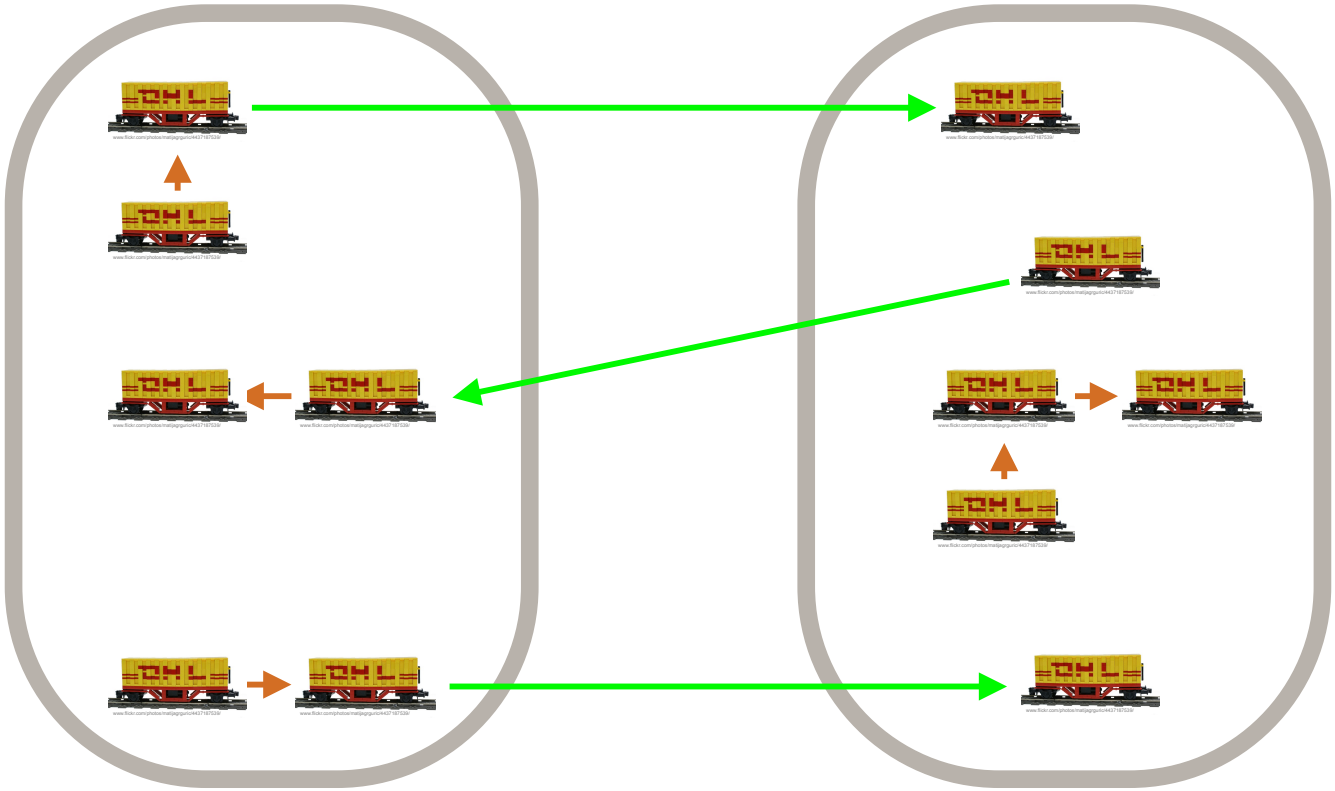
Eine Anwendung = viele Container

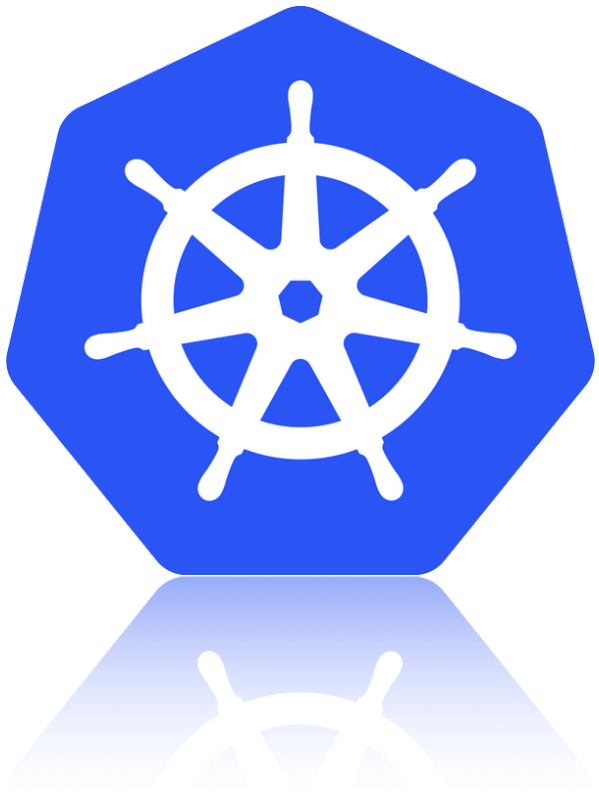
- Eine Applikation besteht nicht nur aus einem Container
  - > ein Container isoliert nur einen Prozess
- Module / Services in eigenen Containern
  - > Stichwort Microservices



Warum Orchestrierung?

