

# Software erklärt: Was ist Containerisierung?

Geschrieben von Coursera Staff • Aktualisiert am 25. Feb. 2025


Teilen

Erfahren Sie mehr über Containerisierung, eine leistungsstarke Software-Methode, die Code und benötigte Anwendungen in einer einzigen Einheit verpackt, und wie sie in der Praxis funktioniert.



[Read in English \(Auf Englisch lesen\)](#)

Containerisierung ist eine Möglichkeit, Softwarecode und benötigte Ressourcen in eine unabhängige Einheit zu packen, die auf jedem Betriebssystem ausgeführt werden kann. In diesem Artikel erfahren Sie, was Containerisierung ist, wie sie entstanden ist, welche Vorteile Sie davon haben und welche Beispiele es in führenden Branchen gibt, die diese Technik nutzen.



Kurs

### Containerisierte Anwendungen auf AWS

In diesem Kurs lernen Sie Container-Technologien kennen und erfahren, wie Sie damit Ihre Anwendungen modernisieren können. Außerdem lernen Sie, wie ...

★ 4.6

(173 Bewertungen)

25.733 bereits angemeldet

Stufe anfangler

Mehr erfahren

Durchschnittliche Zeit: 12 Stunde(n)

In Ihrem eigenen Lerntempo lernen

Kompetenzen, die Sie erwerben:

Amazon Webdienste, Skalierbarkeit, Docker (Software), Kubernetes, Bereitstellung von Anwendungen, YAML, Systemüberwachung, ... Devops-Tools, Containerisierung, Amazon Elastic Compute Cloud, Microservices, Serverloses Rechnen, Amazon CloudWatch

## Was ist Containerisierung?

Bei der Containerisierung handelt es sich um eine Methode der Softwareentwicklung, bei der eine Anwendung und die zugehörigen Elemente in einem „Container“ verkapselt werden.

Dieser Container dient als autarke Einheit, die es der Anwendung ermöglicht, einheitlich und zuverlässig in verschiedenen Computerumgebungen zu laufen. Der Hauptvorteil dieses Ansatzes besteht darin, dass die Anwendung von allen Inkonsistenzen, die zwischen verschiedenen Systemen bestehen könnten, isoliert wird, indem CPU-, Speicher- und Netzwerkressourcen virtualisiert werden und sichergestellt wird, dass die Leistung der Anwendung konsistent bleibt.

Die Containerisierung bietet einen systematischen Weg, um die Portabilität, Skalierbarkeit und Ressourceneffizienz von Anwendungen zu erreichen. Mit dieser Methodik kann das technische Team den Beteiligten versichern, dass die Software wie vorgesehen funktioniert, egal wo sie eingesetzt wird. Der Container enthält nicht nur die Anwendung, sondern auch ihre Abhängigkeiten, Bibliotheken und Binärdateien, so dass Sie die Software einfach in den Container stecken und ohne großen Aufwand ausführen können.

## Ursprünge der Containerisierung

Das Konzept der Containerisierung ist nicht so neu

Die Wurzeln lassen sich bis zu Unix zurückverfolgen, als Technologien wie „chroot“ in den späten 1970er Jahren eingeführt wurden.

Chroot isolierte Bereiche für laufende Prozesse. Die Containerisierung gewann jedoch mit der Einführung der Docker-Engine im Jahr 2013 an Dynamik. Docker vereinfachte und standardisierte den Prozess der Erstellung, Bereitstellung und Ausführung von Anwendungen in Containern (bekannt als Paketierung) und ebnete damit den Weg für die moderne Containerisierung.

## Wie funktioniert Containerisierung?

Das Hauptziel der Containerisierung besteht darin, autonome und portable Softwarepakete zu erstellen, die unabhängig von der Plattform, auf der sie bereitgestellt werden, konsistent ausgeführt werden. Erreicht wird dies durch die Erstellung und Verwendung von Container-Images, d. h. Dateien, die die für die Ausführung einer Anwendung in einem Container erforderlichen Daten enthalten. Containerebenen sind unveränderlich, d. h. sie können auf dem Hostsystem nicht geändert werden.

In einer containerisierten Architektur sind diese Images die oberste Schicht in einer mehrschichtigen Struktur. Die grundlegende Schicht ist die Infrastrukturebene, die sich auf die eigentliche Hardware bezieht, die für die Ausführung der containerisierten Anwendung verantwortlich ist. Über der Infrastrukturschicht befindet sich das Betriebssystem. Darauf folgt die Container-Engine, eine spezielle Software, die die Erstellung und Verwaltung von Containern erleichtert. Diese Schicht fungiert als Bindeglied zwischen dem Betriebssystem und den Containern, überwacht die Ressourcennutzung und stellt sicher, dass die Container voneinander und von der zugrunde liegenden Infrastruktur isoliert bleiben. Schließlich gibt es noch die Anwendungs- und Abhängigkeitsschicht, die den eigentlichen Anwendungscode zusammen mit den zugehörigen Bibliotheken und Konfigurationen enthält.

