Oracle und Docker

Oracle Datenbanken in Docker Container

Stefan Oehrli

@stefanoehrli

www.oradba.ch

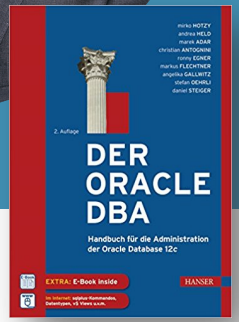
Stefan Oehrli

Plattform Architekt, Trainer und Partner bei Trivadis
- Seit 1997 in verschiedenen IT-Bereichen tätig
- Seit 2008 bei der Trivadis AG
- Mehr als 20 Jahre Erfahrung im Umgang Oracle Datenbanken

Fokus: Daten schützen und Datenbanken sicher betreiben
- Security Assessments und Reviews
- Datenbank Sicherheitskonzepte und deren Umsetzung
- Oracle Backup & Recovery Konzepte und Troubleshooting
- Oracle Enterprise User Security, Advanced Security, Database Vault, …
- Oracle Directory Services

Co-Autor des Buches Der Oracle DBA (Hanser, 2016/07)

@stefanoehrli www.oradba.ch



ORACLE ACE

BASEL | BERN | BRUGG | BUKAREST | DÜSSELDORF | FRANKFURT A.M. | FREIBURG I.BR. | GENF
HAMBURG | KOPENHAGEN | LAUSANNE | MANNHEIM | MÜNCHEN | STUTTGART | WIEN | ZÜRICH

trivadis

Agenda

trivadis

- Einleitung
- Docker Images, Container und Volumes
- Datenbank Image
- Datenbank Container
- Anwendungsfälle
- Lizenzierung
- Zusammenfassung

trivadis

Einleitung

Was ist Docker?

trivadis

- Eine Software-Containerplattform
  - Entstanden aus Linux / Linux-Containern
- Verfügbar…
  - … für Linux, Mac OS und Windows
  - … als Community Edition (CE) und Enterprise Edition (EE)
- Teil der Linux Open Container Initiative (OCI)
- Docker ist nicht die einzige Implementierung von Linux-Containern
  - Core OS / Rkt, LXC Linux Containers, OpenVZ, Mesos Containerizer,…
  - Docker hat den grössten Marktanteil
- Zielsetzung erhöhen der **Austauschbarkeit und Reproduzierbarkeit**

Oracle Support für Docker

trivadis

- Diverse Oracle-Produkte sind für Docker unterstützt und zertifiziert.
  - MOS Note 2216342.1 *Oracle Support for Database Running on Docker*
  - MOS Note 2017945.1 *Support Information for Oracle WebLogic Server and Oracle Fusion Middleware Running in Docker Containers*
- Container Oracle Linux 7 / UEK4 Kernel oder Red Hat Enterprise Linux 7 als Basis Image
- Oracle Quellen für Images oder offizielle Build Quellen
  - Oracle Docker Build Quellen auf GitHub https://github.com/oracle/docker-images
  - Oracle Container Registry https://container-registry.oracle.com
  - Oracle Container Engine and Registry https://developer.oracle.com/containers
  - Oracle auf Docker Hub https://hub.docker.com/publishers/oracle

Was gibt es sonst noch über Container?

trivadis

- Es gibt nicht DEN Container
- Single Process / Service
  - Kein Ersatz für VMs
- Geringer Platz Bedarf
- Verstehen, wie man Container baut, aber
  - ...nicht jeder Container muss individuell angepasst werden
  - ...verfügbare und vertrauenswürdige Container-Repositorys nutzen
- Umgang mit Identitäten und sensiblen Daten
- Umgang mit Daten persistent

Quelle: Volkswagen Advertising (June 2016)