- Container müssen miteinander kommunizieren
- Was, wenn Container auf Server verteilt werden?



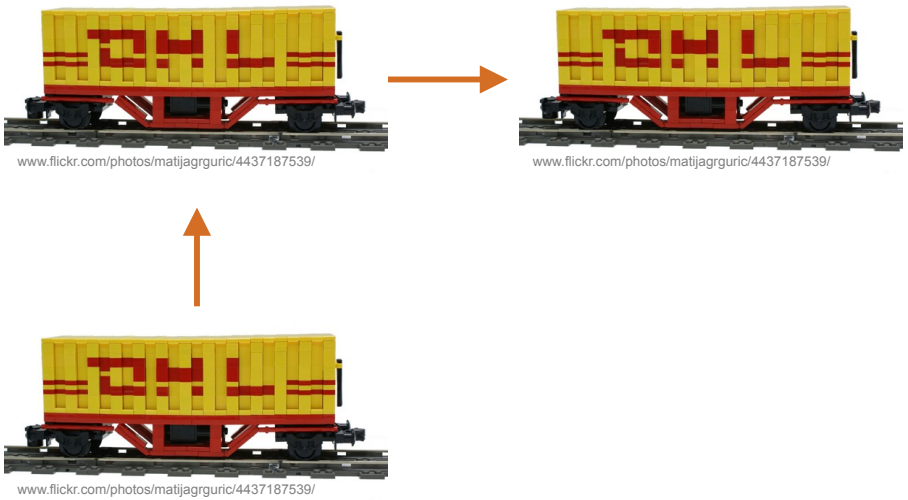


https://www.flickr.com/photos/tomwahlin/with/3167491292/

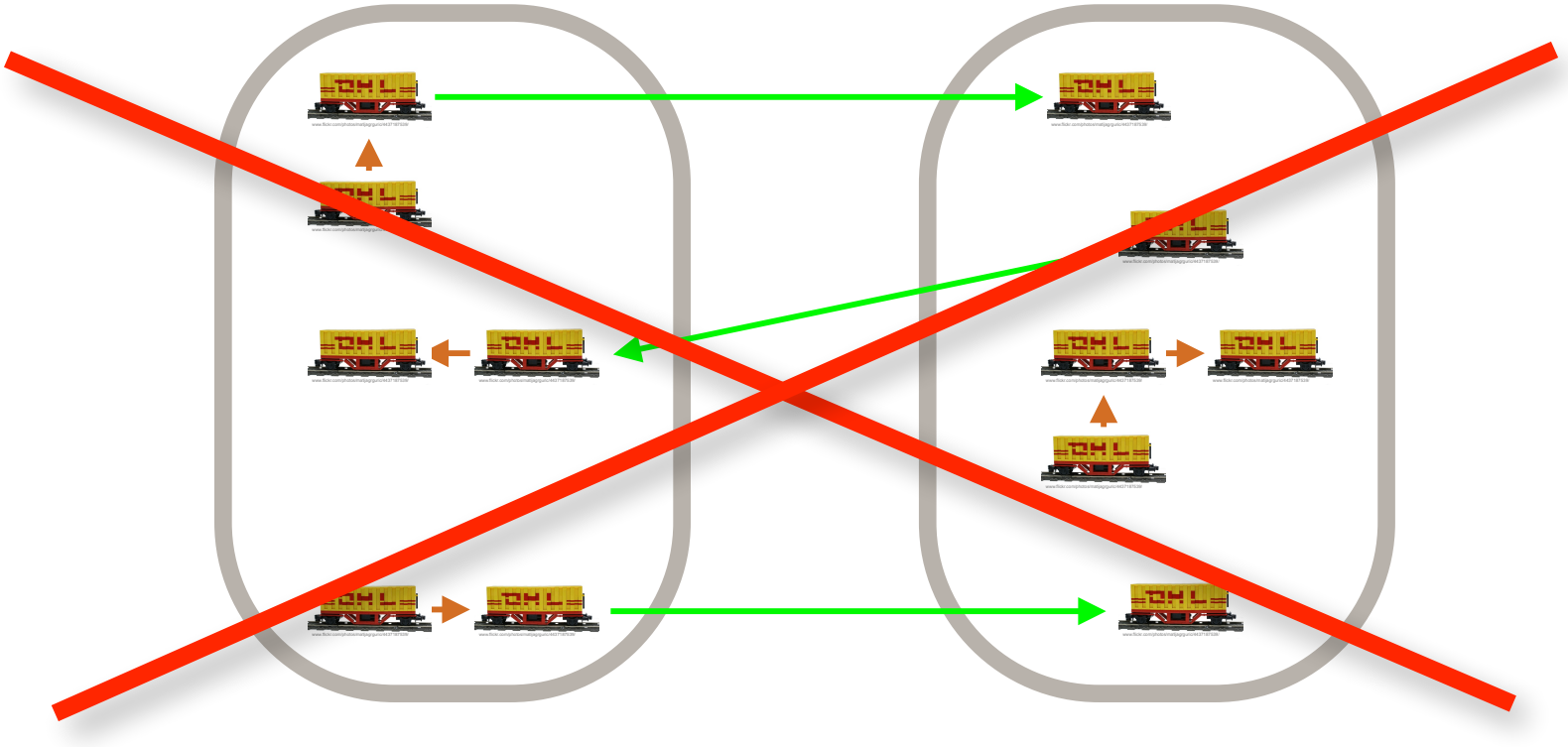

https://www.flickr.com/photos/ryandesiderio/1023387642

Container Links

```
docker run -d -P --name db42
docker run -d -P --name app42 --link db42:db

--link [name]:[alias]
```



