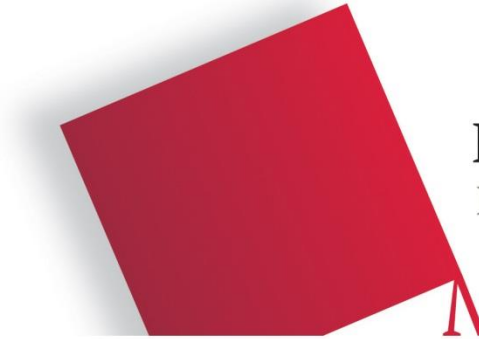


HOCHSCHULE
FÜR ANGEWANDTE
WISSENSCHAFTEN
MÜNCHEN

Containerisierung mit Docker

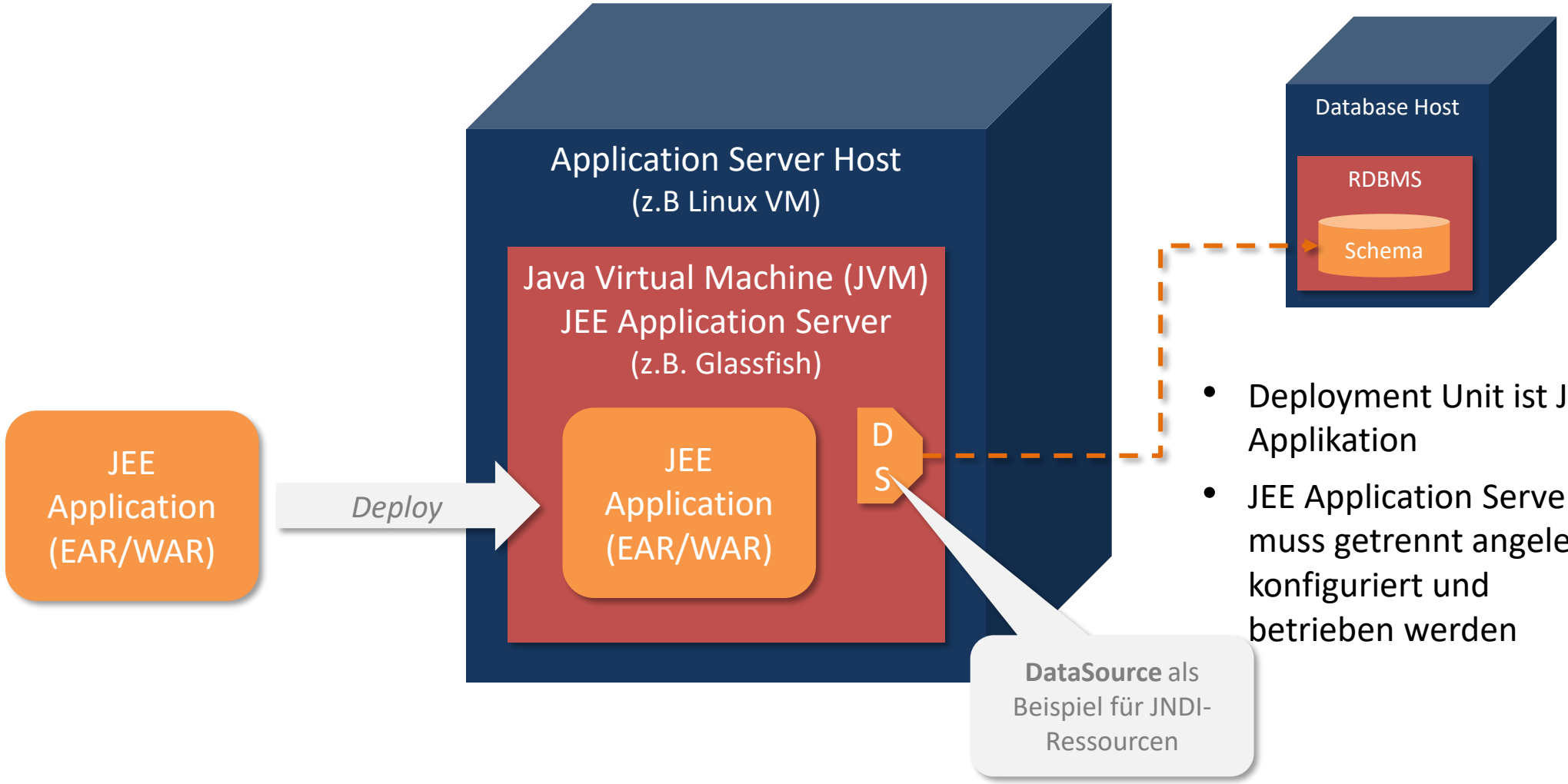
FWP Aktuelle Technologien zur Entwicklung verteilter Java-Anwendungen

