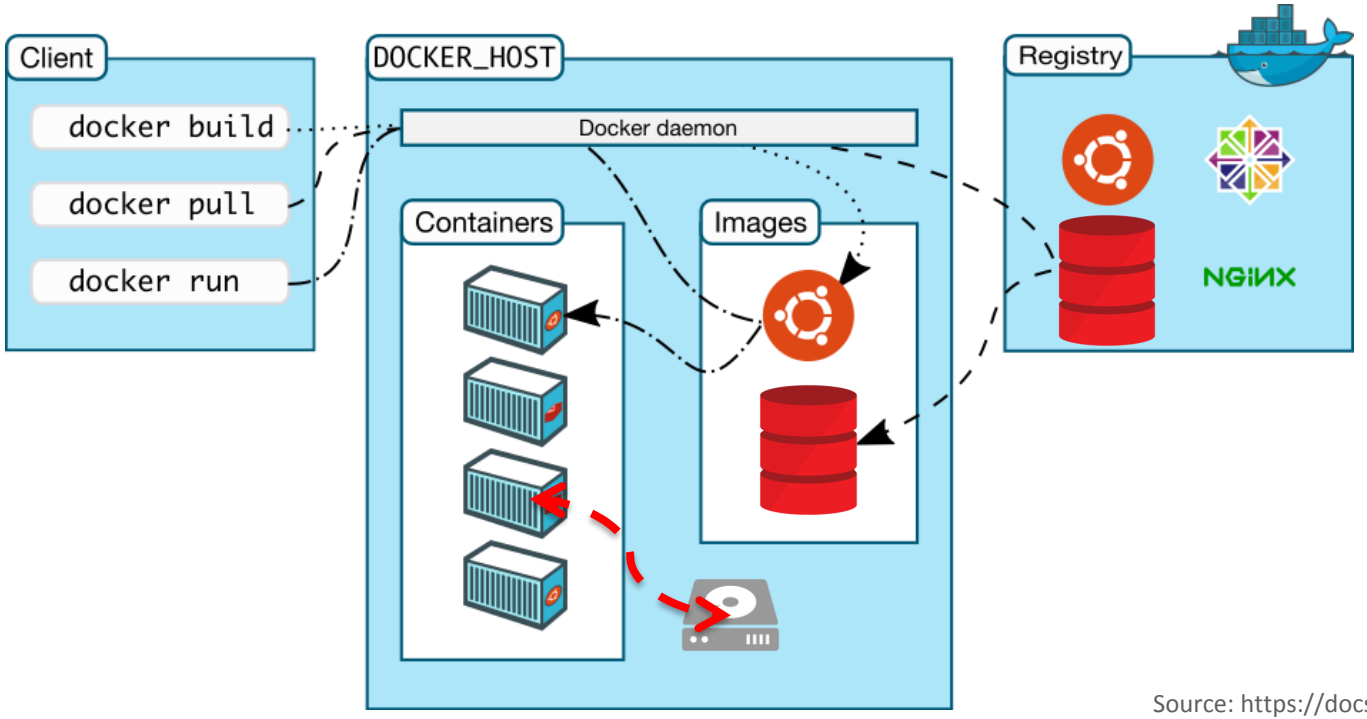
Best practices for running Oracle Database and Weblogic Server on Docker

