

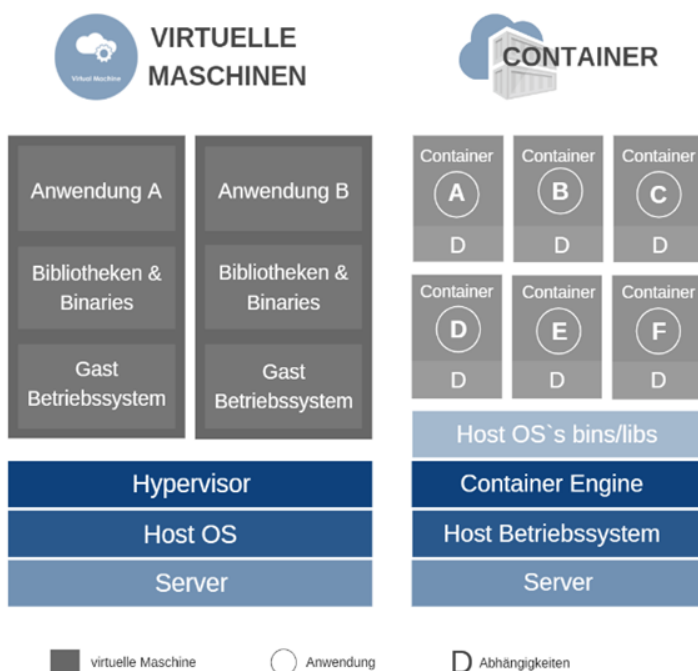
DOCKER CONTAINER VS. VIRTUELLE MASCHINE

Ist Docker mit Containern als isolierte Applikation wirklich die bessere Alternative zu virtuellen Maschinen?

Wir würden diese Frage gerne einfach mit „ja“ oder „nein“ beantworten und wahrscheinlich hast du es schon vermutet – so einfach ist das nicht. Wie Docker als containergestützte Technologie funktioniert und welche Vorteile sich daraus ergeben, kannst du in unserem Artikel [„schneller, besser, Docker“](#) nachlesen. In diesem Blog sprechen wir über die Unterschiede zwischen virtuellen Maschinen und Containern und helfen dir mit der Beschreibung nützlicher Terminologien auf die Sprünge (siehe ab Seite 3).

Virtuelle Maschinen (auch als VM's bezeichnet) sind virtuelle Computer, die dieselben Funktionen wie physische Rechner bieten - sie führen Anwendungen und ein Betriebssystem aus. Bei virtuellen Maschinen handelt es sich jedoch um Computerdateien, die auf einem physischen Computer ausgeführt werden. Sie funktionieren als separate Computersysteme.

Container sind eine einfache Lösung, um Anwendungen in ihren entsprechenden Umgebungen dynamisch bereitzustellen und zu übertragen, ganz ohne Probleme bei der Kompatibilität zu bekommen. Ebenso sind sie zustandslos, voneinander isoliert und persistieren keine Daten, falls dies nicht gewünscht ist.



Die Unterschiede im Betriebssystem

Virtuelle Maschinen führen stets ein vollständiges Betriebssystem samt Kernel aus. Das fordert mehr Systemressourcen, Arbeitsspeicher und Speicherplatz. Docker als ein Container-Typ führt hingegen nur die notwendigen Komponenten eines Betriebssystems aus.

Container lassen sich konfigurieren, sodass nur die Dienste im Container enthalten sind, die wirklich zum Ausführen, zum Beispiel einer App, notwendig sind. Auf einen Server passen mehr Container als virtuelle Maschinen.



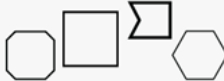



Virtuelle Maschinen sowie Containeranwendungen und Ihre Abhängigkeiten



Unterschiede in Isolierung & Sicherheit

Sollte sich die Software mit einem Virus infizieren, bieten virtuelle Maschinen eine hohe Sicherheit. Sie ermöglichen eine Isolierung vom Betriebssystem des Hosts von anderen virtuellen Maschinen. Container bieten hier vorerst etwas weniger Sicherheit, da die Anwendungen zumeist auf einfachere Art und Weise vom Host und anderen Containern isoliert werden. Die Betonung liegt hierbei auf vorerst, denn die Sicherheit lässt sich erhöhen, zum Beispiel durch die Linux-Kernel-Erweiterung SE-Linux (Security-Enhanced Linux). Diese besteht aus einem Kernel-Patch und Erweiterungen für Systemprogramme und gewährleistet dadurch wiederum einen hohen Grad an Sicherheit.