Michael Theis, Lehrbeauftragter

Docker erleichtert das Bereitstellen und den Betrieb von Apps im Vergleich zum klassischen Betrieb

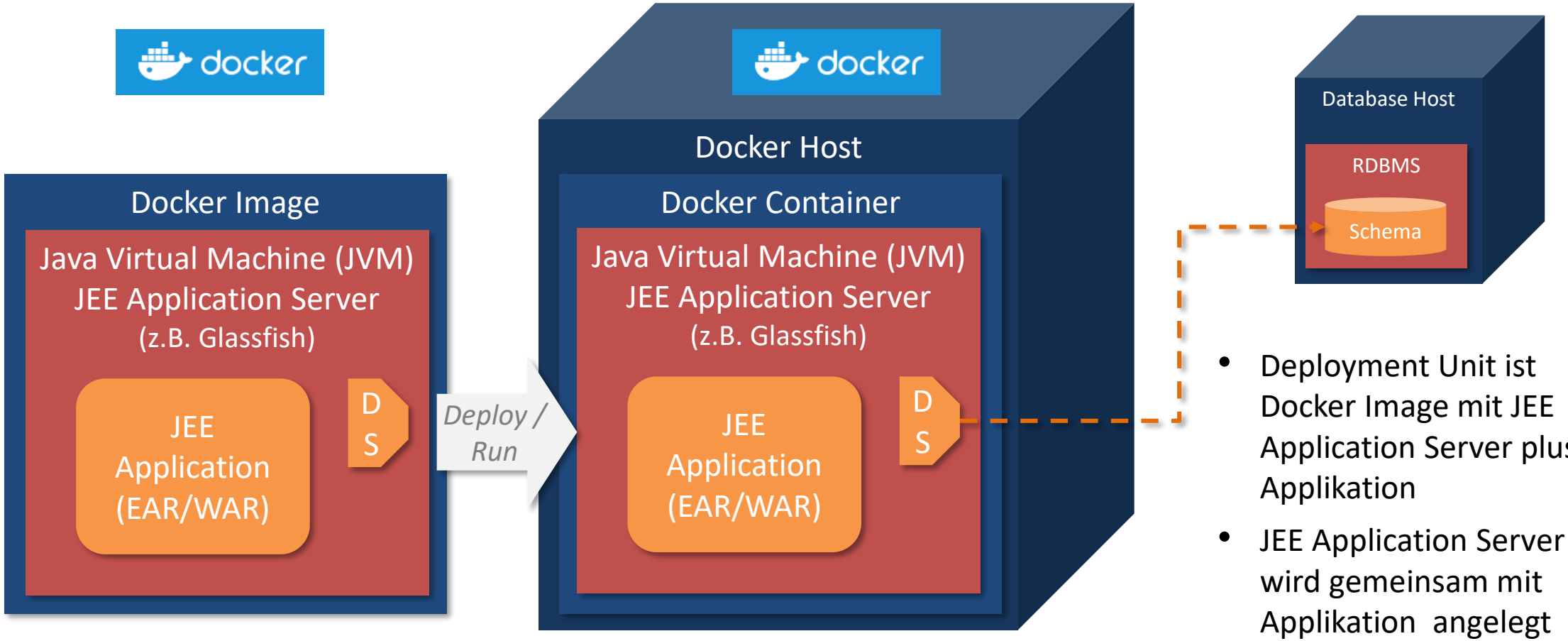
MOTIVATION

Klassischer Betrieb auf Bare Metal/IaaS



- Deployment Unit ist JEE Applikation
- JEE Application Server muss getrennt angelegt, konfiguriert und betrieben werden