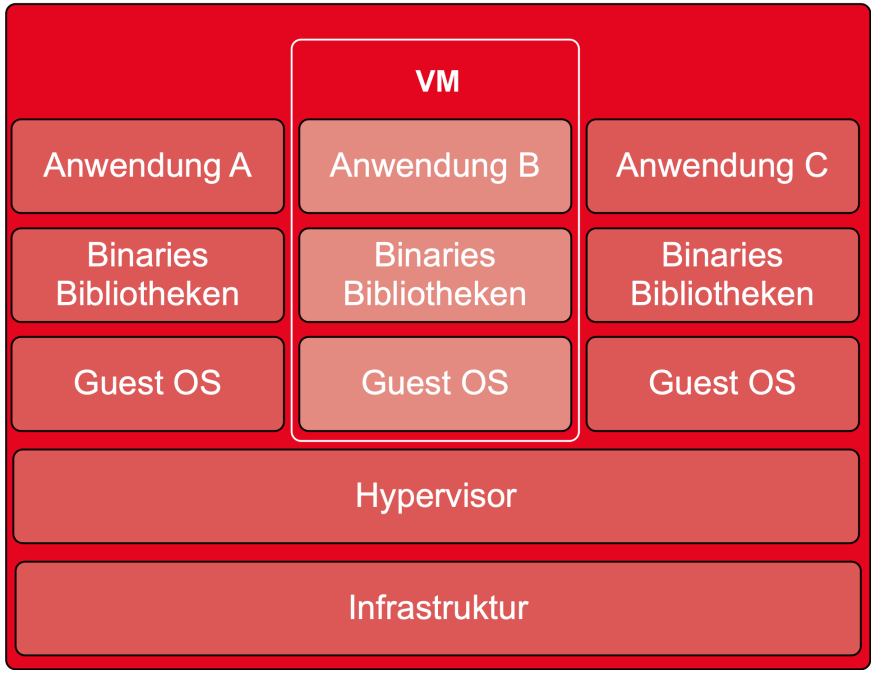
trivadis

Docker Images, Container und Volumes

Virtualisierung

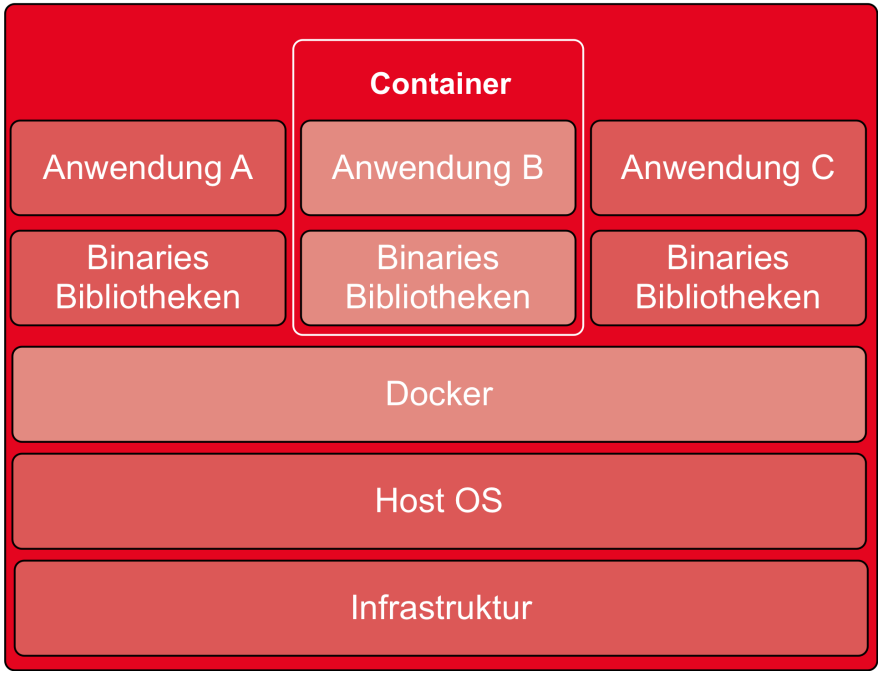
trivadis

- Nutzung gemeinsamen Infrastruktur
- jede VM ist ein "Server"
  - Gastbetriebssystem
  - Software und Bibliotheken
  - Anwendungen
- Redundanz
- Aufwand für die Einrichtung der VMs