Die Hauptunterschiede zwischen Virtuellen Maschinen und Containern

	VIRTUELLE MASCHINEN	CONTAINERS
Anlaufzeit	MINUTEN	MILLIESEKUNDE
Speicherplatz		
Übertragbarkeit		
Effizienz		
Betriebssystem/ Kernel	DEDIZIERT	GETEILT

Detaillierte Unterschiede zwischen virtuellen Maschinen und Docker

Werden containergestützte Technologien virtuelle Maschinen vollständig ablösen?

Nun stellt sich eine Frage - haben VMs denn überhaupt noch Ihre Daseinsberechtigung, denn wir lesen viele Vorteile, die für Docker und Container sprechen? Lilli Kaufhold, Senior Data Scientist, hat hierauf eine Antwort und erklärt, welchen Part die containergestützten Technologien im Alltag von Supper und Supper übernehmen.

Bei Supper und Supper verwenden wir beide Technologien – Docker und auch virtuelle Maschinen. Unser Standardanwendungsfall ist es, eine virtuelle Maschine in der Cloud zu starten und dann unsere Dienste in Docker-Containern auf diesen Maschinen bereit zu stellen. Normalerweise kümmert sich der Host-Provider (z.B. Azure oder AWS) um die Einrichtung der VM mit einem bestimmten Image. Wir sehen die echte Hardware darunter nicht. In meinem Tagesgeschäft verwalte ich nur Docker aktiv.

Virtuelle Maschinen spielen daher immer noch eine wichtige Rolle und werden dies auch weiterhin tun. Sie werden für den Benutzer vielleicht in Zukunft nur weniger sichtbar sein, da sie als Standard-Sicherheitsmaßnahme integriert werden.

Wichtige Terminologien in Docker

Wie bereits angekündigt, findest du hier einige wichtige Terminologien, welche dir im Bereich containergestützte Technologien immer wieder über den Weg laufen werden.

Docker-Image

Ein Docker-Image ist eine Datei, die aus mehreren Schichten besteht und zur Ausführung von Code in einem Docker-Container dient. Es handelt sich um eine Reihe von Anweisungen, aus der Docker-Container erstellt werden.

Docker-Container

Ein Docker-Container ist eine Laufzeitinstanz eines Images. Es ermöglicht Entwicklern, Anwendungen mit allen Bestandteilen wie Bibliotheken und anderen Abhängigkeiten zu verpacken.



Docker- File

Hierbei handelt es sich um ein Textdokument, das notwendige Befehle enthält, die bei der Ausführung helfen, ein Docker-Image zusammenzustellen. Das Image wird mithilfe einer Docker-Datei erstellt.

Docker-Engine

Das ist die Software, die die Container hostet und ist eine Client-Server-basierte Anwendung. Sie besteht aus verschiedenen Hauptkomponenten:

SERVER	Der Server ist für die Erstellung und Verwaltung von Docker-Images, Containern, Netzwerken und Volumes auf dem Docker-Server zuständig. Er wird als Dämon-Prozess bezeichnet
REST-API	Die REST-API legt fest, wie die Anwendungen mit dem Server interagieren können. Sie weist ihn an, was zu tun ist.
CLIENT	Der Client ist eine Docker-Befehlszeilenschnittstelle (CLI). Diese ermöglicht es uns, mit Docker unter der Verwendung der Docker-Befehle zu interagieren.
DOCKER HUB	Docker Hub ist das offizielle Online-Repository, indem du andere Docker-Images finden kannst, die zur Verfügung stehen. Du kannst Container-Images finden, verwalten und mit anderen teilen.

Wie wir nun erfahren haben, ist Docker tatsächlich eine gute Alternative zu virtuellen Maschinen, jedoch macht es Sinn und ist auch nach wie vor möglich, auf beide Technologien zurückzugreifen, wie wir bei Supper und Supper.

Hast du noch Fragen, Anregungen oder möchtest einfach Hallo sagen? Kontaktiere uns gerne per Email unter: info@supperundsupper.com.