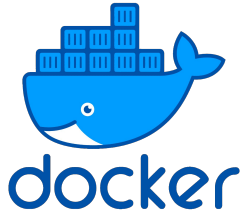
Docker & Alternativen

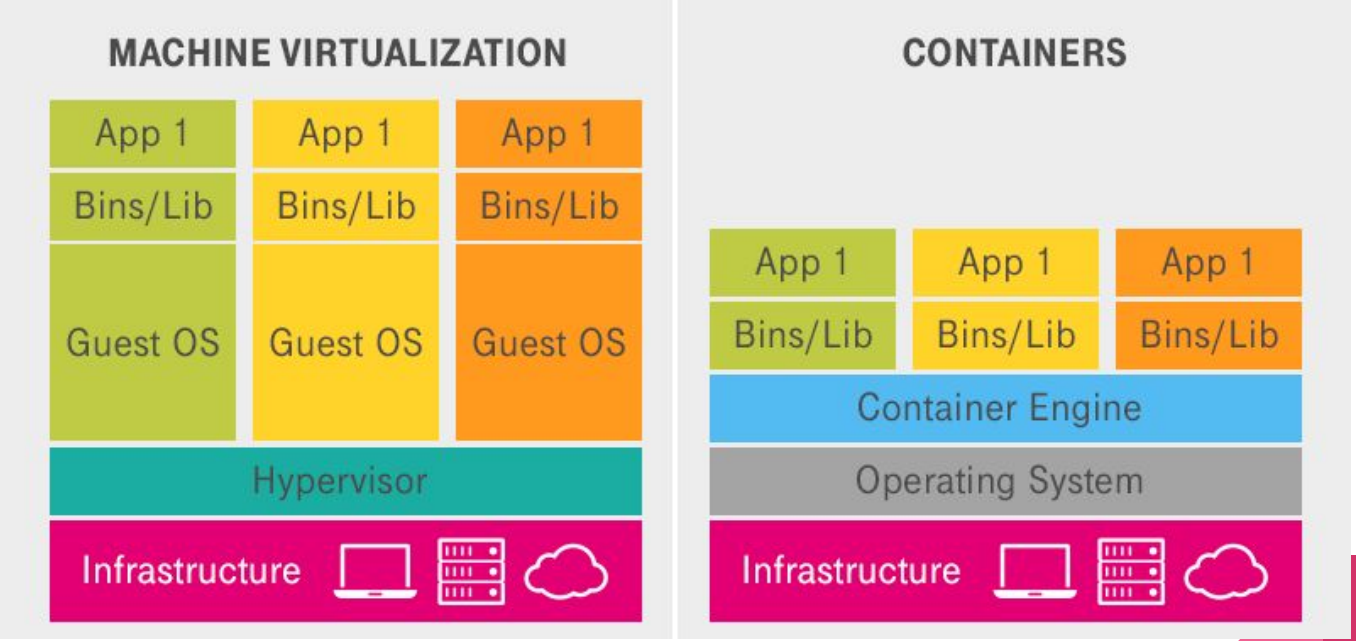
Niklas Aichinger

Docker

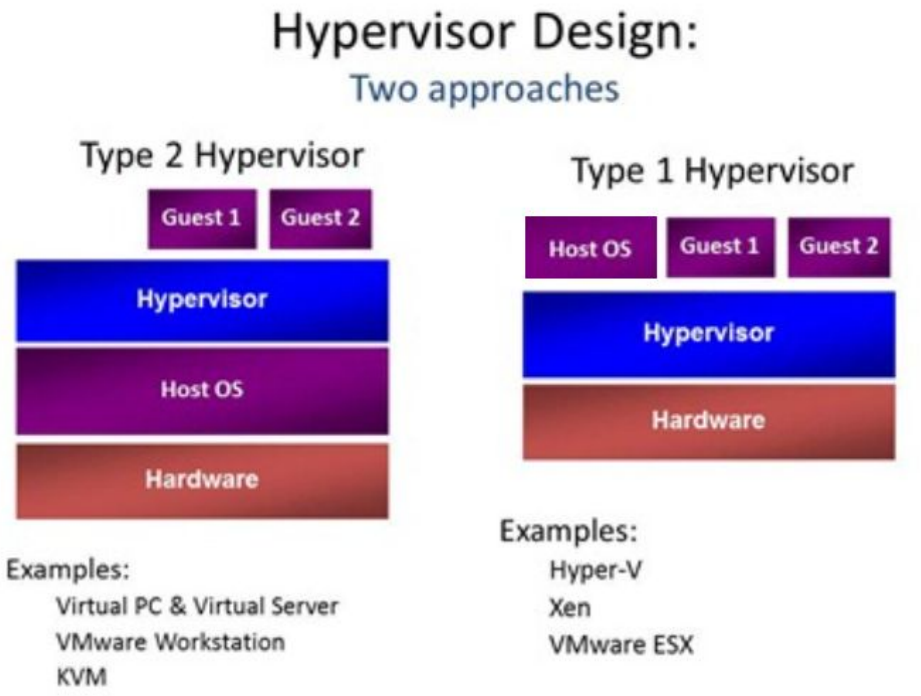
- Um eine Technologie und alle Abhängigkeiten in einen Container zu verpacken
- In Docker containern bleibt die Laufzeitumgebung unverändert auch auf anderen Host Systemen
- Leichtgewichtig Virtualisiert