Durch diesen strukturierten Ansatz der Containerisierung werden Anwendungen systematisch isoliert und verwaltet, wodurch Konsistenz, Übertragbarkeit und Effizienz über verschiedene Computersysteme hinweg gewährleistet werden.

## Vorteile der Containerisierung

Die Containerisierung bietet eine ganze Reihe von Vorteilen, die sie zu einer beliebten Option für die Softwarebereitstellung und -verwaltung machen. Hier sind einige der wichtigsten Vorteile:

- **Agile Entwicklungsumgebung:** Container lassen sich im Vergleich zu virtuellen Maschinen (VMs) einfacher einrichten. Dadurch können DevOps-Teams ihre Entwicklungs-, Paketierungs- und Bereitstellungsabläufe beschleunigen.
- **Hohe Portabilität:** Einmal geschrieben, können Container auf jedem System ausgeführt werden.
- **Effizient:** Container benötigen weniger Ressourcen als VMs, was die Kosten senkt, die Skalierung erleichtert und zu schnelleren Boot-Zeiten führt.
- **Erhöhte Sicherheit:** Container isolieren Anwendungen nicht nur vom Hostsystem, sondern auch voneinander. Diese Isolierung erhöht die Sicherheit und minimiert die potenziellen Auswirkungen von Systemschwachstellen.
- **Isolierung von Fehlern:** Wenn ein Container ausfällt, hat dies keine Auswirkungen auf die anderen. Dies kommt Microservices-basierten Anwendungen zugute, bei denen mehrere Komponenten zur Funktionalität eines größeren Programms beitragen. Sie können bestimmte Microservices reparieren oder skalieren, ohne einen systemweiten Ausfall zu verursachen.

## Nachteile der Containerisierung

Die Verwendung von Containern bietet viele Vorteile, kann aber auch einige Nachteile mit sich bringen. Zu den Nachteilen der Containerisierung gehören:

- **Nicht universell:** Obwohl Container effizient, vielseitig und portabel sind, gibt es monolithische Anwendungen, die nicht unbedingt Skalierbarkeit oder eine schnelle Bereitstellung benötigen, und VMs sind besser geeignet.
- **Grenzen bei Abhängigkeiten:** Da Container von einem physischen Betriebssystem abhängig sind und auf diesem aufbauen, können sie aufgrund von Abhängigkeiten weniger portabel sein.
- **Begrenzte Tools:** Für die Überwachung und Verwaltung von Containern benötigen Sie Tools. In der Branche herrscht jedoch ein Mangel an modernen Tools.

## Wer nutzt Containerisierung? Beispiele aus der Praxis

Die Containerisierung ist in der Welt von heute unglaublich populär. Um die Auswirkungen der Containerisierung in der Praxis besser zu verstehen, sollten Sie sich einige praktische Beispiele aus verschiedenen Branchen und Anwendungen ansehen. Hier sind einige bekannte Unternehmen, die Container verwenden:

- **Netflix:** Netflix nutzt die Containerisierung, um die Skalierung seiner Dienste zu erleichtern und die Anforderungen von Millionen von Nutzern zu erfüllen. Netflix hat eine Container-Management-Plattform namens Titus entwickelt, die Streaming, Inhaltssysteme und Benutzerempfehlungen unterstützt.
- **Uber:** Uber nutzte die Containerisierung, um seine wachsende Nutzerbasis effizienter zu verwalten. Dies geschah durch die Skalierung von Apache Hadoop mit Docker-Containern.



- **Salesforce:** Salesforce nutzt Container, um robustere und anpassbare Lösungen für seine Kunden zu ermöglichen. Die Liste der Container, die für Anwendungen, Builder-Sites, Integration und eingebettete Dienste verwendet werden, wächst.

## Containerisierung vs. Virtualisierung

Containerisierung und Virtualisierung sind beides Technologien, mit denen isolierte Umgebungen für die Ausführung von Anwendungen geschaffen werden können, allerdings auf unterschiedliche Weise.

Container sind leichter und schneller, da sie das Betriebssystem und die Ressourcen des Hostsystems gemeinsam nutzen. Dadurch sind sie hochgradig portabel und schnell einsatzbereit und eignen sich gut für Szenarien, die Agilität erfordern, wie z. B. Microservices und DevOps.

Virtuelle Maschinen hingegen arbeiten mit einem eigenen, vollständigen Betriebssystem, was sie ressourcenintensiver und langsamer beim Hochfahren macht. VMs schaffen virtualisierte Computing-Umgebungen, wie die erste Generation der Cloud. Sie sind die Virtualisierung von physischer Hardware, während Container die Virtualisierung eines Betriebssystems sind. Eine VM wird zu einem „Gast“ des Host-Computers, was bedeutet, dass Sie das Gastbetriebssystem bearbeiten können, ohne den Host zu gefährden.

Die Containerisierung wird häufig aus Gründen der Flexibilität und der einfachen Verwaltung eingesetzt, während die Virtualisierung gewählt wird, wenn Teams mehrere Betriebssysteme auf einem einzigen Rechner ausführen möchten. Je nach Ihren Bedürfnissen und Prioritäten können Sie die für Sie richtige Lösung wählen.