Container Links

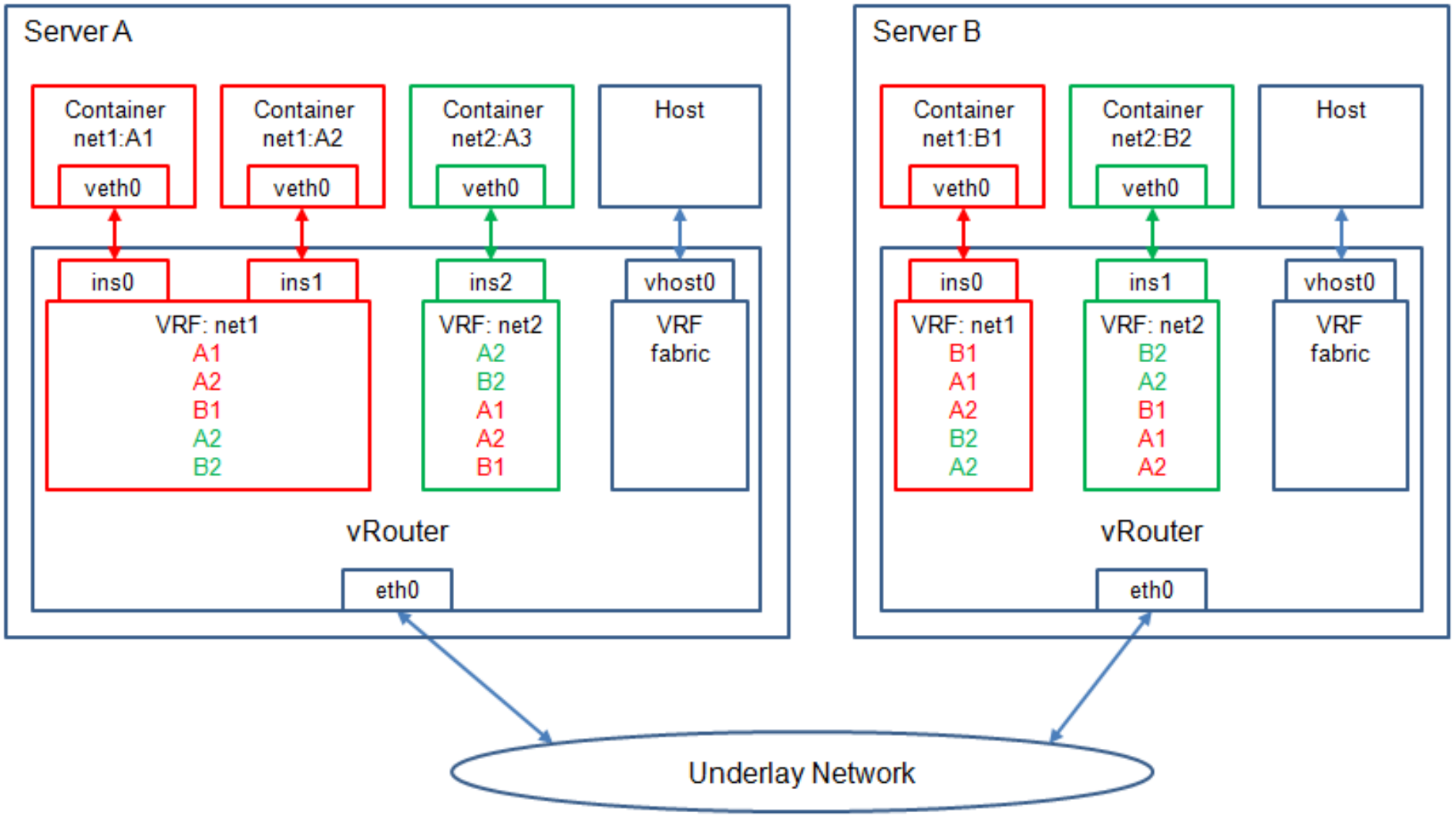



https://www.flickr.com/photos/zunami/3780220955/

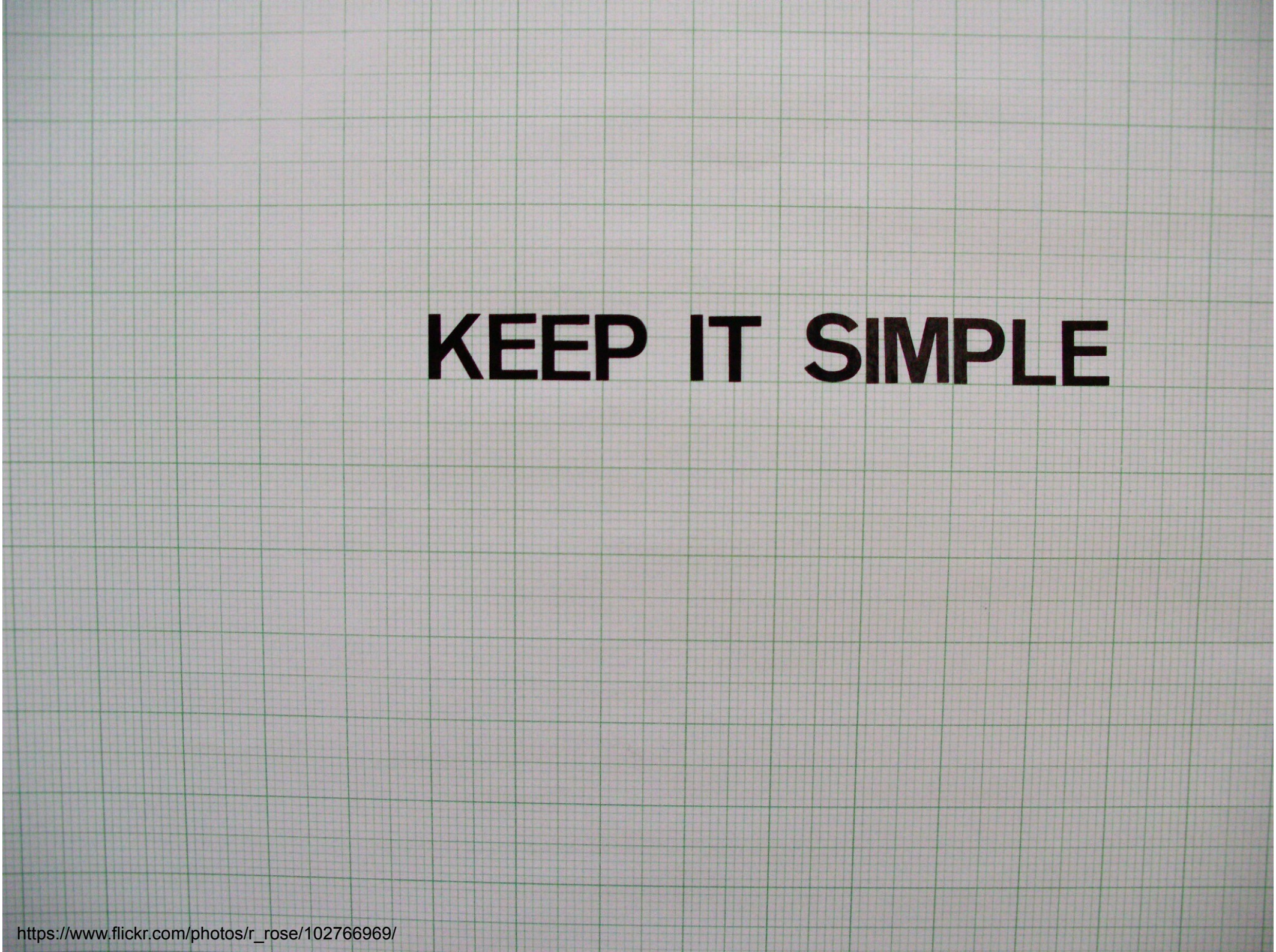
Was können wir tun?