Docker Container

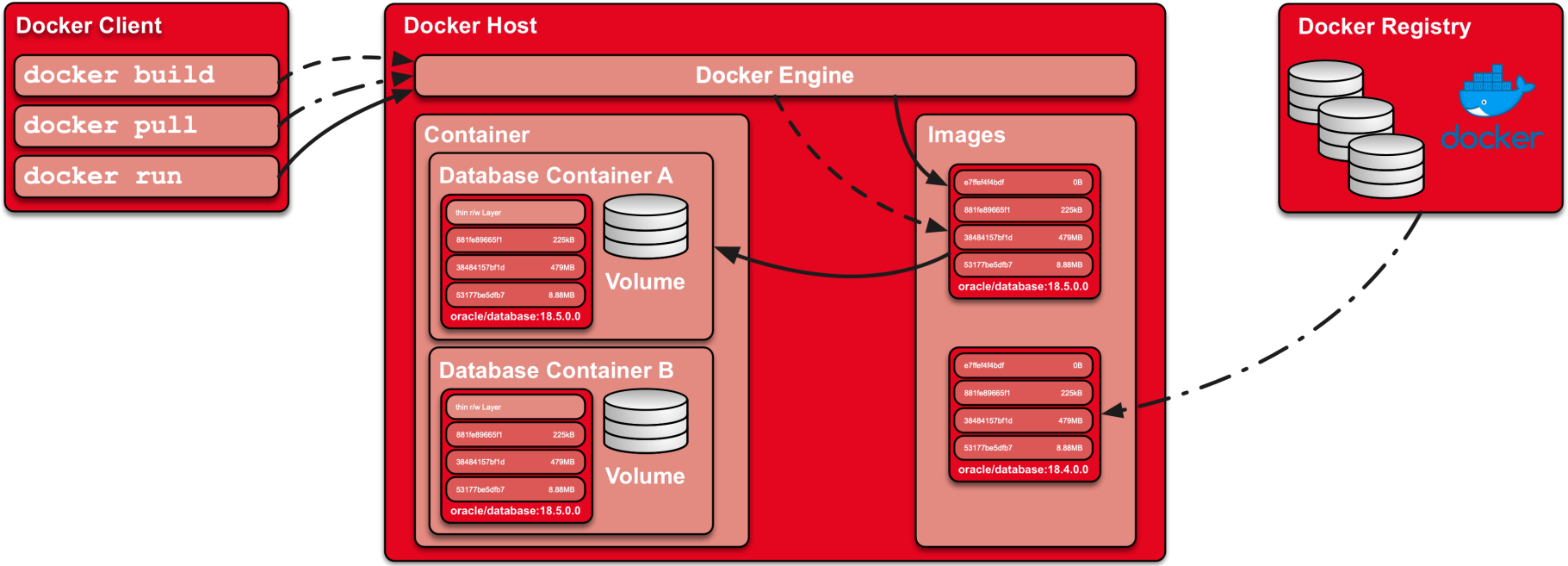
trivadis



- Nutzung gemeinsamen Infrastruktur
- ein Host-Betriebssystem
- Anwendungen sind "verpackt"
  - Nur notwendige Bibliotheken und Softwarekomponenten
- Verwendung vorhandener Images
- Schlank
- Reproduzierbar
  - Weil automatisch erstellt
- Docker Image sind **unveränderlich**