schwergewichtige / leichtgewichtige Virtualisierung



Type 1 vs Type 2

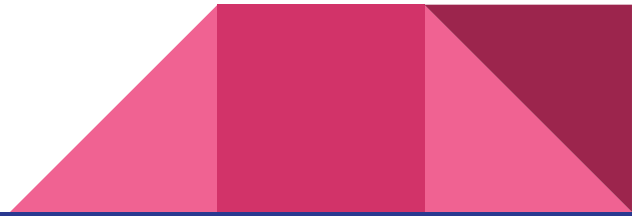


Docker Bestandteile

- **Dockerfile**: Anleitung wie der Container aufgebaut ist (Kochrezept)

- **Image**: Vorlage für einen Container welche in einer registry gespeichert werden

- **Container**: Ist eine ausführbare instanz eines Images welches eine Applikation und alle Abhängigkeiten enthält



Docker Alternative: Podman

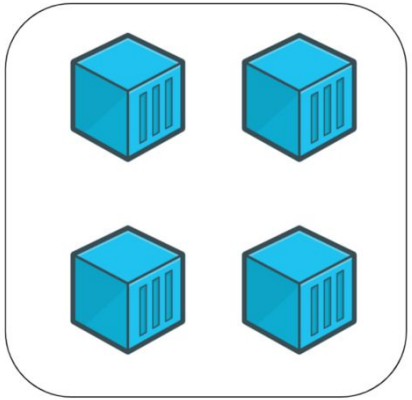
- Benötigt keinen Daemon um container zu starten (deshalb lightweight und sicherer)
- Besserer Support für gestartete Containers als non-root Users, führt zu besserer Security.
- In den meisten fällen kann man podman mit einem alias verwenden (“alias docker=podman”)

podman

Docker compose

- Ist ein Tool um multi-Container Anwendungen laufen zu lassen
- Mit YAML kann man diese Anwendungen konfigurieren
- Verwendet in: Development, Testing, Production

Docker

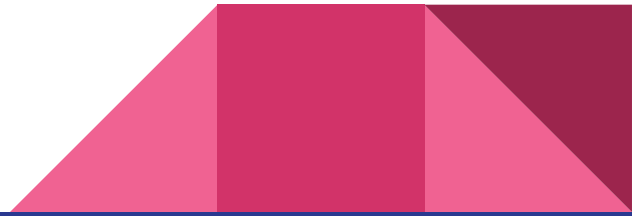


Docker-Compose

Container Orchestration

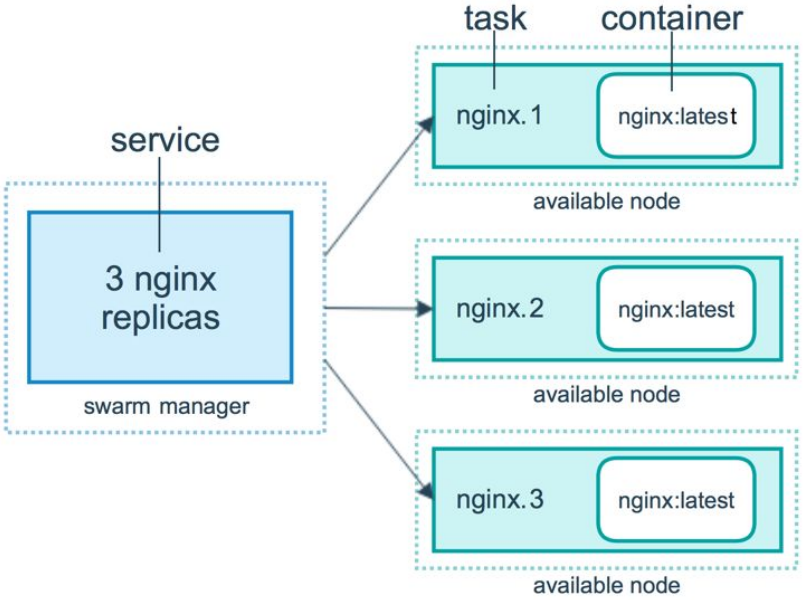
- Um so größer eine Applikation skaliert wird um so mehr benötigt man Tools welche das Deployment unterstützen

- Tools welche beim verwalten, skalieren und warten helfen nennt man Orchestrators



Docker Swarm

- Word verwendet um mehrere docker engines zu verwalten welche im swarm mode laufen
- Ein node ist eine Instanz einer docker engine welcher ein part des swarm ist
- Ein docker stack file ist ein docker compose file mit zusätzlichen Settings
  - (replica, deploy, ...)



Kubernetes

Verteilung von Container-Anwendungen in einem Cluster von Hosts.

Automatische Skalierung von Containern basierend auf Ressourcenbedarf.

Load-Balancing und Self-Healing von Anwendungen.

Fortgeschrittene Funktionen wie StatefulSets, DaemonSets und Services.