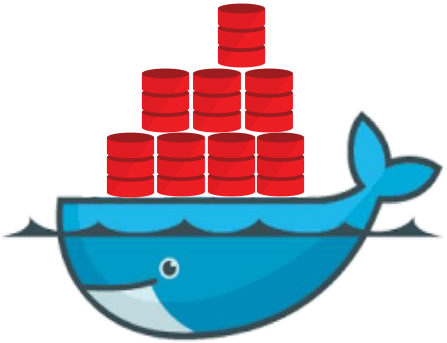
ORACLE®

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Docker Key Components

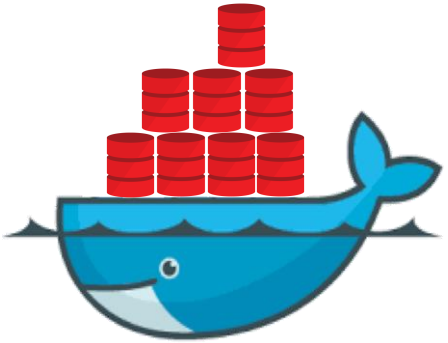




- Registry
- Images
- Containers
- Docker daemon/engine

Source: https://docs.docker.com/engine/docker-overview/

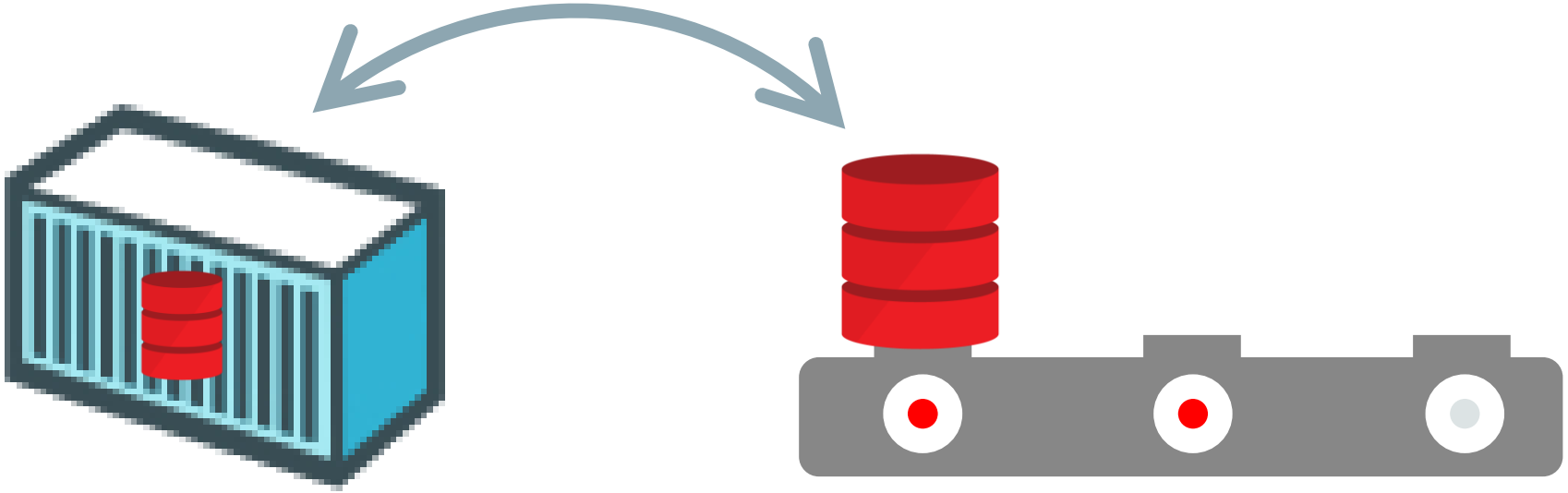
Oracle Database on Docker



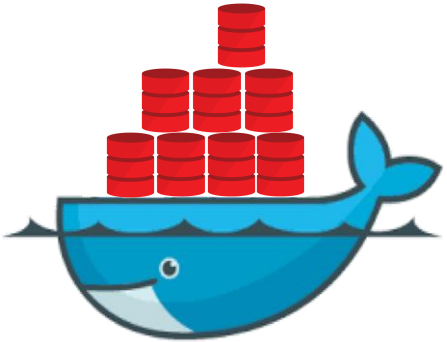
- Oracle Database is fully supported on Docker
  - Oracle Linux 7 - UEK 4
  - Red Hat Enterprise Linux 7
- Oracle images on Oracle Container Registry & Docker Store
- Docker build files on GitHub
- RAC is **not supported**

Oracle Database on Docker

- Docker container contains single-PDB CDB (**no MTO license required**)
- PDB can be plugged, unplugged, etc.
- PDB can move bi-directional