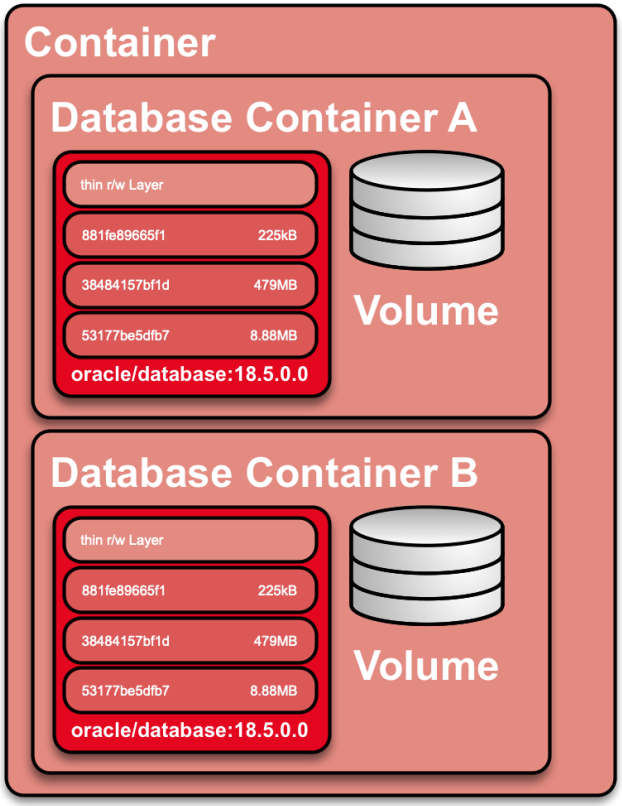
Docker Komponenten

trivadis



Volumes

- Docker Image sind **unveränderlich**
  - Änderungen nur in einem neuen Image
- Ein Container basiert auf einem Image
  - Top Layer read/write
  - Keine Persistenz über Lebensdauer des Containers
- Sicherstelen der Datenpersistenz durch Volumes
  - Daten liegen ausserhalb vom Container
- Lifecycle…
  - Neues Image
  - Neuer Container
  - aktuelles Volume



trivadis

Datenbank Image

Voraussetzungen

trivadis

- Docker Umgebung z.B. Docker Desktop für Mac OS
- Bereitstellen der Oracle Software
  - Oracle Database Enterprise Edition 18c (18.3.0.0)
  - Oracle Database Release Update 18.6.0.0.0 (Patch 29301631)
  - Oracle OJVM Release Update 18.6.0.0.190416 (Patch 29249584)
  - OPatch 12.2.0.1.17 for DB 18.x releases (APR 2019) (Patch 6880880)
- Bereitstellen des Docker Basis Images

```
docker pull oraclelinux:7-slim
```

- Bereitstellen der Docker Build Skripte z.B. https://github.com/oehrlis/docker

```
git clone https://github.com/oehrlis/docker
```

Image erstellen

trivadis

- Docker Build mit dem Kommando **`docker build`** starten

```
cd docker/OracleDatabase/18.6.0.0
docker build -t oracle/database:18.6.0.0 .
```

- Dockerfile nutzt Multi-Stage Build
  - Verfügbar ab Docker 17.05
  - Verwendung von mehreren FROM Anweisungen
  - Aufteilung des Build Prozesses in mehrere Abschnitte
  - Reduktion der Image Grösse
- Software ist teil vom Build Kontext und wird kopiert
  - Optional herunterladen der Software von einem lokalen Web Server

Dockerfile

trivadis

- Base Image
  - Setzen der Umgebung
  - OS Konfiguration des Basis Images
- Build Image
  - Kopieren der Oracle Binaries und Patch Files
  - Installation der Oracle Software
  - Installation der Oracle Patch, Trivadis Basenv™
- Ziel Image
  - Kopieren der Software vom Build Image
  - Abschluss der Installation (root.sh Skripte)
  - Definition von Ports, Volume sowie Start Skript

Dockerfile

```
FROM oraclelinux:7-slim AS base
...
```

```
FROM base AS builder
...
```

```
FROM base
...
```

Image Build Skripte

trivadis

Dockerfile nutzt verschieden Skripte für die Konfiguration des Images

- **00_setup_oradba_init.sh** Installation der aktuellsten OraDBA Init Skripte
- **01_setup_os_db.sh** OS Setup Konfiguration
  - Benutzer, Gruppen erstellen
  - YUM Software Pakete installieren
- **10_setup_db_18.6.sh** Installation der Oracle Binaries
  - Oracle Basis Release
  - Release Updates und Oracle JVM Update
- **20_setup_basenv.sh** Installation von Trivadis BasEnv™
- **5n_xxxxx_database.sh** Verschiedene Skripte zur Konfiguration des Containers

