

DOCKER

UND SEINE BUILD-PROBLEME

UND WIE EIN GUTER PAKETMANAGER SIE LÖST

Stefan Langenmaier

ZUR PERSON

STEFAN LANGENMAIER

- Operations Engineer bei [makandra](#)



- Ruby Webentwicklung, Linux-Hosting sowie Beratung
früher
 - Berater für Suchmaschinen (Solr/Elasticsearch)
 - Consultant für Java-Webentwicklung

WAS IST DOCKER?

Gute Frage!

CONTAINERIZATION UNTERSCHIEDE/ALTERNATIVEN

- lxc/lxd
- docker
- ~~rkt~~
- ~~metfy~~
- containerd/runc
- cri-o
- podman
- uml
- OpenVZ
- Linux VServer
- systemd-nspawn
- flatpak/snap
- jails
- zones

DOCKERPROBLEME

Allgemeine Kritik an Docker in den [Referenzen](#).

[Buildprobleme](#)

DAS SPEZIFISCHE PROBLEM

- Experimentieren mit Containern/Images
- Erstellung eines Produktivsystems

BEISPIEL (PHUSION PASSENGER)

```
FROM phusion/baseimage:0.11
MAINTAINER Phusion <info@phusion.nl>

ADD . /pd_build
RUN /pd_build/install.sh
CMD ["/sbin/my_init"]
EXPOSE 80 443
```

Quelle

BEISPIEL (REDMINE)

Redmine 4.1 Dockerfile

```
FROM ruby:2.6-slim-buster
```

Ruby-Image-Tags

Wieso gibt es fünf verschiedene Varianten von ruby:2.6? Was versteckt sich dahinter?

BEISPIEL (REDMINE)

Redmine 4.1 Dockerfile

```
RUN set -eux; \
  savedAptMark="$(apt-mark showmanual)"; \
  apt-get update; \
  apt-get install -y --no-install-recommends \
    dirmngr \
    gnupg \
  ; \
  rm -rf /var/lib/apt/lists/*; \
  \
  dpkgArch="$(dpkg --print-architecture | awk -F- '{ print $NF }')"; \
  \
  # grab gosu for easy step-down from root
  # https://github.com/tianon/gosu/releases
  export GOSU_VERSION='1.12'; \
  wget -O /usr/local/bin/gosu "https://github.com/tianon/gosu/releases/download/$GOSU_VERSION/gosu-$dpkgArch"; \
```

Ziemlich lange Zeile

DOCKER BUILD CAPABILITIES

Nicht verfügbar*

MOUNTS

Nicht verfügbar

DOCKER MULTISTAGE BUILD

Langsam

RANT RESUMÉ

Bekannte Limitierungen - könnten technisch gelöst werden.

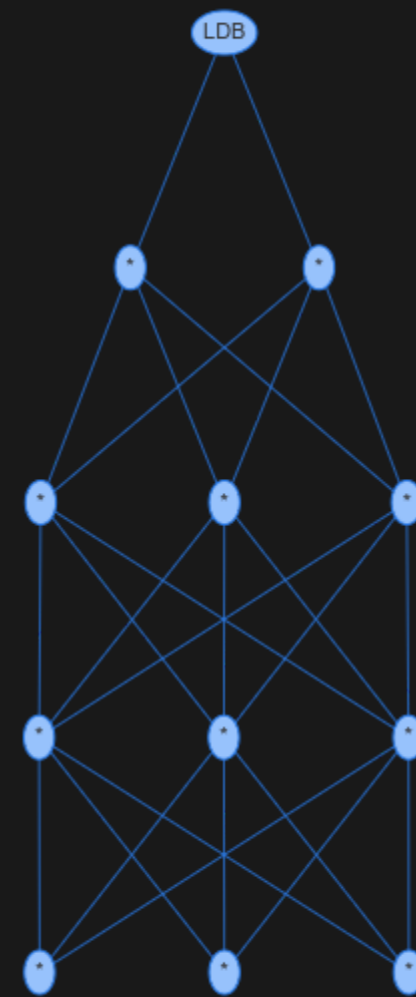
BEISPIELPROBLEM (NEXTCLOUD)

	Webserver-Interface	PHP	DB	Nextcloud	Sum
Kombinationen	2	3	3	3	
Docker (*)	2	6	18	54	80
Komposition (+)	2	5	8	11	11

LET'S VISUALIZE IT (STACKED)



LET'S VISUALIZE IT (COMPOSED)



DIESES PROBLEM KANN MAN IGNORIEREN, WENN MAN

- Fan von Monokulturen ist
- auf Vendor Lock-in steht

DIE LÖSUNG

gute Paketmanager

- Stabilität
- Support
- Best Practices
- Dokumentation

REKAPITULATION

Fokus auf die Erstellung der Images (mit "normalem" Paketmanager)

BEISPIEL - EIN IMAGE FÜR NEXTCLOUD

Referenz: das offizielle [Image \(20-apache\)](#).

SETUP

1. `export PORTAGE_CONFIGROOT=/var/build-root/portage-configroot/`
2. `export ROOT=/var/build-root/rootfs/`
3. `emerge @system nginx --jobs=5 -v`

SETUP II

1. `emerge nextcloud --onlydeps --jobs=5 -v`
2. `webapp-config -d nextcloud -s nginx -c -I nextcloud 20.0.5`

AUFRÄUMEN

1. `bash ./cleanup.sh`

IMAGE ERSTELLUNG

1. `sudo tar -c . | sudo docker import - <imagename>`

FUNKTION

1. `docker run --rm --tmpfs /run -it --name nextcloud -p 127.0.0.1:8080:80/tcp testimage /sbin/init`

DANKE - Q/A

LINKS/REFERENCES

- [Wikipedia: Operating-system-level Virtualization](#)
- [Container runtimes](#)
- [Microservices combinatorial explosion of versions](#)
- [Alpine-Docker & Python](#)
- [Profiling for image reduction](#)
- [Boycott Docker](#)
- [CoreOS and rocket](#)
- [packer](#)
- [buildah](#)
- [Container rant](#)
- [Docker without Docker](#)