- Ein eigenes Netzwerk für Docker aufziehen
- Tools gibt es … bestimmt ganz easy …
- Tunnel, Bridges, ….
- z.B. Handmade mit *OpenVSwitch*
  - > Auf jedem Docker Host wird Teil der IP-Range vergeben
- … oder mit Weave - *powerstrip-weave*
- … oder mit *OpenContrail*

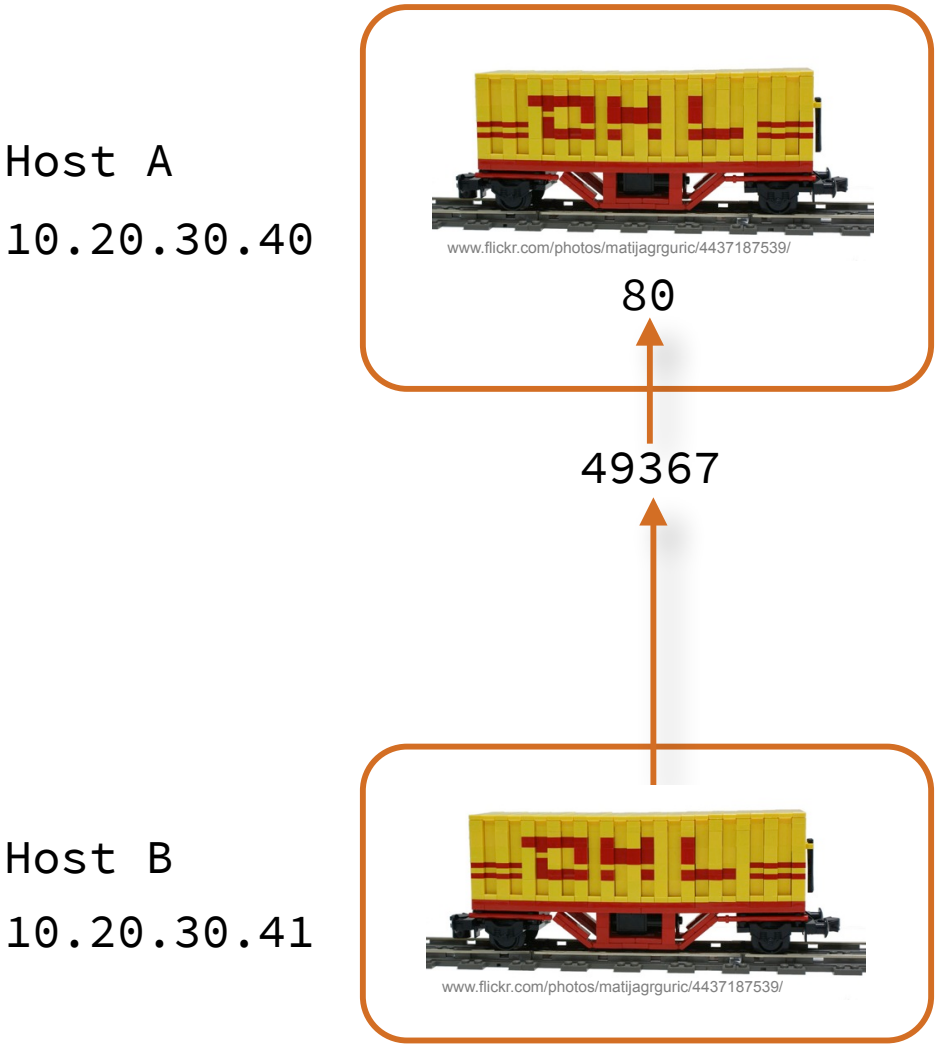
OpenContrail