trivadis

Datenbank Container

Oracle Datenbank Container

trivadis

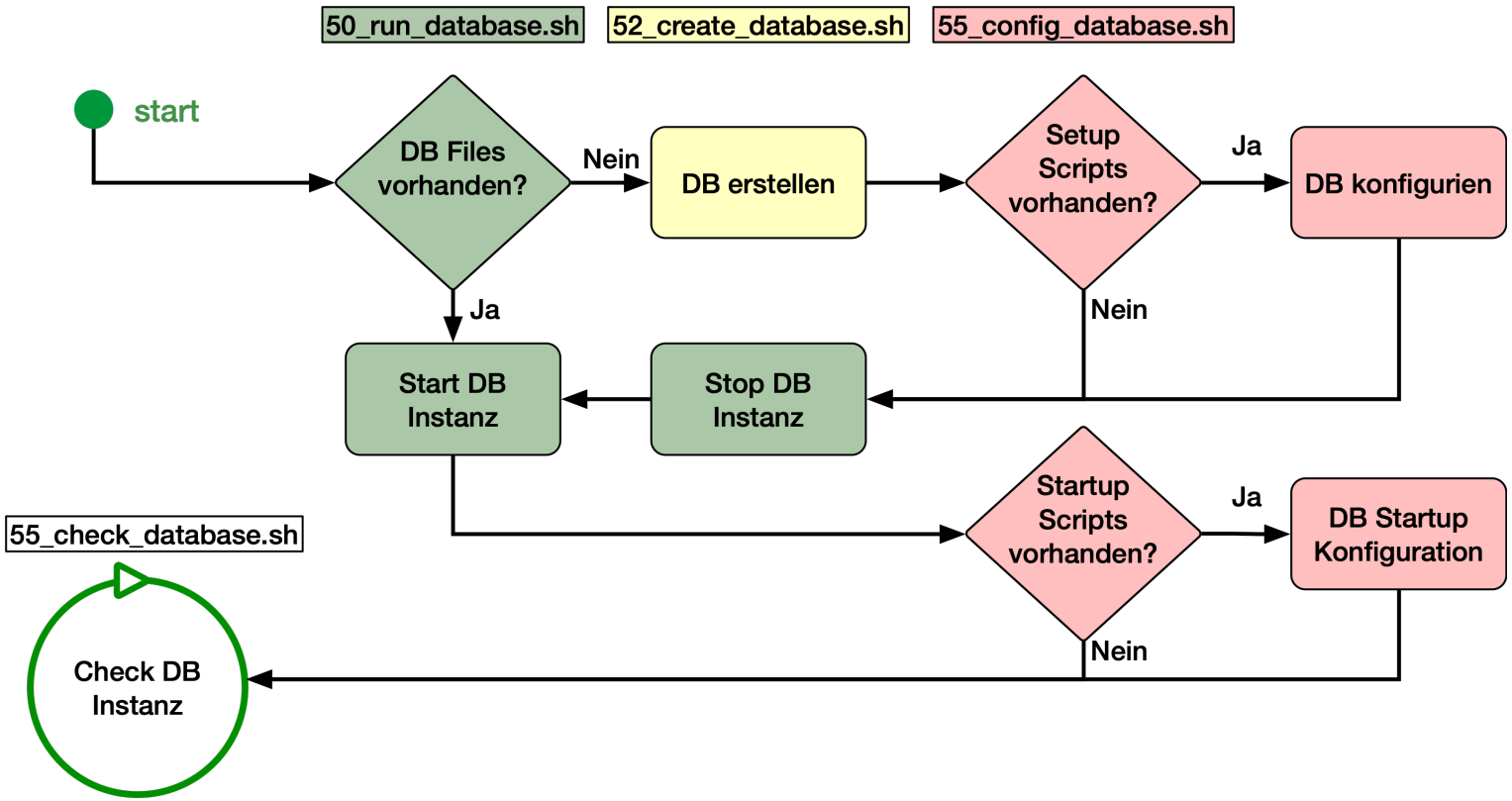
- Starten bzw. Instanziieren eines Oracle Datenbank Containers mit **`docker run`**
  - festlegen des Host (**`--hostname`**) und Container Namens (**`--name`**)
  - Volume für die Datenbank Files
  - Oracle Instanz Name als Umgebungsvariable **`ORACLE_SID`**
  - Ausführen des Conainer Kommandos **`CMD`**

```
docker run --detach --hostname tdb186s --name tdb186s \
  --volume /data/docker/volumes/tdb186s:/u01 \
  -p 1521:1521 -p 5500:5500 -e ORACLE_SID=TDB186S \
  oracle/database:18.6.0.0
```

- Kontrolle des Container Logs und Status der DB mit **`docker logs -f tdb186s`**

Ablauf beim Container Start

trivadis



Container Scripte

trivadis

- **50_run_database.sh** prüft das Volumen und startet die Datenbank mit…
  - **50_start_database.sh** Ist keine die Datenbank vorhanden, wird das Skrip
  - **52_create_database.sh** aufgerufen
- **50_start_database.sh** startet den Listener und die Datenbank Instanz
- **52_create_database.sh** erstellt mit dem dbca eine Datenbank. Basis Parameter lassen sich mit Umgebungsvariablen anpassen.
- **55_config_database.sh** prüft, ob Konfigurationsdateien (*.sh oder *.sql) im Verzeichnis `${INSTANCE_INIT}/setup` respektive `${INSTANCE_INIT}/startup` vorhanden sind.
- **55_check_database.sh** wird für den Healthcheck des Docker Containers genutzt und prüft den Status der Datenbank Instanz.

