

Software im Container – überall gleich

Docker hat 2013 eine Linux-Technik für Eingeweihte zum Alltagswerkzeug gemacht: Anwendungen samt Abhängigkeiten werden als Image verpackt und laufen auf jedem Rechner mit Docker identisch – schlanker und schneller gestartet als eine virtuelle Maschine.

Zusammenfassung

Docker ist eine Plattform zum Bauen, Verteilen und Ausführen von Anwendungen in Containern. Vorgestellt wurde sie im März 2013 von Solomon Hykes und dem Unternehmen dotCloud (später Docker, Inc.). Die Software ist in Go geschrieben und nutzt Funktionen des Linux-Kernels – Namespaces zur Isolation und Control Groups (cgroups) zur Ressourcenbegrenzung.

Im Mittelpunkt stehen Images und Container: Ein Image ist eine unveränderliche, aus Schichten aufgebaute Vorlage, die in einem Dockerfile beschrieben wird. Ein Container ist die laufende Instanz eines Images. Images werden über Registries wie Docker Hub geteilt, wo es offizielle Images für Datenbanken, Webserver und Programmiersprachen gibt.

Mit Docker Compose lassen sich mehrere zusammengehörige Container – etwa Webanwendung, Datenbank und Cache – in einer einzigen YAML-Datei beschreiben und mit einem Befehl starten. Für grössere Umgebungen gibt es Docker Swarm als eingebauten Orchestrator; in der Praxis hat sich dafür allerdings Kubernetes durchgesetzt, das Container ebenfalls auf Basis von OCI-Standards ausführt.

Docker Engine ist quelloffen. Für Windows und macOS gibt es Docker Desktop, das eine kleine Linux-VM im Hintergrund nutzt; für grössere Unternehmen ist die Nutzung kostenpflichtig. Wer es schlank mag, installiert die Engine direkt auf einem Linux-Server – etwa unter Debian – und verwaltet sie bei Bedarf über Oberflächen wie Portainer.

Eckdaten

Die wichtigsten Eckdaten im Überblick:

Typ
Container-Plattform

Erstveröffentlichung
März 2013
durch dotCloud / Solomon Hykes

Entwickler

Docker, Inc. & Community

Programmiersprache
Go

Lizenz
Apache 2.0
Docker Engine

Kerntechnik
Namespaces & cgroups
des Linux-Kernels

Beschreibung
Dockerfile & compose.yaml

Registry
Docker Hub

Plattformen
Linux, Windows, macOS
Desktop-Varianten mit Hintergrund-VM

Standards
OCI
containerd & runc im Hintergrund

Die bekanntesten Befehle

`docker run -d --name web -p 8080:80 nginx`
Startet einen Container aus einem Image im Hintergrund (`-d`), gibt ihm einen Namen und leitet Port 8080 des Hosts auf Port 80 im Container.

`docker ps` / **`docker ps -a`**
Listet laufende Container, mit `-a` auch gestoppte.

`docker images` / **`docker pull image`**
Zeigt lokale Images bzw. lädt ein Image aus einer Registry wie Docker Hub.

`docker build -t name:tag .`
Baut aus dem Dockerfile im aktuellen Verzeichnis ein Image mit Namen und Tag.

`docker exec -it container bash`
Öffnet eine interaktive Shell in einem laufenden Container.

`docker logs -f container`
Zeigt die Ausgabe eines Containers und folgt neuen Zeilen live.

`docker stop container` / **`docker rm container`**
Stoppt einen Container bzw. löscht ihn. Images werden mit `docker rmi` entfernt.

`docker compose up -d` / **`docker compose down`**

Startet bzw. beendet alle Dienste aus einer `compose.yaml` – der Standardweg für Multi-Container-Setups.

`docker volume ls` / **`docker network ls`**
Listet Volumes (persistente Daten) und Netzwerke auf.



`docker system prune`
Räumt gestoppte Container, ungenutzte Netzwerke und hängende Images auf. Mit `-a` und `--volumes` wird deutlich gründlicher gelöscht – Vorsicht.

`docker inspect container` / **`docker stats`**
Zeigt die vollständige Konfiguration als JSON bzw. die Ressourcennutzung der laufenden Container.

Zitate

„Build once, run anywhere“
Das Versprechen der Container: Was auf dem Laptop läuft, läuft auch auf dem Server – samt allen Abhängigkeiten im Image.

„Works on my machine“ – jetzt mit Beweis
Wenn die Umgebung als Image mitgeliefert wird, verliert der berühmteste Satz der Softwareentwicklung seine Ausrede.

„Ein Prozess, ein Container“
Die Faustregel für saubere Container: kleine, austauschbare Einheiten statt grosser virtueller Maschinen.

Spannendes aus dem Web

WEBSITE	URL
Docker – offizielle Webseite	docker.com →
Docker-Dokumentation	docs.docker.com →
Docker Hub	hub.docker.com →

Diese Geräte könnten dir auch gefallen

- Microsoft
- Windows
- Debian
- 2know
- Linux