31.8.–3.9.2015
in Nürnberg

Herbstcampus

Wissenstransfer
par excellence

Testen mit Containern

Docker für Integrationstests

Stefan Hildebrandt

consulting.hildebrandt.tk

TESTEN MIT CONTAINERN

DOCKER FÜR INTEGRATIONSTESTS

Stefan Hildebrandt / @hildebrandttk

VORTRAG MIT ALLEN BILDERN UNTER CONSULTING.HILDEBRANDT.TK/VORTRAEGE.HTM

PERFECT DEVELOPMENT AND TEST ENVIRONMENT?

NOT SHARED!

- developers
- testers
- ci-jobs

EASY SETUP

FULLY DEFINED

FAST STARTUP

GOOD TURNAROUND

- developers
- testers

LOW MAINTENANCE

FLEXIBLE SIZING

- As much as needed
- as little as possible

IN SYNC WITH SOURCE CODE

DO YOU KNOW DOCKER ?

USEFUL DOCKER FEATURES

CONTAINER

- no virtual machine -> no extra kernel
- no own init services (syslog, network...)
- process separation (linux process groups)
- own network (bridge device, iptables)
- very low overhead

DOCKER HOST

- localhost
- virtual machine
- remote host (computing center / cloud service)

IMAGES

- incremental layers
- reuse of layers
- name
- tag

REGISTRY

- central repository for images
- public
  - Docker Hub
- private
  - reference impl. V1 / V2
  - Docker Hub Enterprise
  - Artifactory
  - dogestry: Distributed load/store

IMAGES COMPRESSION

- registry: compressed
- network transfer: compressed
- docker host: unpacked

DOCKER NETWORK

- br
- iptables
- NAT

PORT MAPPING

- images defines possible ports
- binding on container creation
- syntax: docker -p <HOST-Port>:<CONTAINER-Port>

INTER CONTAINER COMMUNICATION

- links
  - extra entries within /etc/hosts
- definition on container creation
- syntax: docker --link <other-container-name>:<alias-within-container>

SHARED DIRECTORIES

- native (bind) mount
- high performance
- image defines possible mount points
- binding on container creation
- syntax: docker -v <HOST-PATH>:<CONTAINER-PATH>

DATA ONLY CONTAINER

- container used by other(s) for data storage
- syntax: docker --volumes-from <DATA-CONTAINER>

ENVIRONMENT PARAMETERS

- image defines possible environment variables
- definition on container creation
- syntax: docker -e "deep=purple"

THAT'S A BIT MUCH!

DOCKER-COMPOSE

- definition of
  - runtime parameters
  - build parameters
- live cycle commands
  - build
  - start, up, kill, stop
  - rm
- for full environments

DEVELOPMENT WITH DOCKER

APPLICATION UNDER DEVELOPMENT

REMOTE DEPLOYMENT TO CONTAINER

↗ fast setup
↗ easy updates
↘ remote deployment
↘ remote debugging
↘ less ide support

DOCKER BUILD DEPLOYMENT

- ↗ fast setup
- ↗ easy updates
- ↗ defined deployment
- ↘ remote deployment
- ↘ remote debugging
- ↘ less ide support
- ↘ bad turnaround

DOUBLE CHECK REQUIRED TIMES AND BENEFIT FOR

- Setup
- Build
- Deployment
- Roundtrip
- Debugging

APPLICATION UNDER TEST

- arquillian tests: prefer embedded containers
- if not suitable: use same checklist as before

SUPPORT SYSTEMS

- relational databases
- nosql databases
- centralized logging
- ...

INTEGRATION TEST ENVIRONMENT

- peripheral system
- test double systems
- runtime for test tools
- pre defined test data
- hint: use only tagged images!
- hint: commit and push containers after failed test-run

PERIPHERAL SYSTEM

- best case: add existing image to docker-compose
- create ci job to build tagged application image
- example: docker-petclinic

TEST DOUBLE SYSTEMS

- simple, self developed systems
  - spring boot
  - dropwizard
  - Java EE
  - Rails
  - Grails
  - what ever
- soap ui dummy ($)
- central $$$ systems
  - could be proxied with additional credentials, routes, ...
  - don't forget the versions!!!

PRE DEFINED TEST DATA

- extra layer on system base container
- automated creation
- repeatable build
- don't use dump files
- put data to a location within container!
  - easy distribution
  - fast startup
  - fast rollback

JAILER

- defined database model
  - extraction from existing schema
  - defined-by-hand
- extraction of full entities
- custom selects, obfuscating, ...

COPY PRODUCTION (NEAR) DATABASE

- real test data
- obfuscate
- reduce to minimum
- automate

MYSQL

- start mysqld within container
- create tables
  - mysqldump --no-data <SOURCE_DB>| mysql <TARGET_DB>
- select and copy data
  - SET FOREIGN_KEY_CHECKS=0;
  - mysql -e "<QUERY>" <SOURCE_DB> | mysql <TARGET_DB>
  - SET FOREIGN_KEY_CHECKS=1;
- stop mysqld within container

ORACLE

- start oracle-xe within container
- create db-link
- copy tables
- copy data
- copy constraints, views, triggers, ...
- drop db-link
- stop oracle-xe
- example

DISTRIBUTE CONTAINER

- tag with repository-url
- docker push <FULL-TAG>
- docker pull <FULL-TAG>
- example

RUN CONTAINER

- examples
  - pure docker
  - docker-compose

MAVEN

- round about dozen plugins
- - rhuss/docker-maven-plugin
    - Build
    - Publish
    - run/start
    - fine grained configuration

GRADLE

- variable-definitions within "Dockerfile"'s
- linked multi-container-builds
- put maven-artifacts into the container

GRADLE DOCKER PLUGIN BY MATTHIAS GRÜTER

- uses des docker-cli-client or docker-java
- fokus on building containers
- simple
- buggy, some open pull requests on github -> needs a new master
- Examples

GRADLE DOCKER PLUGIN BY BENJAMIN MUSCHKO

- supports only docker-java
- many tasks for creating, tagging, deleting and pushing images
- for deleting on repositories too
- more complex syntax for those tasks

ARQUILLIAN-CUBE

- await
- reuse running container
- Example arquillan.xml

DATABASE MIGRATION

TYPICAL TOOLS

- flyway
- liquibase
- dbdeploy

EXECUTION TIME

- build-time-provision
- additional startup-provision

USECASES

- database migration
- testing database migration
- add/modify testdata

LOGIN INTO CONTAINER

- ssh
- nsenter

CONCLUSIONS

PRO

- very fast
- easy distribution
- easy handling for users

REQUIREMENTS

- linux
- disc space
- disc speed
- network performance

ISSUES

- Production Ready Registry
- cleanup
- chmod 755 run.sh && ./run.sh --> permission denied with AUFS
- Download from 127.0.0.1 without authentication


QUESTION
THE
ANSWERS

STEFAN HILDEBRANDT - CONSULTING.HILDEBRANDT.TK

- Beratung, Coaching und Projektunterstützung
- Java EE
- Buildsysteme gradle und maven/ant-Migration
- Testautomatisierung
- Coach in agilen Projekten
- DevOps