Container Variablen

trivadis

Variable	Verwendung
ORACLE_SID	Oracle SID respektive Datenbank Name. Standardwert ist TDB186S
CONTAINER	Flag für das Erstellen einer Oracle Container Datenbank Standardwert ist FALSE
ORACLE_PDB	Oracle PDB Name. Standardwert PDB1
ORACLE CHARACTERSET	Oracle Zeichensatz. Standardwert AL32UTF8
ORACLE_PWD	Password für den SYS Benutzer. Standard Passwort wird generiert und im admin Verzeichnis abgelegt
INSTANCE_INIT	Verzeichnis für die Instanz Konfigurationsdateien
ORADBA_RSP	Diverse Variablen für die Anpassung des dbca Templates

Abschluss initialer Container Start

trivadis

- Beim ersten Start wird durch **50_run_database.sh** eine Datenbank angelegt.
- Kontrolle der Ausgabe von **50_run_database.sh** mit `docker logs tdb186s`

```
SQL> Disconnected from Oracle Database 18c Enterprise Edition Release 18.0.0.0.0 - Production
Version 18.6.0.0.0
/u00/app/oracle
/u00/app/oracle
The Oracle base remains unchanged with value /u00/app/oracle
-----------------------------------------------------------------
 - DATABASE TDB186S IS READY TO USE!
-----------------------------------------------------------------
-----------------------------------------------------------------
   Tail output of alert log from TDB186S:
-----------------------------------------------------------------
The following output is now a tail of the alert.log:
Starting background process AQPC
2019-05-21T08:39:24.818101+00:00
AQPC started with pid=45, OS id=3873
2019-05-21T08:39:26.200725+00:00
Starting background process CJQ0
Completed: ALTER DATABASE OPEN
```

Zugriff auf den Datenbank Container

trivadis

- Zugriff für die Anwendungen via exportierte Ports z.B. 1521
  - Unterschiedliche ob auf Windows, MacOS und Linux
  - Docker läuft nur auf Linux “native”
- Zugriff via Kommandozeile mit **`docker exec`** und sqlplus, bash etc.

```
soe@gaia:~/docker/OracleDatabase/18.6.0.0/ [ic12201] docker exec -it tdb186s sqlplus / as sysdba

SQL*Plus: Release 18.0.0.0.0 - Production on Tue May 21 08:23:56 2019
Version 18.6.0.0.0

Copyright (c) 1982, 2018, Oracle.  All rights reserved.


Connected to:
Oracle Database 18c Enterprise Edition Release 18.0.0.0.0 - Production
Version 18.6.0.0.0

SQL>
```

Zugriff auf den Datenbank Container

trivadis

- Zugriff für die Anwendungen via exportierte Ports z.B. 1521
  - Unterschiedliche ob auf Windows, MacOS und Linux
  - Docker läuft nur auf Linux “native”
- Zugriff via Kommandozeile mit **`docker exec`** und sqlplus, bash etc.

```
soe@gaia:~/docker/OracleDatabase/18.6.0.0/ [ic12201] docker exec -it tdb186s bash --login

TYPE (Cluster|DG) : SID/PROCESS  STATUS      HOME      [2019-05-21 08:37:25]
-------------------------------------------------------------------------------
Dummy rdbms_ee     : rdbms18600   n/a         /u00/app/oracle/product/18.0.0.0

DB-instance (N|N)  : TDB186S      open        /u00/app/oracle/product/18.0.0.0

Listener           : LISTENER     up          /u00/app/oracle/product/18.0.0.0

oracle@tdb186s:/u00/app/oracle/ [rdbms18600]
```

trivadis

Anwendungsfälle

Anwendungsfälle

trivadis

- Oracle Datenbanken in Container auch im Microservice Umfeld nutzbar
- Entsprechende Build und Konfigurationsskripte müssen individuell erarbeitet werden
- Real Application Cluster (RAC) für Test und Entwicklung offiziell unterstützt
  - Oracle Beispiel auf GitHub https://github.com/oracle/docker-images
- Nutzung von Oracle Container
  - Docker Container enthält single-PDB
  - PDB kann relativ einfach plugged/unplugged werden
- Verwenden von Docker-Compose
  - Keine manuellen Docker Kommandos
  - Container verlinken
  - Aufbau eines privaten Netzwerks