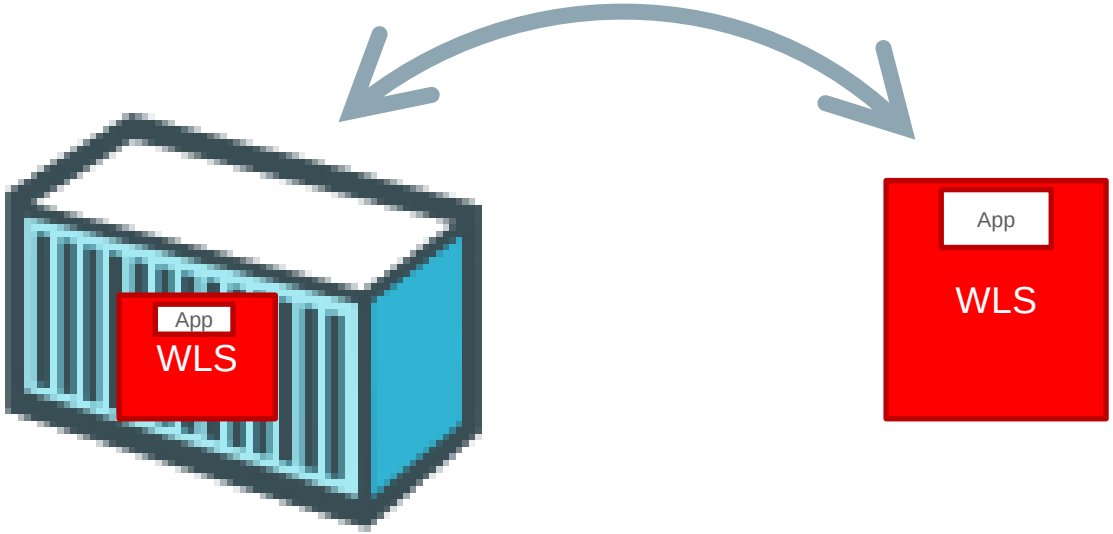
Oracle Weblogic Server on Docker



- Oracle WebLogic Server is fully supported on Docker
  - Oracle Linux 7 - UEK 4
  - Red Hat Enterprise Linux 7
  - Support Statement Doc ID 2017945.1
- Oracle images on Oracle Container Registry & Docker Store
- Docker build files on GitHub

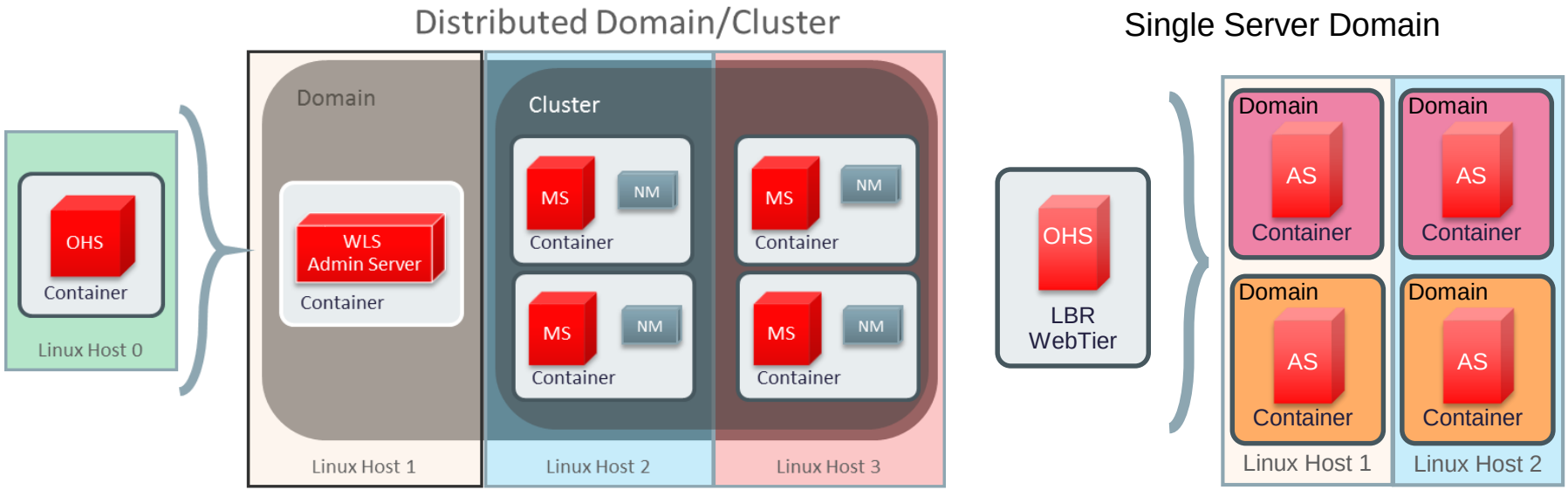
WebLogic Server on Docker

- Docker container contains single WebLogic Server



WebLogic Server on Docker

- WebLogic Configurations in Docker



BEST PRACTICES

ORACLE®

Oracle DB and Docker best practices

- Use `btrfs` (Oracle recommended) or `overlay2` storage drivers
- Do **not** use `btrfs` for data files storage
- B-tree filesystem
- Copy-on-write functionality
- Transparent compression
- Really **not a good fit** for your redo and undo
- Also not really a good fit for data files either
- Github issue #647