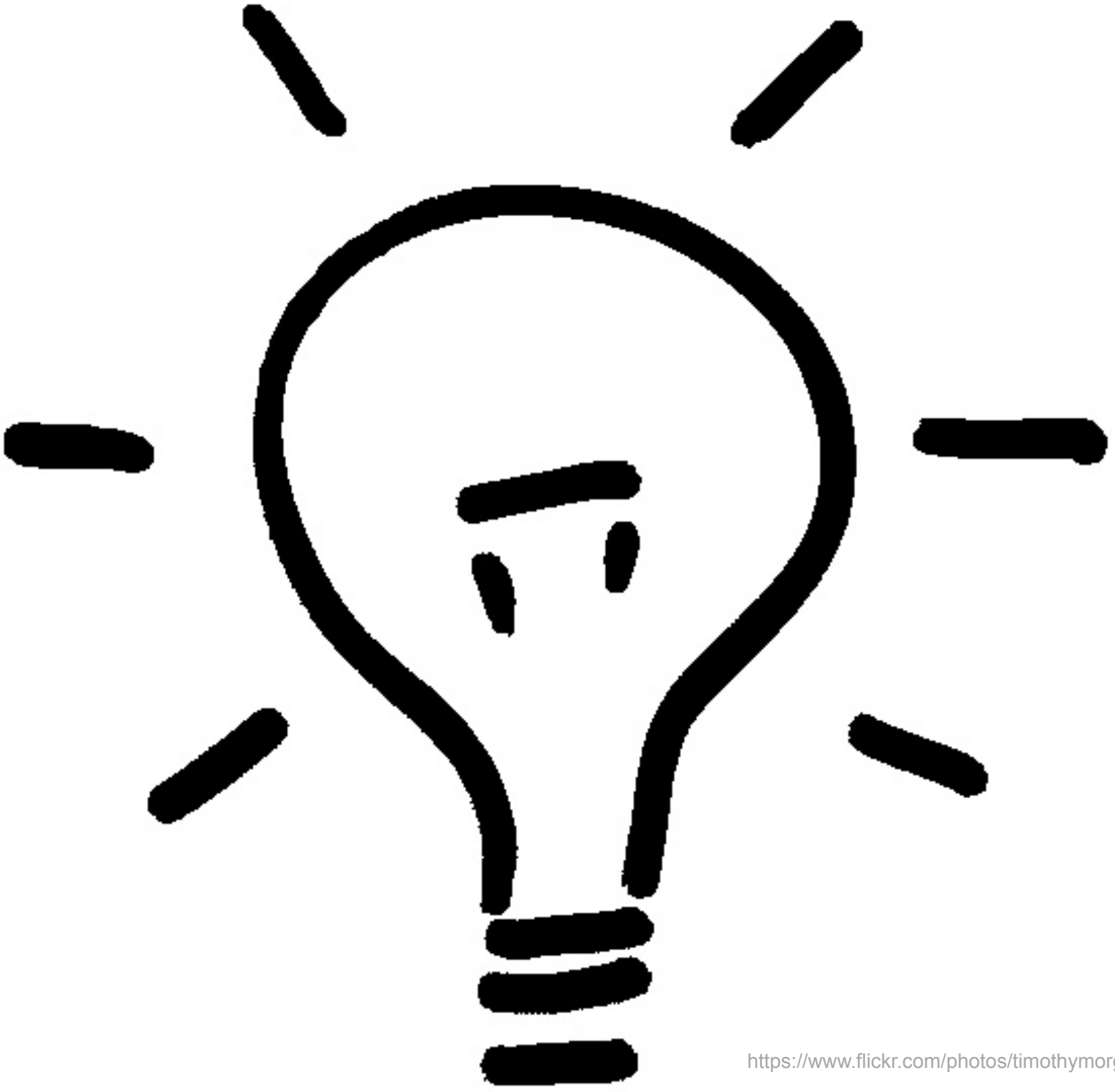


http://www.opencontrail.org/wp-content/uploads/2014/09/opencontrail-docker-figure-1.png



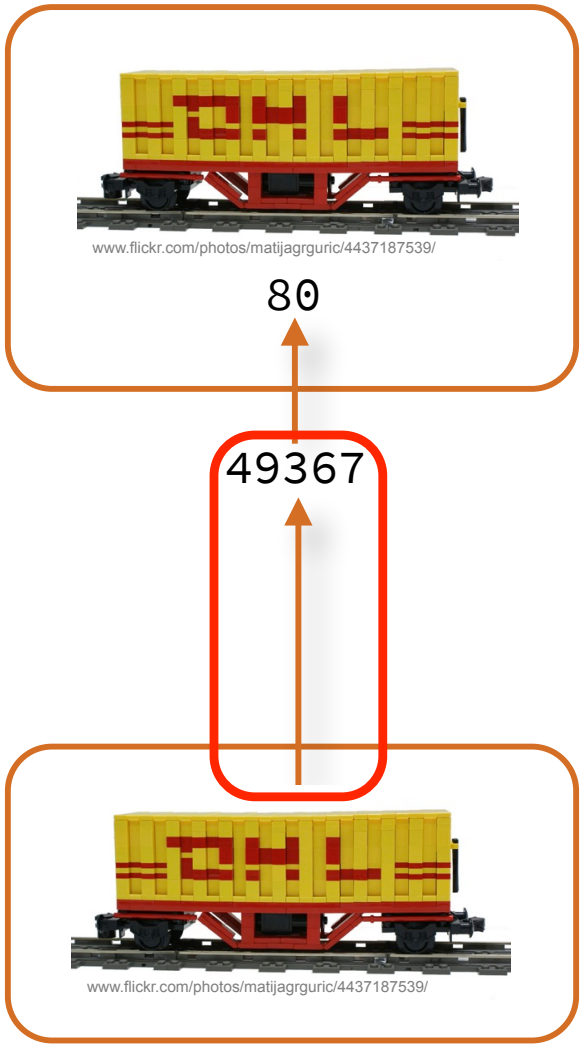
Anforderung ist doch ganz simpel…





https://www.flickr.com/photos/timothymorgan/4420821913

Anforderung ist doch ganz simpel…



adesso

Pure Docker nutzen!