Betrieb mit Docker Containern auf CaaS



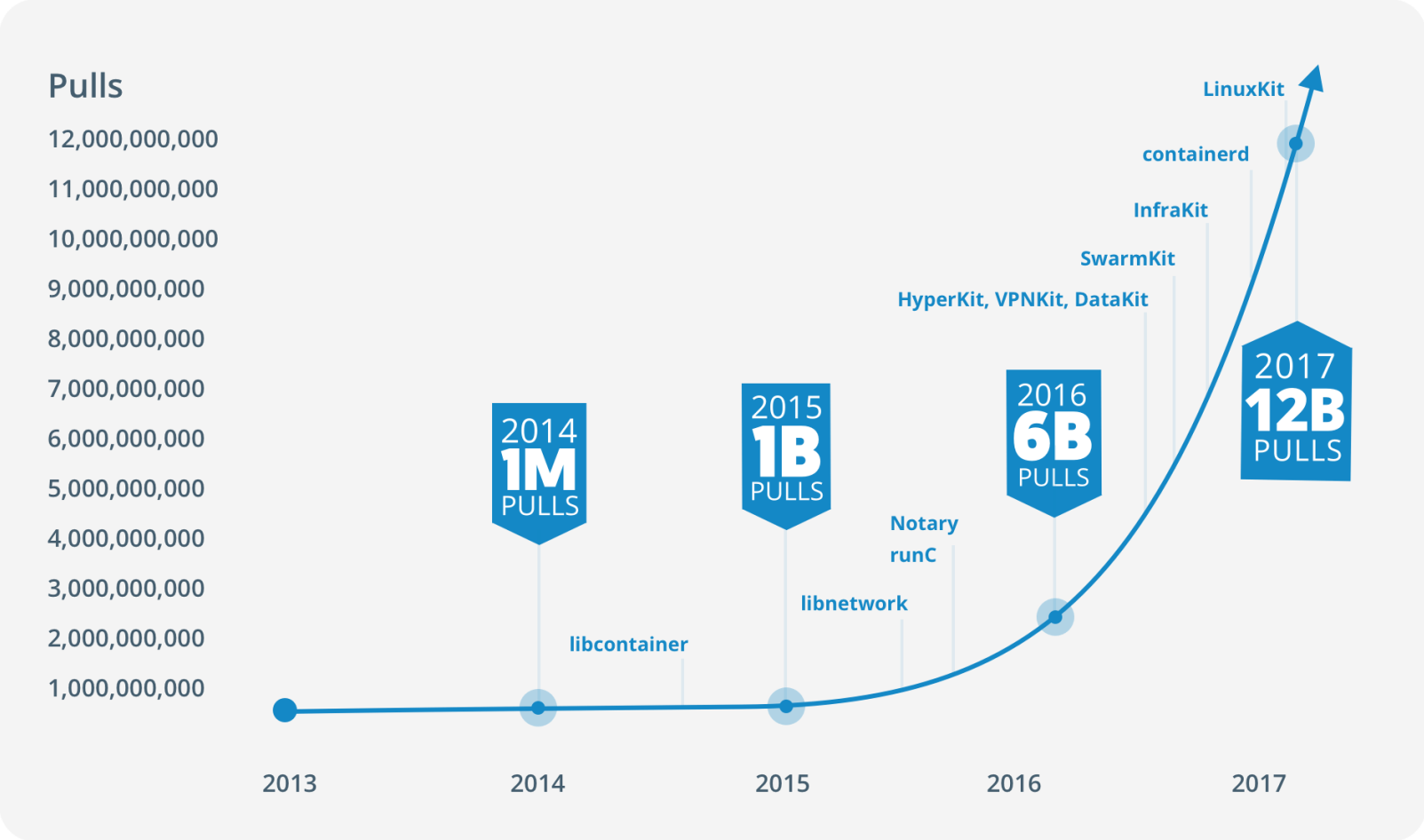
- Deployment Unit ist Docker Image mit JEE Application Server plus Applikation
- JEE Application Server wird gemeinsam mit Applikation angelegt und konfiguriert

Docker erleichtert den Betrieb

- Build erzeugt in sich vollständiges und unveränderliches Artefakt (Docker Image)
- Ein Artefakt (Docker-Image) wird auf allen Zielumgebungen bereitgestellt und über Umgebung konfiguriert
- Auf allen Zielumgebungen wird das gleiche Artefakt ausgeführt (Docker Container)
- Keine manuellen Eingriffe in das Artefakt mehr möglich => weniger Fehler beim Bereitstellen und Ausführen