Docker Compose

trivadis

- Definieren aller Parameter, Volumes, Netzwerke usw. in einer YAML-Datei.
  - Der Standardname ist **docker-compose.yml**
  - Kann mehrere Container enthalten
- Wird über den Befehl **docker-compose** oder den regulären **docker** Befehl verwaltet.
- Starten eines Containers im Hintergrund

```
docker-compose up -d
```

- Abschaltung entweder durch stop oder down
  - **down** unten entfernt den Container
  - **stop** stoppt einfach den Dienst / Container

```
docker-compose down
```

Docker Compose Datei

trivadis

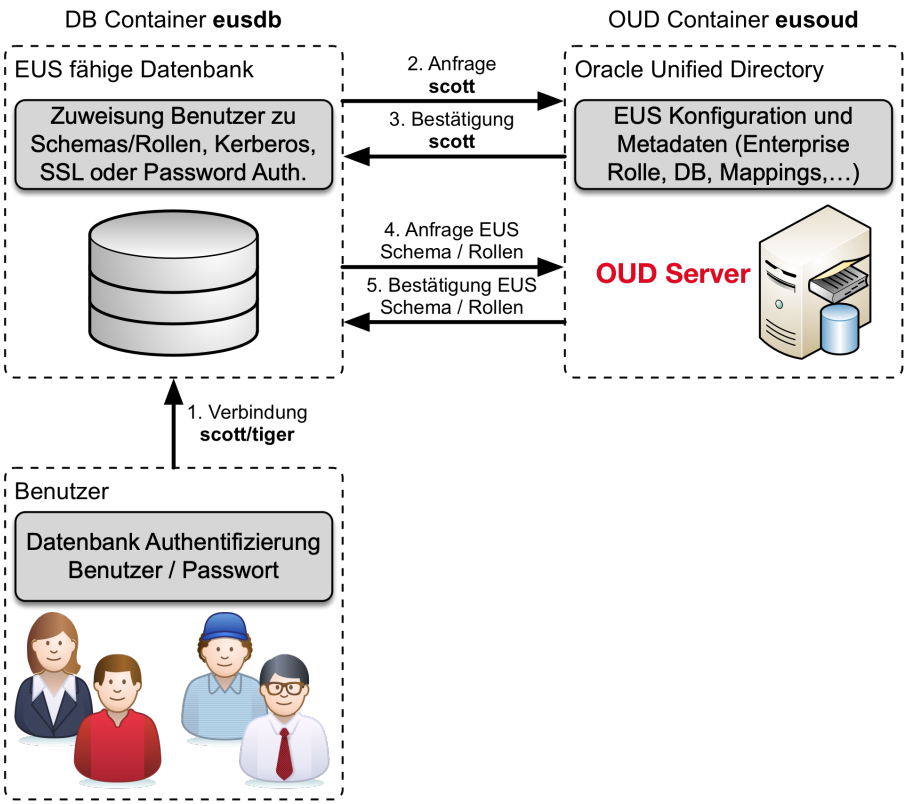
- Beispiel **docker-compose.yml** Datei für tdb186s
  - YAML, achtet auf die richtigen Leerzeichen!

```
tdb186s:
  image: ${DOCKER_USER}/${DOCKER_REPO}:18.6.0.0
  container_name: tdb186s
  hostname: tdb186s
  restart: unless-stopped
  network_mode: bridge
  volumes:
    - ${DOCKER_VOLUME_BASE}/tdb186s:/u01
    - ./config:/u01/config
  ports:
    - "1521"
  environment:
    CONTAINER: 'FALSE’
    INSTANCE_INIT: /u01/config
    ORACLE_SID: TDB186S
```

Oracle Enterprise User Security

trivadis

- Oracle Datenbank Container
  - Demo Schema mit VPD
- Oracle Unified Directory Container
  - Directory mit EUS Suffix
- Initialen Start der Container
  - Erstellen der DB / Directory
  - Registrierung der DB
  - Konfiguration von EUS
- Verfügbar auf GitHub
  - https://github.com/oehrlis/docker/tree/master/samples/eus