- `docker run --add-host webserver:10.20.30.40`
  - fügt einen Eintrag in `/etc/hosts` hinzu
  - z.B. einen weiteren Docker Host
- Port über Umgebungsvariablen
  - `docker run -e webserver_port:49367`
- Nachteil
  - Unflexibel
  - größere Umgebungen sind nicht wartbar

https://www.flickr.com/photos/neilspicys/2348969927

Docker Machine

- Gut um (mehrere) Docker Hosts anzulegen und zu starten

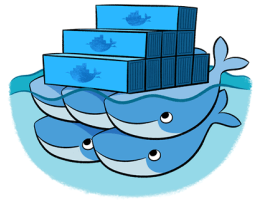
```
docker-machine create \
  --driver digitalocean \
  --digitalocean-access-token=$DIGITAL_OCEAN_TOKEN \
  test-machine
```

- Kommunikation zw. Containern auf verschiedenen Hosts damit noch nicht gelöst

- Also brauchen wir mehr…


docker
SWARM

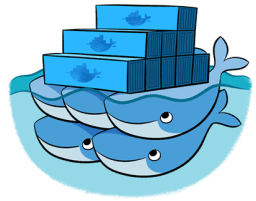
Docker Swarm



- Docker Swarm ist ‚die' Cluster-Lösung von Docker
- Mit Docker Machine einfach anzulegen

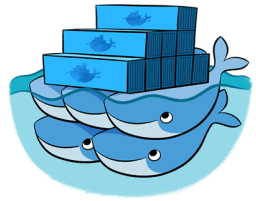
```
docker-machine create --driver digitalocean \
  --digitalocean-access-token=$DIGITAL_OCEAN_TOKEN \
  --swarm --swarm-master \
  --swarm-discovery token://$SWARM_ID swarm-master

docker-machine create --driver digitalocean \
  --digitalocean-access-token=$DIGITAL_OCEAN_TOKEN \
  --swarm --swarm-discovery token://$SWARM_ID \
  swarm-node-01
```

Docker Swarm



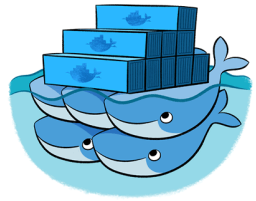
- Cluster-Setup mit verschiedenen Discoverys möglich
  - statische Liste von Nodes
  - Hosted auf Docker Hub
  - Consul
  - etcd
  - Zookeeper
- Mit Consul Unterstützung für Recovery von Nodes, Health-Check usw.

Swarm Manager



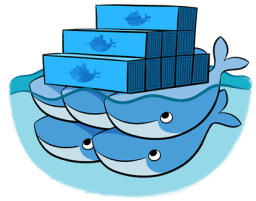
- Der Swarm Manager fungiert als ‚Proxy‘
  - Swarm API entspricht weitestgehend Docker API
- …und verteilt Container auf Knoten in einem ‚Cluster‘
- Verteilung abhängig von z.B.
  - Auslastung CPU und Memory
  - Label der Docker-Hosts
  - Affinität zu anderen Containern

Swarm Manager - Scheduler Strategies



- Scheduler arbeitet auf Basis von *Strategies* und *Filter*
- Aktuell drei Strategies:
  - *spread*
    - default, lastet gleichmäßig aus
  - *binpack* - macht einen Node nach dem anderen voll
  - *random*
- ‚naives' Überprovisionieren möglich

Swarm Manager - Scheduler Filter

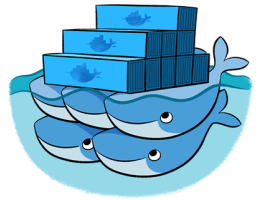


- Der Scheduler bietet diverse Filter zur Knoten-Auswahl
  - *Constraint* - Label auf einer Docker Engine
  - *Affinity* - Zusammen mit einer definierten Ressource
  - *Port* - Auf welchem Node ist Port xy noch frei?
  - *Dependency* - bitte mit Container XY zusammen
  - *Health* - nur auf gesunden Nodes

```
$ docker daemon --label storage=ssd
$ docker run -d -P -e constraint:storage==ssd --name db mysql
```