Oracle DB and Docker best practices

- `/dev/shm` mounted with no exec rights (and only 64MB by default)
- No native compiled PL/SQL
- No Java stored procedures
- No Automatic Memory Management
- …
- `docker run -v /dev/shm --tmpfs /dev/shm:rw,nosuid,nodev,exec,size=1g ...`

Docker volumes

Oracle DB/Weblogic and Docker best practices

- Docker volumes contain data that should be persisted throughout a container's lifespan
- Examples are:
  - Configuration files
  - Log files
  - Database files
  - Domain home
  - Stores (JMS store and JTA Tlogs)
- Files inside a container are owned by the UID of the container's user.
  - UID: 54321 for the `oracle` Unix user
  - Can be mapped to another UID with named volumes

Oracle DB/Weblogic and Docker best practices

- Three ways to create Docker volumes:
  - Anonymous volumes (no name specified)
  - Named volumes (a volume referred to by name)
  - Host-only volume (a location on the host)
  - Third party drivers for shared volume (NFS)

Oracle DB/Weblogic and Docker best practices

Anonymous volumes

- Create or run a container with '-v /container/fs/path' in docker create or docker run
- Use the VOLUME instruction in Dockerfile: VOLUME ["/container/fs/path"]
- Are not easily reusable across containers (you know, anonymous!)
- Transparently copy data from the image into the volume first!

```
"Type": "volume",
  "Name":"7a27a94df5192a20ec6a07502be9a2163faaa150e0076505937578ce785d5376",
"Source":
"/var/lib/docker/volumes/7a27a94df5192a20ec6a07502be9a2163faaa150e0076505937
578ce785d5376/_data",
"Destination": "/opt/oracle/oradata"
```

Oracle DB/Weblogic and Docker best practices

Named volumes

- Use docker volume create --name volume_name
- Create or run a container with '-v volume_name:/container/fs/path' in docker create or docker run

```
[oracle@localhost ~]$ docker volume create GeraldVol1
GeraldVol1
[oracle@localhost ~]$ docker volume inspect GeraldVol1
"Mountpoint": "/var/lib/docker/volumes/GeraldVol1/_data",
"Name": "GeraldVol1",
```

- Are great for making volumes specific
- Transparently copy data from the image into the volume first!
- But…

Oracle DB/Weblogic and Docker best practices

Named volumes

- Hold data in `/var/lib/docker/volumes`
- Is usually owned by `root`
  - Do you really want to give a user root access?
- Contains **all** volumes of all containers
  - Do you really want to give a user access to all volumes?

Oracle DB/Weblogic and Docker best practices

Host-only volumes

- Are created by specifying a location on the host
  - `docker run –v /home/gerald/volumes:…`
- Offer great flexibility where exactly volume data is being put
- But…

Oracle DB/Weblogic and Docker best practices

Host-only volumes

- **Do not** transparently copy data from the image into volume first!
- Have a static absolute path
- You have to set permissions on the file system accordingly first!
- They are “host-only”, i.e not cluster aware (other than being put on a SAN)

Oracle DB/WebLogic and Docker best practices

Docker volumes drivers

- Docker 1.8 and later support a volume plugin which can extend Docker with new volume drivers.
- You can use volume plugins to mount remote folders in a shared storage server directly, such as iSCSI, NFS, or FC.
- The same storage can be accessed, in the same manner, from another container running in another host.
- Containers in different hosts can share the same data.

WebLogic Server and Docker best practices

Docker volumes

- Data volume containers are data-only containers. After a data volume container is created, it doesn't need to be started. Other containers can access the shared data using --volumes-from.

```
# step 1: create a data volume container 'vdata' with two anonymous volumes
$ docker create -v /vv/v1 -v /vv/v2 --name vdata weblogic-12.2.1.3-developer
 # step 2: run two containers c3 and c4 with reference to the data volume container vdata
$ docker run --name c3 --volumes-from vdata -d weblogic-12.2.1.3-developer 'sleep 3600' $ docker run --name c4 --volumes-from vdata -d weblogic-12.2.1.3-developer 'sleep 3600'
```