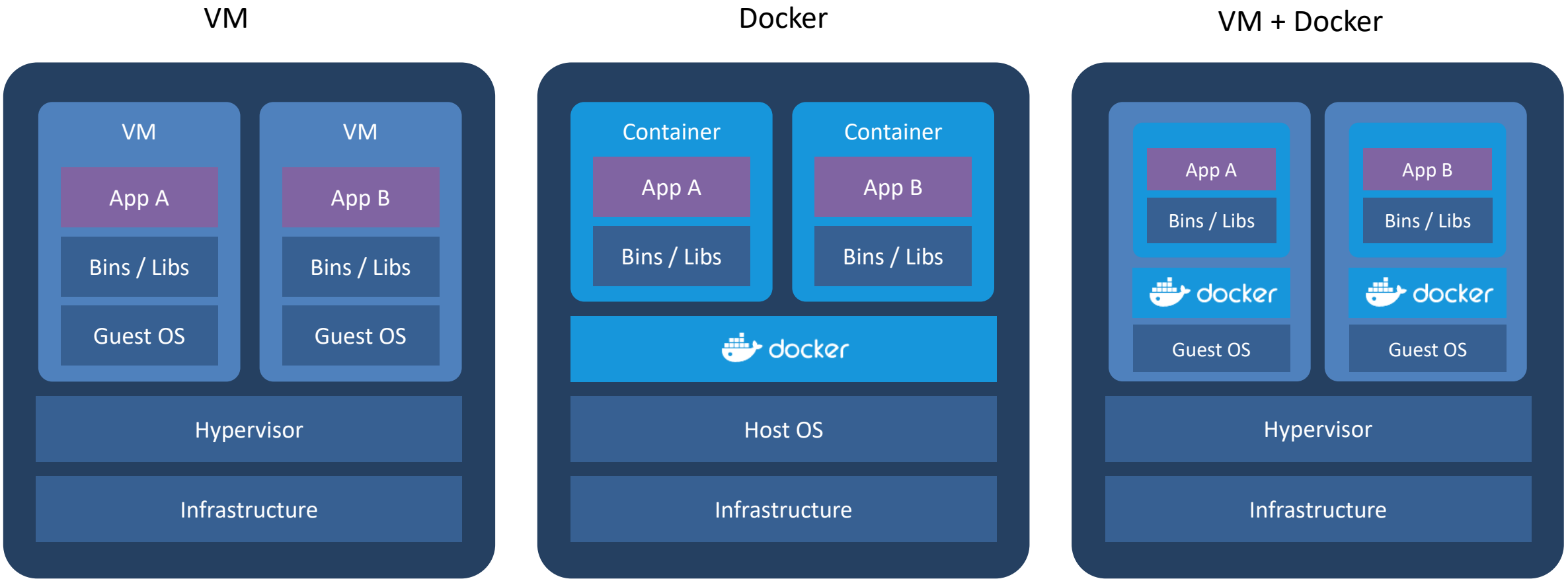
CONTAINERISIERUNG MIT DOCKER

Docker hat die Welt verändert

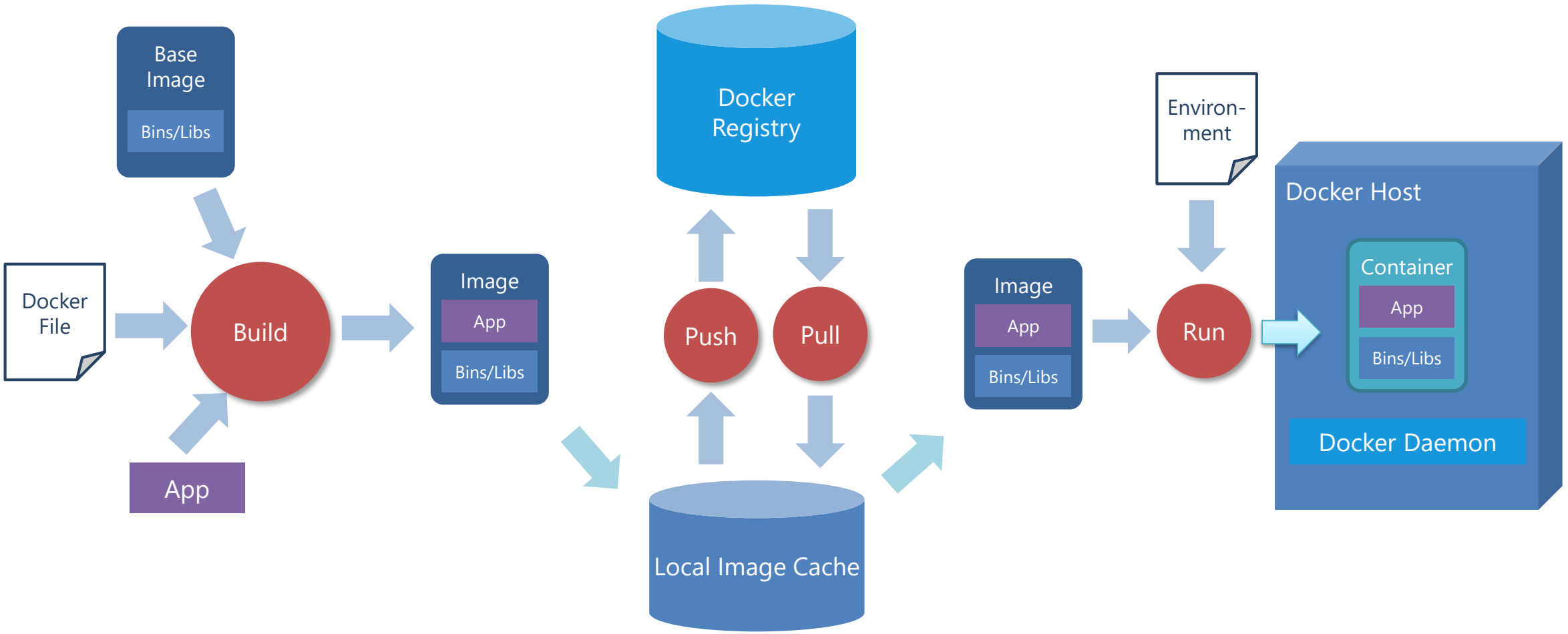


https://www.docker.com/what-container

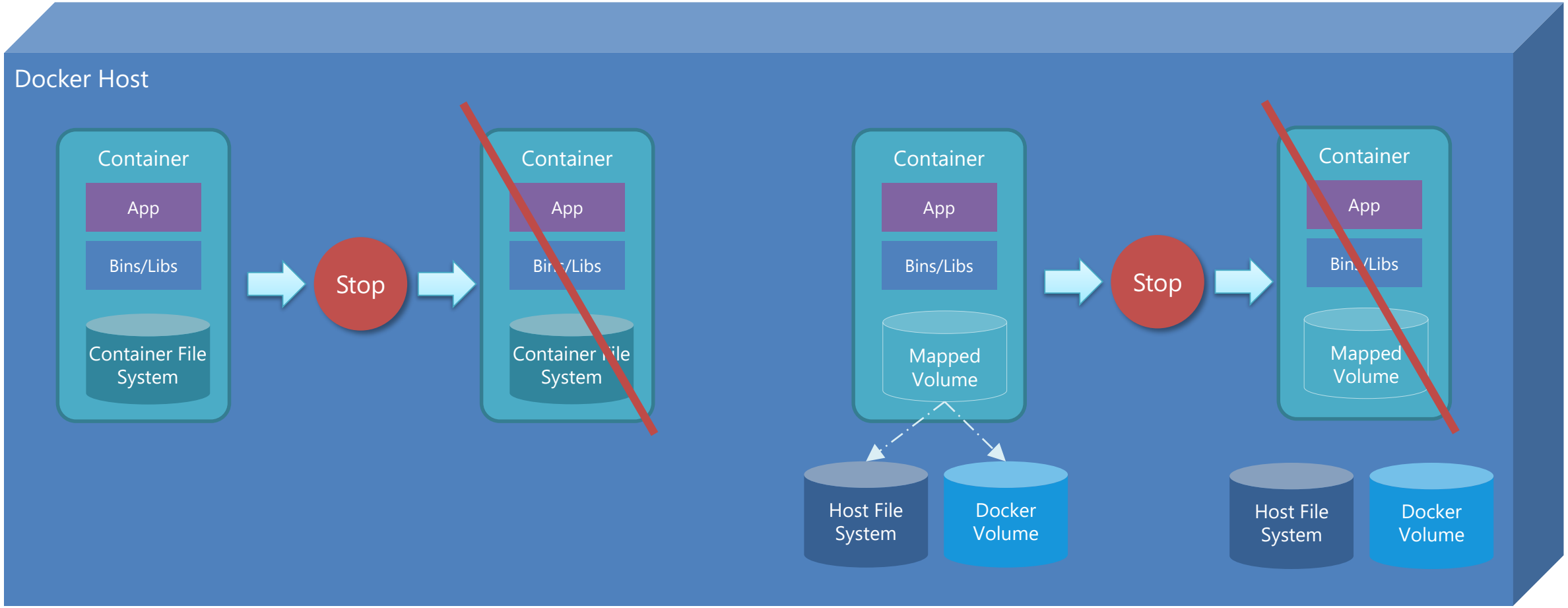
Docker vs VMs



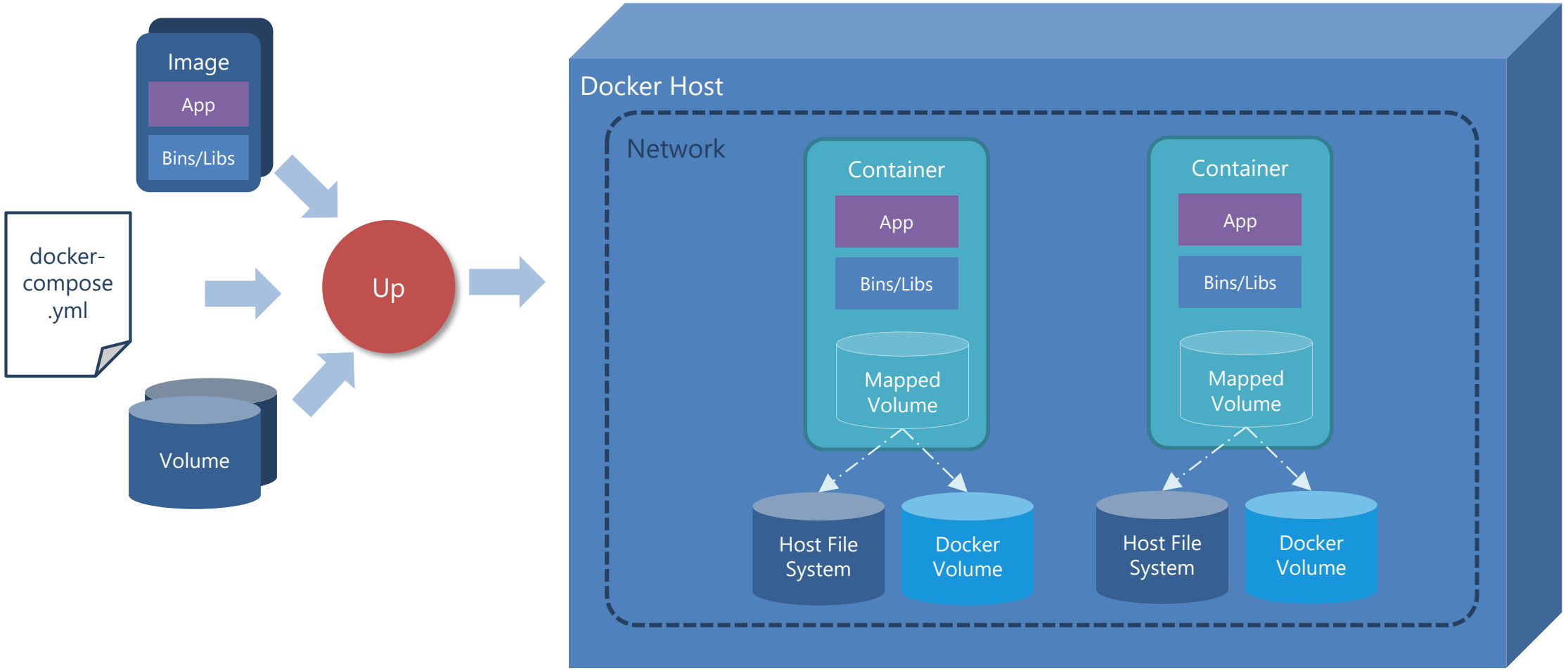
Docker Files, Images & Container



Docker Volumes



Docker Compose



Docker Fact Sheet

- Docker Images können vererbt werden (Inheritance)
- Docker Images sind immutable: Konfiguration bei Ausführung
- Benötigte Ressourcen (CPU, RAM) können beschränkt werden
- Virtuelles Netzwerk mappt interne Ports auf externe Ports
- Maven-Plugins für Steuerung von Docker vorhanden
- Unterstützung von Docker unter Windows verbesserungswürdig => auf Linux setzen
- Best Practices für Docker beachten!

Fragen?

ANHANG

Kontakt

Michael Theis
Lehrbeauftragter Hochschule München

email michael.theis@hm.edu
mobile + 49 170 5403805
web http://www.tschutschu.de/Lehrauftrag.html