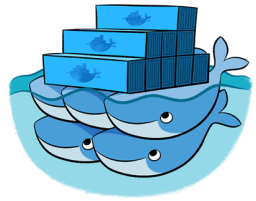
PAUSE

Docker Swarm mit Overlay-Netzwerk



- Ab Docker 1.9 unterstützt *libnetwork* ein Overlay-Netwerk
  - Netzwerk Node-überspannend möglich
- Virtuelles Netzwerk basierend auf *vxlan*
- Minimum Kernel 3.16
  - Kernel-Upgrade bei Ubuntu 14.04.3 nötig (oder Debian 8)
- Consul oder Etcd oder ZooKeeper als K/V-Store
- Hosts werden in allen Containern in */etc/hosts* eingetragen

Docker Swarm mit Overlay-Netzwerk



```
$ docker network create -d overlay multihost
$ docker network ls
NETWORK ID          NAME                DRIVER
36935a0674d4        multihost           overlay
```

- Beim Start des Containers Netzwerk mitteilen:

```
$DOCKER run -d -P --net="multihost" elasticsearch
```

- *libnetwork* ist Plugin-basiert
  - > *overlay*, *weave*
- Swarm 1.0.0 wird mit unterstützt

https://www.flickr.com/photos/schwaber/3238977103

Netzwerk ist nicht alles

- Flexibilität?
- Service Discovery einsetzen!
  - *Consul*, *etcd*, *SkyDNS 2*, …
- Container registriert sich im Service Discovery-System
  - automatisch mit *Registrator* (*)
  - … oder „manuell“ z.B. per REST-Call

Service Discovery einsetzen

- Zu Startzeit Konfigurationsdatei ‚befüllen‘
  - z.B. mit *confd* oder *consul-template*
  - Nachteil wenn Konfigurationen sich ändern
- … oder Umgebungsvariablen setzen
  - u.a. *consul-env*
- *DNS SRV* Request
  - Sowohl IP als auch Port, Auswahl aus mehreren
  - Muss meist in Applikation behandelt werden

Docker Compose

- Ehemals ‚Fig‘
- Python
- *„Compose is a tool for defining and running **multi**-container applications with Docker.“*
- Gut geeignet um komplexe Infrastruktur für Entwickler aufzubauen

Docker Compose - Konfiguration

- Konfiguration über YAML-Dateien:

```
wildfly:
  build: .
  ports:
   - "8080:8080"
  volumes:
   - .:/deploy
  links:
   - mysql
mysql:
  image: mysql
```

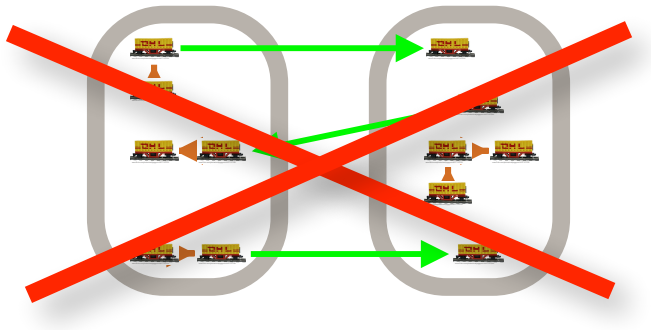
- `docker-compose up`

Docker Compose mit Swarm ?

- *‚Swarm is supported‘*
- … mit Hilfe von Container-Affinity
- *Eventually, Compose and Swarm aim to have full integration*

  … integration is currently incomplete

  (https://github.com/docker/compose/blob/master/SWARM.md)
- Multi-Host Netzwerk wird in 1.5.0 experimentell unterstützt



Alle Puzzleteile zusammen getragen….

- Docker Swarm als Scheduler
  - mit Consul als K/V-Store für den Cluster
  - …und zur Service Discovery
- Overlay-Netzwerk für Kommunikation zw. Containern
- Lose Kopplung!

Was aber noch fehlt…

- Self Healing
  - > was passiert wenn ein ganzer Knoten ausfällt?
  - > ‚Move‘ von laufenden Containern ohne Tools nicht möglich
- Auto Scaling
  - > was passiert wenn System unter Last steht?
- Übergreifendes Monitoring

Alternativen?

- Es gibt viele weitere Orchestrierung-Lösungen
  - maestro-ng
  - Apache Mesos
  - Crane
  - Rancher.io
  - Spotify Helios
  - ….

halil-cem.guersoy@adesso.de
https://twitter.com/hgutwit
https://github.com/hcguersoy/swarm-elastic-demo

http://www.flickr.com/photos/tcmorgan/7372944070