## Jobs, die Container verwenden

Im Zuge der Weiterentwicklung der Containertechnologie haben sich in der Technologiebranche verschiedene Berufe herausgebildet, die sich speziell auf die Nutzung der Leistungsfähigkeit von Containern konzentrieren. Im Folgenden finden Sie einige Berufsbezeichnungen, die mit Containerisierung zu tun haben:

- **DevOps Engineer:** DevOps-Ingenieure verwenden Container für Entwicklungs- und Betriebsaufgaben.
- **Cloud-Architect:** Da Cloud-Dienste häufig Container zur Verwaltung von Anwendungen einsetzen, entwerfen und orchestrieren Cloud-Architekten Cloud-Lösungen, die containerisierte Umgebungen nutzen können.
- **Site Reliability Engineer (SRE):** SREs nutzen die Containerisierung, um eine optimale Verwaltung und Leistung zu gewährleisten. Sie können containerisierte Anwendungen für verschiedene Anwendungen auf unterschiedlichen Plattformen verwenden.


## Erste Schritte mit Coursera


Auf Coursera können Sie die Containerisierung mit erstklassigen Kursen von Branchenexperten weiter erforschen. Sie können den Kurs [Containerisierte Anwendungen auf AWS](#) von Amazon absolvieren, um die Containerisierung in nur 12 Stunden zu erkunden.



**Kurs**

## Containerisierte Anwendungen auf AWS

In diesen Kurs lernen Sie Container-Technologien kennen und erfahren, wie Sie damit Ihre Anwendungen modernisieren können. Außerdem lernen Sie, wie...


**4.6** (137 Bewertungen)


**25.733** bereits angemeldet


**Stufe** anfänger





Mehr erfahren

### Durchschnittliche Note: 12,2 Stunden(n)


In ihrem eigenen Lerntempo lernen

### Kompetenzen, die Sie erwerben:

Amazon Webdienste, Skallerbarkeit, Docker (Software), Kubernetes, Bereitstellung von Anwendungen, YAML, Systemüberwachung,... Devops-Tools, Containerisierung, Amazon Elastic Compute Cloud, Amazon Services, Serverloses Rechnen, Amazon CloudWatch





**Teilen**

Aktualisiert am 25. Feb. 2025  
 Geschrieben von:


**Coursera Staff Redaction**  
 Das Redaktionssteam von Coursera besteht aus äußerst erfahrenen professionellen Redakteuren, Autoren ...

Diese Inhalte dienen nur zu Informationszwecken. Den Lernenden wird empfohlen, eingehender zu recherchieren, ob Kurse und andere angestrebte Qualifikationen wirklich ihren persönlichen, beruflichen und finanziellen Vorstellungen entsprechen.

## Technische Fertigkeiten

- ChatGPT
- Programmieren
- Informatik
- Cybersicherheit
- DevOps
- Ethisches Hacking
- Generative KI
- Java Programmierung
- Python
- Webentwicklung

## Analytische Fähigkeiten

- Künstliche Intelligenz
- Big Data
- Unternehmensanalyse
- Datenanalyse
- Datenverarbeitung
- Finanzplanung
- Maschinelles Lernen
- Microsoft Excel
- Microsoft Power BI
- SQL

## Business-Fähigkeiten

- Buchhaltung
- Digitales Marketing
- E-Commerce
- Finanzen
- Google
- Grafikdesign
- IBM
- Marketing
- Projektmanagement
- Social Media-Marketing

## Karriere-Ressourcen

Wichtige IT-Zertifizierungen

Einkommensstarke Fähigkeiten zu erlernen

So erwerben Sie eine PMP-Zertifizierung

Wie man künstliche Intelligenz lernt

Beliebte Zertifizierungen für Cybersicherheit

Beliebte Datenanalyse-Zertifizierungen

Was macht ein Datenanalyst?

Ressourcen für die berufliche Entwicklung

Berufseignungstest

Teilen Sie Ihre Coursera Lerngeschichte

Coursera

- Info
- Was wir anbieten
- Leitung
- Jobs
- Katalog
- Coursera Plus
- Berufsbezogene Zertifikate
- MasterTrack® Certificates
- Abschlüsse
- Für Unternehmen
- Für Regierungen
- Für Campus
- Werden Sie Partner
- Soziale Auswirkung
- Kostenlose Kurse
- ECTS-Credit-Empfehlungen

## Community

- Kursteilnehmer
- Partner
- Beta-Tester
- Blog
- Der Coursera-Podcast
- Tech-Blog
- Lehrzentrum

Mehr

[Presse](#)  
[Anleger](#)  
[Nutzungsbedingungen/AGB](#)  
[Datenschutz](#)  
[Hilfe](#)  
[Barrierefreiheit](#)  
[Kontakt](#)  
[Artikel](#)  
[Verzeichnis](#)  
[Partnerunternehmen](#)  
[Stellungnahme zu moderner Sklaverei](#)  
[Cookie-Einstellungen verwalten](#)

