

Containerplattformen

Rancher 2.0 vs. OpenShift

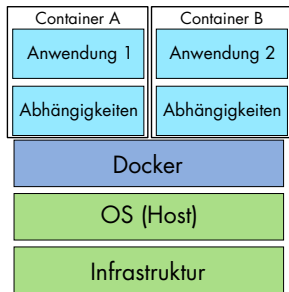


Dr. Jonas Trüstedt

30.1.2018

Virtualisierung in Containern:

- ▶ Minimalisierung der Abhängigkeiten
- ▶ Mit Host geteilter Kernel
 - Wenig Overhead
- ▶ 1 Container pro Funktion/Prozess (Microservices)
 - Anwendungen bestehen aus mehreren Containern



Limits für einzelnen Docker-Host:

- ▶ Anzahl der Container
- ▶ Lastverteilung (Skalierung)
- ▶ Automatisierung

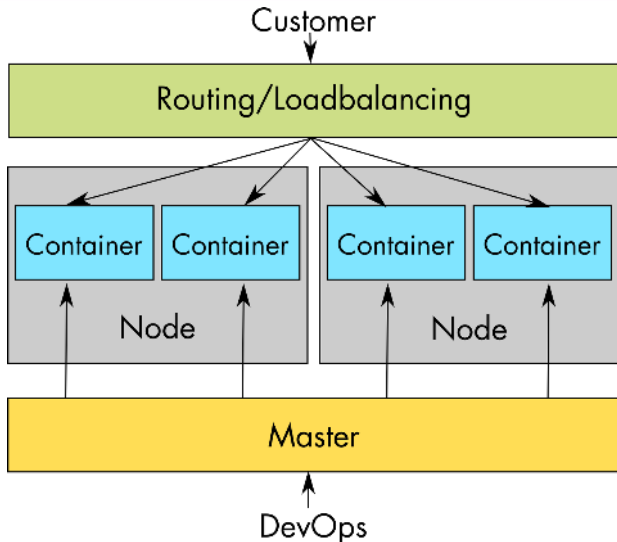
→ Erweiterung der Infrastruktur auf einen Cluster mithilfe von z.B.:

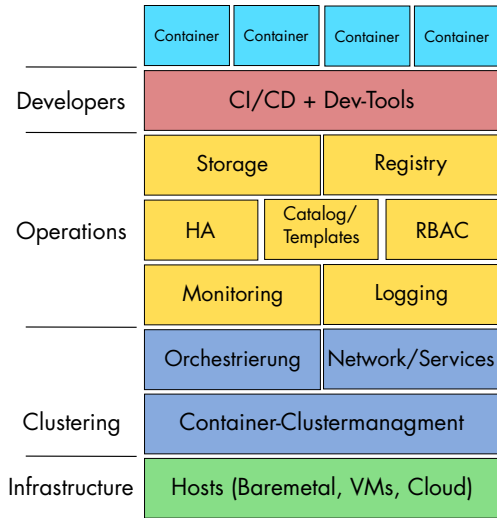


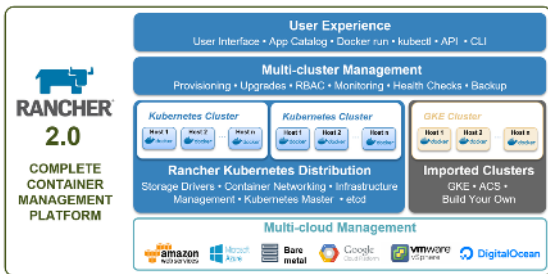
- ▶ Docker Swarm Mode:
 - ▶ Erweiterung auf mehrere Hosts von Docker selbst
- ▶ Rancher mit Cattle:
 - ▶ Unterstützung von Docker, Kubernetes, Mesos, Docker Swarm
 - ▶ Ab Version 2.0 nur mit Kubernetes
- ▶ Kubernetes:
 - ▶ Ursprünglich entwickelt von Google
 - ▶ Fokus auf hohe Skalierbarkeit

Kubernetes inzwischen Industriestandard:

- ▶ Rancher:
Wechsel von Docker zu Kubernetes als Basiskomponente
- ▶ Red Hat:
Openshift mit RHEL Atomic Host
- ▶ Canonical:
Kubernetes mit Ubuntu Core
- ▶ SUSE:
SUSE CaaS mit SLE MicroOS







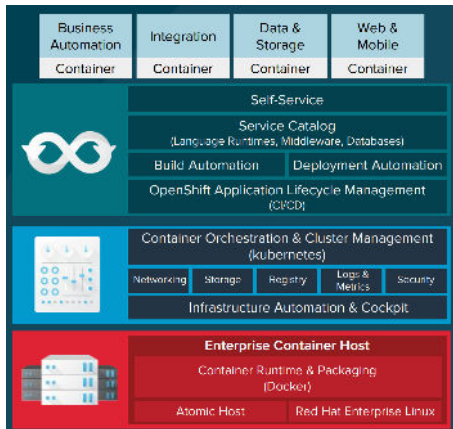
- ▶ Aktuell in Tech Preview 2
- ▶ Beta im Februar 2018
- ▶ Release im März 2018
- ▶ Support von Rancher 1.6 mindestens bis Ende 2018



OPENSIFT

Red Hats Containerplattform:

- ▶ 3.0 Release Juni 2015
- ▶ Aktuelle Version: 3.7
- ▶ Viele ausgebaute Entwicklertools