Oracle DB und Docker Best Practice

trivadis

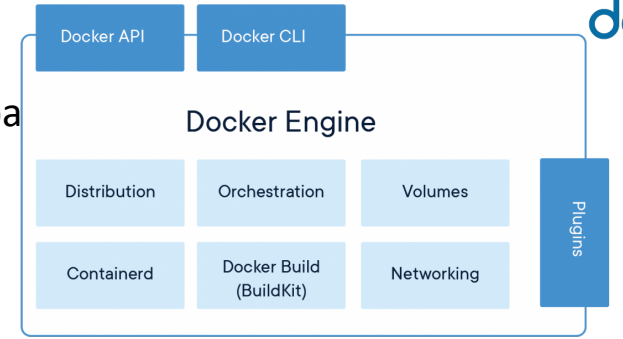
- Immer die aktuellste Docker Version verwenden mindestens Docker CE 17.03
  - **`btrfs`** ist der empfohlene Storage Treiber alternative **`overlay2`**
- Aktuellste Images und Build Dateien verwenden
- Daten Container oder Volumes nutzen um Persistenz sicher zu stellen
- Container sind keine VM's
  - Patching, HA, B&R und Security funktioniert anders
  - Grundsätzlich besteht kein Grund für ein Login in den Container

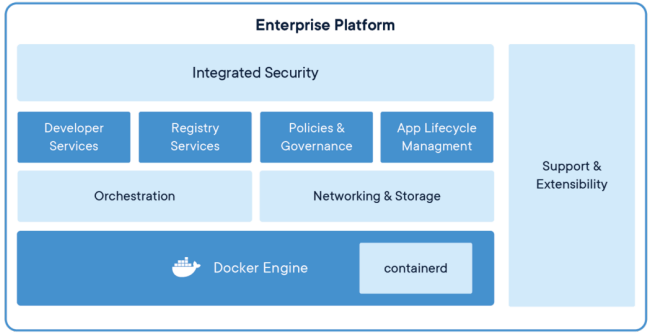
trivadis

Lizenzierung

Lizenzierung - Docker

- Docker Desktop
  - Docker Anwendung für Mac- oder Windows
  - Als Community und neu Enterprise Version verfügba
- Docker Engine
  - Verfügbar für verschiedene Linux Distributionen
  - Community und Enterprise Version verfügbar
- Docker Enterprise
  - Komplette Containerplattform basierend auf Docker Engine
  - Zusätzliche Komponenten wie Support, Trusted Registry, Orchestrierung, Security etc.





Quelle: https://www.docker.com/products

Lizenzierung - Oracle

trivadis

Oracle Software

- Docker ermöglicht die Limitierung von Ressourcen wie CPU, Memory, etc.
- Limitierung **nicht** anwendbar im Kontext der Oracle Lizenzierung!
- CPU Cores der Docker Hosts bestimmen die benötigten Lizenzen.
- Analoge Herausforderung wie bei der Virtualisierung.

Mögliche Lösungsansätze

- Verwendung von Oracle Express Edition 18c im Docker Umfeld
- Oracle ULA (Unlimited License Agreement)
- Aufbau einer dedizierten Docker Infrastruktur für Oracle Software

Quelle: Oracle Inc.

trivadis

Zusammenfassung

Zusammenfassung

trivadis

- Oracle Datenbanken können einfach und schnell in Container aufgebaut werden.
- Docker-basierte Datenbanken sind nicht für eine hohe io-Performance geeignet.
- Durch die Verwendung von Docker Volumes ist die Datenpersistenz sichergestellt.
- Erarbeitung von klaren Anwendungsfällen und passenden Architektur ist Voraussetzung.
- Lizenzierung bleibt wie bei der Virtualisierung eine der grossen Herausforderungen.

Question and answers...

Stefan Oehrli

Solution Manager / Trivadis Partner

Tel.: +41 58 459 55 55

stefan.oehrli@trivadis.com

@stefanoehrli www.oradba.ch

https://url.oradba.ch/SOUG1905

Eine WELT ermöglichen,
in der intelligente IT
LEBEN und ARBEITEN
völlig selbstverständlich
erleichtert.