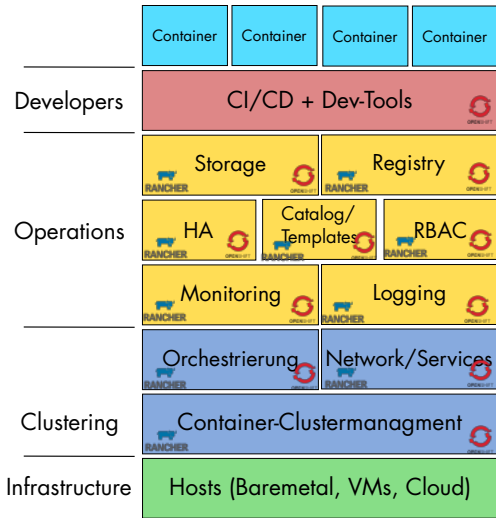
Quelle: Red Hat



- ▶ Open-Source
- ▶ WebGUI + CLI
- ▶ Natives
Kubernetes/Cattle-Kubernetes
- ▶ Aktuell in Tech Preview

- ▶ Open-Source
- ▶ WebGUI + CLI
- ▶ Forked/angepasstes
Kubernetes
- ▶ Stable Release

OpenShift 3.7 vs Rancher 2.0





- ▶ Docker-Host mit Rancher-Server-Container
 - ▶ Konfiguration über WebGUI (SDN, RBAC, etc.)
 - ▶ Starten eines Kubernetes-Cluster in der Cloud
 - ▶ Import von bestehendem Kubernetes-Cluster
- ▶ Ansible-Installer
 - ▶ Initiale Konfiguration über Ansible-Inventory
 - ▶ Umfangreiche Konfigurationsmöglichkeiten
 - ▶ Deutlich komplexer

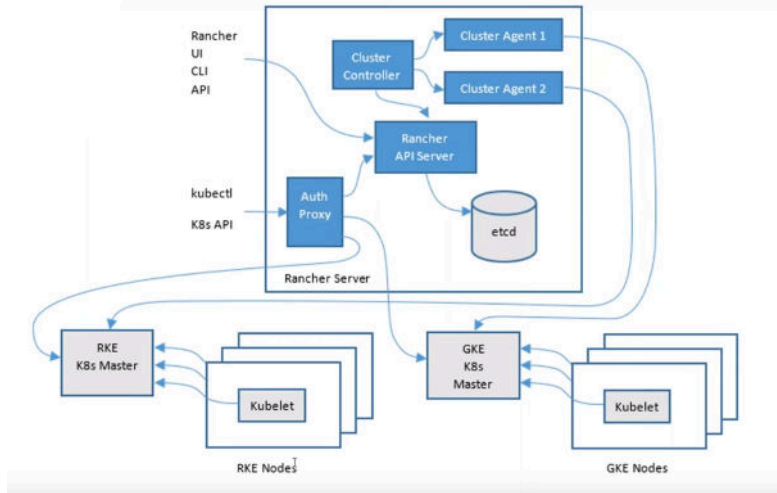


- ▶ Fokus auf Übersichtlichkeit, Intuitivität
- ▶ Adaption der Kubernetes-Namesgebung
- ▶ Unterstützung von kubernetes-yaml, docker-compose, rancher-compose



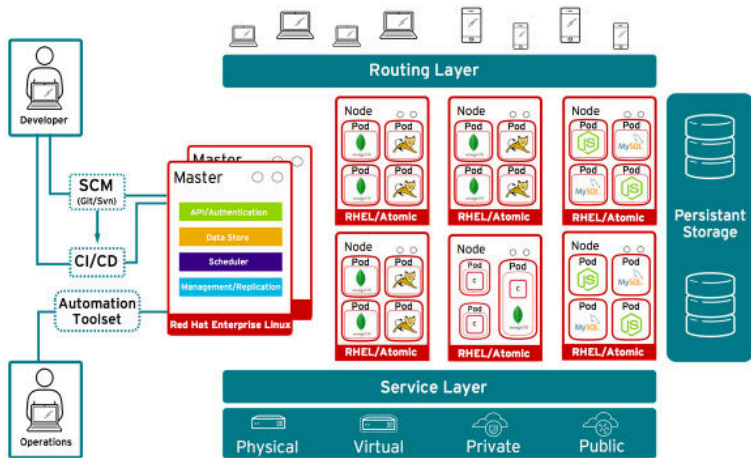
- ▶ Steile Lernkurve
- ▶ Adaption der Kubernetes-Namesgebung
- ▶ Konfiguration in yaml oder json
- ▶ Einfaches Erstellen von Containern über Source-to-Image Builds

Vergleich - Rancher-Aufbau



Quelle: Rancher Meetup-Slides

Vergleich - Openshift-Aufbau



Quelle: blog.openshift.com



- ▶ Aktuell noch keine integrierten Entwicklertools
 - ▶ Communitykatalog mit vorkonfigurierten Templates
 - ▶ Katalog basierend auf Helm
- ▶ Entwicklerfeatures, z.B.:
 - ▶ Integrierter Jenkins
 - ▶ Source-to-Image Builds (S2I)
 - ▶ Schnittstellen zu weiteren Red Hat Produkten (Cloudforms, Ansible)

Persönliches Fazit:

- ▶ Viele Features von Rancher und Openshift ähnlich
 - ▶ Vergleich schwierig (stable vs tech preview/beta)
 - ▶ Openshifts Entwicklertools eindeutiger Vorteil
 - ▶ Rancher intuitiver und einsteigerfreundlicher
- ⇒ Kein "Gewinner" im Vergleich
- ⇒ Abhängig von Anforderungen/Einsatzzweck





Dr. Jonas
Trüstedt

XING: www.xing.com/profile/Jonas_Truestedt

LinkedIn: www.linkedin.com/in/jonas-truestedt